

*Communications in
Applied
Mathematics and
Computational
Science*

**DISCRETE NONHOMOGENEOUS AND
NONSTATIONARY LOGISTIC AND MARKOV
REGRESSION MODELS
FOR SPATIOTEMPORAL DATA
WITH UNRESOLVED EXTERNAL INFLUENCES**
JANA DE WILJES, LARS PUTZIG AND ILLIA HORENKO

vol. 9 no. 1 2014

DISCRETE NONHOMOGENEOUS AND NONSTATIONARY LOGISTIC AND MARKOV REGRESSION MODELS FOR SPATIOTEMPORAL DATA WITH UNRESOLVED EXTERNAL INFLUENCES

JANA DE WILJES, LARS PUTZIG AND ILLIA HORENKO

Dynamical systems with different characteristic behavior at multiple scales can be modeled with hybrid methods combining a discrete model (e.g., corresponding to the microscale) triggered by a continuous mechanism and vice versa. A data-driven black-box-type framework is proposed, where the discrete model is parametrized with adaptive regression techniques and the output of the continuous counterpart (e.g., output of partial differential equations) is coupled to the discrete system of interest in the form of a fixed exogenous time series of external factors. Data availability represents a significant issue for this type of coupled discrete-continuous model, and it is shown that missing information/observations can be incorporated in the model via a nonstationary and nonhomogeneous formulation. An unbiased estimator for the discrete model dynamics in presence of unobserved external impacts is derived and used to construct a data-based nonstationary and nonhomogeneous parameter estimator based on an appropriately regularized spatiotemporal clustering algorithm. One-step and long-term predictions are considered, and a new Bayesian approach to discrete data assimilation of hidden information is proposed. To illustrate our method, we apply it to synthetic data sets and compare it with standard techniques of the machine-learning community (such as maximum-likelihood estimation, artificial neural networks and support vector machines).

1. Introduction

Discrete/categorical dynamical processes with a finite state space represent a challenge for standard data-based analysis tools. Heterogeneity of model properties over time and space as well as the discreteness of the data complicate the employment of standard time-series analysis techniques. Moreover, parametrization of the underlying process is often hampered by incompleteness of observational data.

Illia Horenko is the corresponding author.

MSC2010: primary 62-07, 62H30, 62M05, 62M10, 65C60; secondary 62M02, 62M20, 62M30, 62M45, 62H11.

Keywords: nonstationary, nonhomogeneous, discrete spatiotemporal time-series analysis, Markov regression, logistic, data assimilation.

In this paper, we want to address these problems by introducing a nonstationary, nonhomogeneous regression framework that allows taking a lack of observed information into account.

Adequate modeling and proper statistical handling of discrete processes (e.g., jump processes) is especially important for the proper description of multiscale dynamical systems. A typical modeling approach to multiscale dynamical systems is based on the employment of hybrid models, consisting of continuous and discrete model components [19; 20; 21]. While the continuous dynamics can be described with suitable PDEs, the discrete model can be estimated with appropriate data-based analysis methods. Communication between the two models can be achieved via incorporating the continuous data components (e.g., the output of PDEs or ODEs) as external statistical impact factors (or *covariates*) in the discrete part of the model.

Regression analysis [11] or pattern-recognition techniques such as artificial neural networks (ANN) [2; 24] or support vector machines (SVM) [8; 35] are popular instruments to approach the parametrization of dynamical processes. A common ansatz to model discrete-, categorical- and jump-processes is to deploy discrete choice models (e.g., logit or probit regression), which belong to the family of generalized linear models (GLM) [12; 10]. However, these classical techniques are usually restricted to time-independent model parameters, i.e., stationary models.

In this manuscript, we propose a nonstationary logistic regression model and also provide a direct approach to the discrete structure in the form of a nonstationary Markov regression. The key advantage of the proposed framework is that it allows us to parametrize the considered dynamical system corresponding to the data while taking all external influences into account, even those not explicitly available in the form of observation data. This is achieved by introducing an explicit dependency of the model parameters on time and location, i.e., by including an explicit temporal nonstationarity and spatial nonhomogeneity into the resulting model. Necessary assumptions and details will be given in [Proposition 2.1](#). A new numerical algorithm for the solution of the obtained inverse problem is formulated, and its numerical complexity is compared with the complexities of the standard algorithms of discrete data analysis. An adapted version of Akaike's information criterion is used to determine the best model fit corresponding to the data [30]. The resulting optimal parameters can then be employed to make predictions about future states of the process. In this context, a Bayesian approach to assimilate new hidden information (describing the impact of unresolved external factors) is proposed. Training and testing of the techniques are done on several sets of synthetic data, and the quality of one-step and long-term predictions is investigated.

The remainder of the paper is structured as follows. In [Section 2](#), spatiotemporal ensemble data is considered and the possibility to incorporate implicit external factors via a nonstationary Markov model formulation is demonstrated. A short

introduction to the nonstationary spatiotemporal Markov and logistic regression is given in [Section 3](#), where new aspects are emphasized and existing theory is reviewed. In [Section 4](#), a self-containing strategy to make predictions by means of the determined model parameters and a new approach to assimilate additional hidden data after obtaining new observations are introduced. Proposed methods of discrete data modeling, prediction and assimilation are investigated numerically in [Section 5](#) for different synthetic model scenarios and systematically compared to the standard methods of the machine learning community, i.e., ANN [2; 24; 18; 3] and SVM [8; 35]. A comparison of the different numerical methods is given in terms of the information content (i.e., Akaike information criterion) and the quality of long- and short-term data-based online model predictions.

2. Ensemble data and exterior quantities

In the following, the discrete state $s_i \in \{s_1, \dots, s_{N_s}\}$ of a microscopic cell $\omega(j, l)$, with $l \in \{1, \dots, N_{\text{ens}}\}$ being the index of cells of a lattice on a microscopic level and $j \in \{1, \dots, N_j\}$ the corresponding macroscopic cell, is considered. Put differently, a macroscopic lattice, with each cell being further subdivided into smaller grid cells of a microscopic scale, is regarded. It is assumed that it is possible to assign each microscopic cell $\omega(j, l)$ its discrete state s_i via a stochastic process $\sigma(t, j, l)$ dependent on the time $t \in \{1, \dots, N_T\}$. Discrete dynamical systems of such form are common natural phenomena, e.g., representing the spatiotemporal dynamics of changes in the aggregate states of water in climate/atmosphere/ocean sciences. However, such systems represent a challenge for existing data-based analysis tools as it is usually not possible to have access to the corresponding data on a microscopic scale. Since observations of a single discrete realization $\sigma(t, j, l)$ in many realistic applications are not directly accessible, one resorts to the often available information on relative frequencies (with respect to the states) of a finite ensemble of microscopic locations on a macroscopic level. In detail, this means considering all the cells $\omega(j, l)$ with $l \in \{1, \dots, N_{\text{ens}}\}$ for fixed j (corresponding to the macroscopic scale) and measuring/observing the empirical probability

$$\tilde{\pi}_i(t, j) = \frac{N_{s_i}(t, j)}{N_{\text{ens}}}, \quad (1)$$

which is the ratio of $N_{s_i}(t, j)$, the number of cells $\omega(j, l)$ currently (i.e., for fixed time t) in state s_i , to N_{ens} , the total number of microscopic lattice cells contained in each macroscopic grid location (i.e., for fixed j). Formally, the total number of microscopic cells $\omega(j, l)$ currently in state s_i is defined as

$$N_{s_i}(t, j) = \sum_{l=1}^{N_{\text{ens}}} \delta_{s_i}(\sigma(t, j, l)), \quad (2)$$

whereas $\delta_{s_i}(\cdot)$ is the Kronecker delta for the value s_i , i.e., $\delta_{s_i}(\sigma(t, j, l)) = 1$ if $\sigma(t, j, l) = s_i$, otherwise it is zero. Further, a vector of empirical probabilities

$$\tilde{\pi}(t, j) = \begin{bmatrix} \tilde{\pi}_1(t, j) \\ \vdots \\ \tilde{\pi}_{N_S}(t, j) \end{bmatrix} \in [0, 1]^{N_S \times 1} \quad (3)$$

is a good estimate of the actual probability distribution as the number of microscopic cells N_{ens} in each macroscopic cell j is usually exceptionally large, i.e.,

$$\pi_i(t, j) := \mathbb{P}[\sigma(t, j, l) = s_i] \approx \tilde{\pi}_i(t, j) \quad (4)$$

with

$$\pi(t, j) = \begin{bmatrix} \pi_1(t, j) \\ \vdots \\ \pi_{N_S}(t, j) \end{bmatrix} \in [0, 1]^{N_S \times 1}. \quad (5)$$

Thus, for the remainder of this manuscript, we assume that the observed relative frequencies are equal to the probabilities, i.e., that $\pi(t, j)$ can be observed for $t \in \{1, \dots, N_T\}$ and $j \in \{1, \dots, N_J\}$. Further, it is assumed that the process σ is driven by time- and space-dependent external forces $\bar{u}(t, j) \in \mathbb{R}^{N_F \times 1}$, influencing the underlying system. A graphical interpretation of the discrete dynamical process σ by means of an example realization $\sigma(t, j, l)$ for fixed time t with only two possible states s_1 and s_2 (displayed in gray and white) is shown in [Figure 1](#). The image also displays the relation of the different lattice scales; i.e., each macroscopic cell contains a microscopic lattice with N_{ens} cells.

In the following, the aim is to approximate the dynamical system of interest underlying the stochastic process σ with data-based analysis tools by means of observations $\pi(t, j)$ and available measurements of exterior influencing quantities $\bar{u}(t, j)$.

Implicit external factors. In the following section, we will continue under the assumption that the stochastic process $\sigma(t, j, l)$ is a Markov process, i.e., the probability of the process to be in state s_i depends on the time-wise previous state.¹ A Markov process can be described via a transition matrix $P(\bar{u}(t, j)) \in [0, 1]^{N_S \times N_S}$. The transition probabilities $\pi(t+1, j)$ for the next time step can then be expressed through the so-called master equation:

$$\pi(t+1, j)^\top = \pi(t, j)^\top P(\bar{u}(t, j)). \quad (6)$$

Simultaneous measurement/modeling of all of the external factor components may impose a serious problem for realistic applications as it is impossible to have

¹Existing spatial correlations are going to be considered by including information on neighboring cells in the external factors.

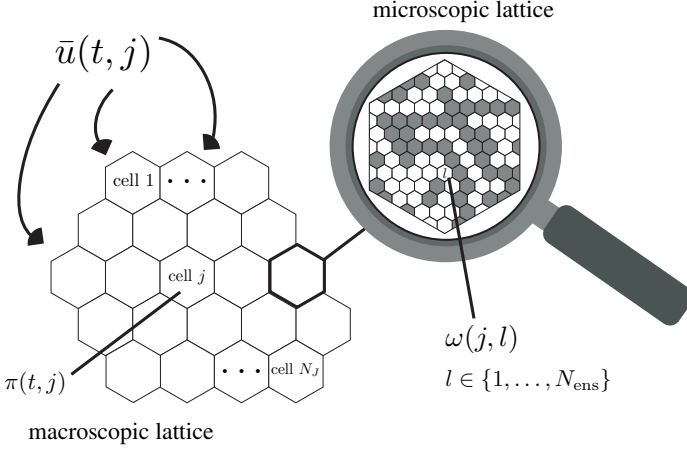


Figure 1. The above figure displays a graphical interpretation of the relation between the microscopic locations $\omega(j, l)$ and the macroscopic observation $\pi(t, j)$. The time t is fixed, and the considered system has two states, i.e., $N_S = 2$, which are displayed in white and gray. Thus, the process $\sigma(t, j, l)$ takes values in the set $\{s_1, s_2\} = \{\text{white, gray}\}$. The honeycomb lattice on the left-hand side corresponds to the macroscopic cells $j \in \{1, \dots, N_J\}$ associated with the observations $\pi(t, j)$. The microscopic lattice indexed $l \in \{1, \dots, N_{\text{ens}}\}$ is illustrated using a fine grid only clearly visible with a magnifying glass (see the hexagonal lattice on the right) and is contained in each cell of the coarse-grid. Additionally, the dependence of the dynamics of σ on external factors $\bar{u}(t, j)$ is visualized.

access to all the quantities influencing a system of interest in general. Therefore, in the following, we will distinguish between explicit and implicit external factors

$$\bar{u}(t, j) = \begin{bmatrix} u(t, j) \\ u^{\text{unres}}(t, j) \end{bmatrix} \in \mathbb{R}^{(N_E + N_I) \times 1} \quad (7)$$

and consider the known

$$u(t, j) = \begin{bmatrix} u_1(t, j) \\ \vdots \\ u_{N_E}(t, j) \end{bmatrix} \in \mathcal{U} \subset \mathbb{R}^{N_E \times 1} \quad (8)$$

as well as the unresolved factors

$$u^{\text{unres}}(t, j) = \begin{bmatrix} u_1^{\text{unres}}(t, j) \\ \vdots \\ u_{N_I}^{\text{unres}}(t, j) \end{bmatrix} \in \mathbb{R}^{N_I \times 1}, \quad (9)$$

according to their availability in the measurement/observation process.² It is important to stress that a vector of external factors $\bar{u}(t, j)$ consists of any quantities potentially playing a role in the dynamics of the regarded system including random,

² $N_F = N_E + N_I$.

deterministic or artificially added elements. For instance, the vector can contain influences other than the currently regarded scales $t \in \{1, \dots, N_T\}$ and $j \in \{1, \dots, N_J\}$ (time-wise as well as location-wise). Specifically, this means that important external forces coming from the microscopic scale as well as exterior factors having an impact on the state of the microscopic grid cells are included in $u^{\text{unres}}(t, j)$. Note, in particular, that the vector of implicit external factors is, as already mentioned above, not limited to deterministic factors but can have stochastic random processes as entries. Further, in order to consider existing spatial correlations, the mean of previous neighboring cell states that are calculated from the observational data $\pi(t-1, j)$ are added to the vector of explicit external factors representing another example of the wide range of possible and allowed quantities contained in $\bar{u}(t, j)$. Along the lines of [16], the abstract dependency of the transition matrix $P(\bar{u}(t, j))$ on unresolved external factors $u^{\text{unres}}(t, j)$ is approached by approximating the matrix with an appropriate linear combination of explicitly time- and space-dependent matrices. Specifically, such a nonstationary and nonhomogeneous formulation is possible under the following conditions:

Proposition 2.1. (1) *If the function $P(\bar{u}(t, j))$ is continuously differentiable and has bounded second derivatives, it can be decomposed in the form*

$$P(\bar{u}(t, j)) = P_0(t, j) + \sum_{e=1}^{N_E} P_e(t, j)u_e(t, j) + \varepsilon(t, j) \quad (10)$$

with $\mathbf{E}[\varepsilon(t, j)] = 0$ and $P_e(t, j) \in \mathbb{R}^{N_S \times N_S}$.

- (2) *If in addition to (1) the deviations of the entries of vector $\bar{u}(t, j)$ from their respective means are statistically independent in j and t , also the different realizations of $\varepsilon(t, j)$ are independent of each other in j and t .³*
- (3) *If the function $P(\bar{u}(t, j))$ is three times continuously differentiable and has bounded third derivatives, it can be decomposed in the form*

$$P(\bar{u}(t, j)) = P_0(t, j) + \sum_{e=1}^{N_E} (P_e(t, j) + \rho_e(t, j))u_e(t, j) + \varepsilon(t, j) \quad (11)$$

with $\mathbf{E}[\varepsilon(t, j)] = 0$ and $\mathbf{E}[\rho_e(t, j)] = 0$. Realizations of the noise process $\rho_e(t, j)$ for different t, j and e are not necessarily independent of each other or of $\varepsilon(t, j)$ realizations.

Proof. (1) For this proof, without loss of generality, we will assume that the external factors are ordered such that the explicit factors are the first N_E entries of $\bar{u}(t, j)$.

³ This does not necessarily imply that $\varepsilon(t, j)$ should also be identically distributed, i.e., i.i.d.

By performing a Taylor expansion on the transition matrix $P(\bar{u}(t, j))$ around the means $\mu(t, j) = [\mathbf{E}(\bar{u}_1(t, j)), \dots, \mathbf{E}(\bar{u}_{N_E+N_I}(t, j))] \in \mathbb{R}^{(N_E+N_I) \times 1}$, we obtain

$$P(\bar{u}(t, j)) = P(\mu(t, j)) + \sum_{e=1}^{N_E} \frac{\partial P(\mu(t, j))}{\partial \bar{u}_e(t, j)} (\bar{u}_e(t, j) - \mu_e(t, j)) + \sum_{|\alpha|=2} R_\alpha(\bar{u}(t, j)) (\bar{u}(t, j) - \mu(t, j))^\alpha, \quad (12)$$

where α is a multi-index and

$$R_\alpha(\bar{u}(t, j)) = \frac{2}{\alpha!} \int_0^1 (1-x) D^\alpha P(\mu(t, j) + x(\bar{u}(t, j) - \mu(t, j))) dx. \quad (13)$$

Note that $R_\alpha(\bar{u}(t, j))$ is bounded as the second derivatives of P are assumed to be bounded. Resorting the terms and defining

$$P_e(t, j) = \frac{\partial P(\mu(t, j))}{\partial \bar{u}_e(t, j)}, \quad e = 1, \dots, N_E, \quad (14)$$

$$\begin{aligned} \varepsilon(t, j) = & \sum_{e=N_E+1}^{N_E+N_I} \frac{\partial P(\mu(t, j))}{\partial \bar{u}_e(t, j)} (\bar{u}_e(t, j) - \mu_e(t, j)) \\ & + \sum_{|\alpha|=2} R_\alpha(\bar{u}(t, j)) (\bar{u}(t, j) - \mu(t, j))^\alpha \\ & - \mathbf{E} \left[\sum_{e=N_E+1}^{N_E+N_I} \frac{\partial \bar{P}(\mu(t, j))}{\partial \bar{u}_e(t, j)} (\bar{u}_e(t, j) - \mu_e(t, j)) \right. \\ & \left. + \sum_{|\alpha|=2} R_\alpha(\bar{u}(t, j)) (\bar{u}(t, j) - \mu(t, j))^\alpha \right], \quad (15) \end{aligned}$$

$$\begin{aligned} P_0(t, j) = & P(\mu(t, j)) - \sum_{e=1}^{N_E} \frac{\partial P(\mu(t, j))}{\partial \bar{u}_e(t, j)} \mu_e(t, j) \\ & + \mathbf{E} \left[\sum_{e=N_E+1}^{N_E+N_I} \frac{\partial P(\mu(t, j))}{\partial \bar{u}_e(t, j)} (\bar{u}_e(t, j) - \mu_e(t, j)) \right. \\ & \left. + \sum_{|\alpha|=2} R_\alpha(\bar{u}(t, j)) (\bar{u}(t, j) - \mu(t, j))^\alpha \right] \quad (16) \end{aligned}$$

yields (10) and especially $\mathbf{E}[\varepsilon(t, j)] = 0$.

(2) If the entries of the vector $\bar{u}(t, j) - \mu(t, j)$ for fixed $e \in \{1, \dots, N_F\}$ are independent for all j and t , the $\varepsilon(t, j)$ (as defined above) are just functions of the independent variables; thus, they are independent of each other again.

(3) The proof of this statement is given in [Appendix A](#). \square

Remarks 2.2. • The noise processes $\rho_e(t, j)$ and $\varepsilon(t, j)$ are not pairwise independent for fixed j and t . Further, there are no a priori assumptions concerning the distribution of $\rho_e(t, j)$ and $\varepsilon(t, j)$.

- Although the error $\varepsilon(t, j)$ is expected to be close to zero, it is important to mention that the variance of $\varepsilon(t, j)$ can take any value and therefore can lead to an arbitrary error term. This problem occurs most likely when the main influencing quantities are not available in the form of observational data.
- As the result of the proposition, the two expansions (10) and (11) deploy two conceptually different models of the noise for the master equation (6). Whereas (10) deploys a purely additive noise term, next-order expansion (11) contains a mixture of additive and multiplicative noise processes. Because of its simplicity, expansion (10) will be used for the construction of the nonhomogeneous and nonstationary data-driven Markov estimators in [Section 3](#).

Summarizing, an approach to address the predicament of missing data, specifically in the context of external influences, is proposed for dynamical system with an underlying Markovian process. It is assumed that the transition matrix has a linear structure so that the implicit dependency on unresolved external factors can be reflected in the explicit dependency on time and location.

3. Method

In this section we introduce methods for the analysis of discrete spatiotemporal data. As the details of nonstationary analysis of temporal data have already been addressed in earlier papers [[16](#); [17](#); [9](#); [30](#)], we will restrict this introduction to a short overview and will only emphasize new aspects concerning, e.g., the spatial component of the data or the details concerning the logistic regression.

3A. Inverse problem formulation. For a general consideration of the observed processes $\sigma(t, j, l)$, we assume that the correlation between the dynamical system and the measurements $\pi(t, j) \in [0, 1]^{N_S \times 1}$ can be expressed with a *direct mathematical model*

$$\pi(t+1, j) = f(\pi(t, j), \dots, \pi(t - N_M, j), \theta(\bar{u}(t, j))), \quad (17)$$

defined by a model function $f(\cdot)$ dependent on current and previous observations up to a memory depth N_M and model parameters $\theta(\bar{u}(t, j))$ from some parameter space Ω dependent on external factors $\bar{u}(t, j) \in \mathbb{R}^{N_F \times 1}$. Note that $\bar{u}(t, j)$ is a vector of all influences driving the system of interest. In particular, it can include information from the microscopic scale (e.g., from locations $\omega(j, l)$ with $l \in \{1, \dots, N_{\text{ens}}\}$) and other spatial components (e.g., neighboring cells), thus allowing to model

any existing spatial correlations. Further, the analytic expression of the model function f can also include random processes, e.g.,

$$f(\theta(t, j)) := \theta(t, j) + \lambda(t, j). \quad (18)$$

In this basic example, the random process $\lambda(t, j)$ has an expected value zero for all t and j , is i.i.d. (independent identically distributed) and can be interpreted as measurement errors or implicit quantities influencing the considered system. The reader is referred to [30] for more model function examples. For a given model function f and parameter function $\theta(\bar{u}(t, j))$, the problem of finding an appropriate time series $\pi(t, j)$ is called the *direct mathematical problem*. In this manuscript, we consider the opposite *inverse problem*: given the observations $\pi(t, j)$, which parameters $\theta(\bar{u}(t, j))$ with respect to the model function f describe the data “best”? In order to find model parameters $\theta(\bar{u}(t, j))$ that minimize the “distance” between the data and the model-based time series, we need to introduce a measuring functional

$$g(\pi(t+1, j), \dots, \pi(t-N_M, j), \theta(\bar{u}(t, j))) : \\ [0, 1]^{N_S} \times \dots \times [0, 1]^{N_S} \times \Omega \rightarrow \mathbb{R}_{\geq 0}, \quad (19)$$

which we will refer to as a *model distance function*. The corresponding inverse problem is defined as

$$\mathbf{L}(\theta(\bar{u}(t, j))) \\ = \sum_{t=1}^{N_T} \sum_{j=1}^{N_J} g(\pi(t+1, j), \dots, \pi(t-N_M, j), \theta(\bar{u}(t, j))) \rightarrow \min_{\theta(\bar{u}(t, j))} \quad (20)$$

and is referred to as an *averaged clustering functional*. A suitable function g can be derived from any metric $d(\cdot, \cdot)$:

$$g(\pi(t+1, j), \dots, \pi(t-N_M, j), \theta(\bar{u}(t, j))) \\ = (d(\pi(t+1, j), \mathbf{E}[f(\pi(t, j), \dots, \pi(t-N_M, j), \theta(\bar{u}(t, j)))]))^2. \quad (21)$$

We will consider the Euclidean metric $d_2(x, y) = \|x - y\|_2$ for the remainder of the manuscript. We will introduce two different model functions, f^{logit} and f^{Markov} , on which we will focus for the remainder of the paper. In particular, these two models will be numerically investigated in [Section 5](#).

3A1. Logistic regression. The model f^{logit} , introduced in the following, is a non-stationary and nonhomogeneous spatiotemporal extension of discrete choice models, which are standard techniques in the context of discrete data regression. This model class is a member in the generalized linear model (GLM) family [12; 10]. Discrete choice models can be derived from *utility theory* where the state of the regarded

process $\sigma(t, j, l)$ is assumed to be associated with a benefit or utility measure. In detail, this means that the process can be expressed as the function

$$\sigma(t, j, l) = \begin{cases} s_1 & \text{if } \mathcal{C}_1[u(t, j), B^1(t, j)] > \mathcal{C}_i[u(t, j), B^i(t, j)] \forall i \neq 1, \\ \vdots & \\ s_{N_S} & \text{if } \mathcal{C}_{N_S}[u(t, j), B^{N_S}(t, j)] > \mathcal{C}_i[u(t, j), B^i(t, j)] \forall i \neq N_S, \end{cases} \quad (22)$$

whereas

$$\mathcal{C}_i[u(t, j), B^i(t, j)] := \beta_0^i(t, j) + \sum_{e=1}^{N_E} \beta_e^i(t, j) u_e(t, j) + \xi^i(t, j) \quad (23)$$

is the utility measure dependent on unknown coefficients

$$B^i(t, j) = \begin{bmatrix} \beta_0^i(t, j) \\ \vdots \\ \beta_{N_E}^i(t, j) \end{bmatrix} \in \mathbb{R}^{(N_E+1) \times 1} \quad (24)$$

on observable (explicit) factors $u(t, j) \in \mathcal{U} \subset \mathbb{R}^{N_E \times 1}$ and on errors $\xi^i(t, j)$ characterizing the influences that could not be obtained through measurement (e.g., implicit external factors) [28; 29]. This implies that the probability for the dynamical process $\sigma(t, j, l)$ to be in state s_i can be expressed as follows:⁴

$$\begin{aligned} \mathbb{P}[\sigma(t, j, l) = s_i] &= \mathbb{P}[\mathcal{C}_i[u(t, j), B^i(t, j)] > \mathcal{C}_h[u(t, j), B^h(t, j)] \forall h \neq i] \quad (25) \\ &= \mathbb{P} \left[\beta_0^i(t, j) + \sum_{e=1}^{N_E} \beta_e^i(t, j) u_e(t, j) + \xi^i(t, j) \right. \\ &\quad \left. > \beta_0^h(t, j) + \sum_{e=1}^{N_E} \beta_e^h(t, j) u_e(t, j) + \xi^h(t, j) \forall h \neq i \right] \\ &= \mathbb{P} \left[\beta_0^i(t, j) - \beta_0^h(t, j) + \sum_{e=1}^{N_E} [\beta_e^i(t, j) - \beta_e^h(t, j)] u_e(t, j) + \xi^i(t, j) \right. \\ &\quad \left. > \xi^h(t, j) \forall h \neq i \right]. \end{aligned}$$

Various discrete choice models arise assuming different parametric forms of distributions for the random error terms $\xi^1(t, j), \dots, \xi^{N_S}(t, j)$. The logistic regression and the probit model are the most prominent examples of that model class; e.g., for logit models, the random part of the utility is assumed to be i.i.d. extreme value distributed (also known as Gumbel distribution), and for probit models, it is assumed to be multivariate normal. Results gained with either approach are similar, and

⁴Note that the probability of $\mathcal{C}_i[u(t, j), B^i(t, j)] = \mathcal{C}_h[u(t, j), B^h(t, j)]$ is assumed to be zero (see [29]).

significant differences are rare [26]. A multinomial logistic model, i.e., $N_S \geq 2$, is considered in the following. Consequently, the errors $\xi^1(t, j), \dots, \xi^{N_S}(t, j)$ are assumed to be i.i.d. with the Gumbel distribution resulting in the state probabilities

$$\mathbb{P}[\sigma(t, j, l) = s_i] = \frac{\exp\left(\beta_0^i(t, j) + \sum_{e=1}^{N_E} \beta_e^i(t, j)u_e(t, j)\right)}{\sum_{h=1}^{N_S} \exp\left(\beta_0^h(t, j) + \sum_{e=1}^{N_E} \beta_e^h(t, j)u_e(t, j)\right)} \quad \forall i. \quad (26)$$

The reader is referred to [29; 36] for a detailed probabilistic derivation. The corresponding model function f^{logit} with logistic regression parameter $B(t, j) = [B^1(t, j), \dots, B^{N_S}(t, j)] \in \mathbb{R}^{(N_E+1) \times N_S}$ is expressed as

$$\pi(t, j) := \theta^{\text{logit}}(B(t, j), u(t, j)) + \zeta(t, j), \quad (27)$$

where

$$\theta^{\text{logit}}(B(t, j), u(t, j)) = \begin{bmatrix} \mathbb{P}[\sigma(t, j, l) = s_1] \\ \vdots \\ \mathbb{P}[\sigma(t, j, l) = s_{N_S}] \end{bmatrix} \in \mathbb{R}^{N_S \times 1} \quad (28)$$

and $\zeta(t, j)$ is assumed to be an error process (e.g., please see the error of example model function given in (18)) related to the unknown implicit external influences and possible measurement errors. Note that there is no additional assumption concerning the probability distribution of $\zeta(t, j)$. The inverse problem corresponding to (27) with a model distance function g induced by the Euclidean metric has the form

$$\mathbf{L}(B(t, j)) = \sum_{t=1}^{N_T} \sum_{j=1}^{N_J} \|\pi(t, j) - \theta^{\text{logit}}(B(t, j), u(t, j))\|_2^2 \rightarrow \min_{B(t, j)}. \quad (29)$$

The standard logit model is one of the most used discrete choice models; nevertheless, it is important to check whether the problem setting of a certain considered application fits the model properties and whether it would be more reasonable to deploy a different discrete choice model. In this context, it is important to note that the logit model exhibits the *independence of irrelevant alternatives* (IIA) property [27], which states that for any two alternatives states s_i and s_h the ratio of the corresponding probabilities is

$$\exp\left(\beta_0^i(t, j) - \beta_0^h(t, j) + \sum_{e=1}^{N_E} (\beta_e^i(t, j) - \beta_e^h(t, j))u_e(t, j)\right). \quad (30)$$

In other words, the ratio does not depend on any state other than s_i and s_h and the relative odds remain the same [36]. Although this property might be realistic in some choice situations, it might be inappropriate in others [7]. Specifically, for

sets with similar states, i.e., states that are good substitutes of one another in the regarded system/application, the IIA property becomes implausible. This issue is often motivated with an example originating from a discussion McFadden offered in [29] on the subject: an individual takes one of the choices in the alternative set of states {auto, blue bus} with probability distribution [2/3 1/3], and then a red bus is added to the set of states, which causes the “intuitive” probability distribution, i.e., [2/3 1/6 1/6], to vary from the one implied by the IIA axiom [1/2 1/4 1/4].

The direct model function given in (27) can be extended in order to describe processes with memory, e.g., by including the previous (in time) and/or neighboring (in location space) values of the probability density $\pi(t, j)$ as the additional components of the external factors vector $u(t, j)$, e.g.,

$$u_{N_E+1}(t, j) := \pi_1(t-1, j). \quad (31)$$

Such logit models with Markov effects incorporated in the above form of external factors are known as *dynamical logit models* [32; 15]. One of the main drawbacks of the logistic regression ansatz is the internally embedded mapping (from the closed interval $[0, 1]$ to the continuum of real numbers $(-\infty, \infty)$) used to approach the discrete/categorical data with continuous regression techniques. This transformation causes computational instability on the boundaries of the logistic cumulative density function. Further, it is not possible to directly access the impact of the explicit external factors, which complicates the interpretations of the exterior influences.

Nevertheless, logistic regression is a good option for systems with nonlinear behavior. As a matter of fact, a nonlinear process can also be interpolated via a sequence of piecewise linear but nonstationary and nonhomogeneous local models. But in a case when the dynamics of the observed process are nonlinear as well as nonstationary and nonhomogeneous, it is more sensible to describe the system with an intrinsically nonlinear model (e.g., the nonstationary nonhomogeneous logistic regression).

3A2. Markov regression. As a locally linear alternative to the logistic regression model described above, we consider a nonstationary nonhomogeneous Markov regression. In order to incorporate all external factors in the model, we assume that the transition matrix $P(\bar{u}(t, j))$ corresponding to an observed Markovian dynamical process $\sigma(t, j, l)$ is continuously differentiable and has bounded second derivatives. Employing the results of Proposition 2.1, the following decomposition of the transition matrix is considered:

$$P(t, j, u(t, j)) = P_0(t, j) + \sum_{e=1}^{N_E} P_e(t, j) u_e(t, j). \quad (32)$$

The model function f^{Markov} is defined on the basis of an adapted stochastic master

equation (6):

$$\pi(t+1, j)^\top := \pi(t, j)^\top (P(t, j, u(t, j)) + \varepsilon(t, j)). \quad (33)$$

Then it is possible to formulate the following inverse problem:

$$\begin{aligned} & \mathbf{L}(P(t, j, u(t, j))) \\ &= \sum_{t=1}^{N_T} \sum_{j=1}^{N_J} \|\pi(t+1, j)^\top - \pi(t, j)^\top P(t, j, u(t, j))\|_2^2 \rightarrow \min_{P(t, j, u(t, j))}. \end{aligned} \quad (34)$$

3B. Interpolation. The optimization problem (20) exhibits several computational drawbacks such as ill-posedness (in the sense of Hadamard [13]) and therefore needs to undergo a series of changes in the form of regularizations. In the following, we make use of the fact that many real-life systems from various areas of application exhibit a certain level of persistence. Subsequently, it is possible to interpolate the model parameter function $\theta(\bar{u}(t, j))$ with a fixed number of N_K stationary and homogeneous model parameters $\theta_k(u(t, j))$ and corresponding affiliations $\gamma_k(t, j)$ with $k \in \{1, \dots, N_K\}$. This approach leads to a less ill-posed description of the considered dynamical system. Thus, assuming the existence of such local models $\Theta(u(t, j)) = [\theta_1(u(t, j)), \dots, \theta_{N_K}(u(t, j))]$ and weights $\Gamma(t, j) = [\gamma_1(t, j), \dots, \gamma_{N_K}(t, j)] \in [0, 1]^{1 \times N_K}$, the model distance functional first introduced in (19) can be phrased in the following interpolated formulation:

$$\begin{aligned} & g(\pi(t+1, j), \dots, \pi(t-N_M, j), \theta(\bar{u}(t, j))) \\ &= \sum_{k=1}^{N_K} \gamma_k(t, j) g(\pi(t+1, j), \dots, \pi(t-N_M, j), \theta_k(u(t, j))). \end{aligned} \quad (35)$$

The affiliation process $\Gamma(t, j)$ characterizes the regime behavior and the nonstationary and nonhomogeneous nature of the dynamical system. The weights $\gamma_k(t, j)$ have the specification to take positive values and sum up to one over all N_K local models, i.e.,

$$\sum_{k=1}^{N_K} \gamma_k(t, j) = 1 \quad \forall j \in \{1, \dots, N_J\}, t \in \{1, \dots, N_T\}, \quad (36)$$

$$\gamma_k(t, j) \geq 0 \quad \forall j \in \{1, \dots, N_J\}, t \in \{1, \dots, N_T\}, k \in \{1, \dots, N_K\}. \quad (37)$$

Then the corresponding inverse problem can formally be expressed by

$$\mathbf{L}(\Gamma(t, j), \Theta(u(t, j))) = \sum_{j=1}^{N_J} \mathbf{L}_j(\Gamma(:, j), \Theta(u(t, j))) \rightarrow \min_{\Gamma(t, j), \Theta(u(t, j))} \quad (38)$$

with

$$\begin{aligned} & \mathbf{L}_j(\Gamma(\cdot, j), \Theta(u(t, j))) \\ &= \sum_{t=1}^{N_T} \sum_{k=1}^{N_K} \gamma_k(t, j) g(\pi(t+1, j), \dots, \pi(t-N_M, j), \theta_k(u(t, j))). \end{aligned} \quad (39)$$

Note that the constraints are independent for each location j . This independence in space and the structure of the functional \mathbf{L} will be exploited in the numerical optimization of (38) with respect to $\Gamma(t, j)$. The main idea is that every location j can be regarded separately due to the fact that the overall functional \mathbf{L} is a sum of local (uncoupled in $\Gamma(\cdot, j)$) functionals \mathbf{L}_j with (uncoupled in $\Gamma(\cdot, j)$) constraints (36) and (37). A corresponding numerical algorithm exploiting this structure of the problem will be discussed in detail in Section 3D. The influence of the implicit external factors $u^{\text{unres}}(t, j)$ is reflected in the explicit time- and space-dependence of the affiliation process $\Gamma(t, j)$.

In case of the logistic regression, this regularization means that we need to find a set of locally stationary and homogeneous (i.e., not dependent on time t and location j) model parameters $\{B_1, \dots, B_{N_K}\}$ with $B_k = [B_k^1, \dots, B_k^{N_S}] \in \mathbb{R}^{(N_E+1) \times N_S} \forall k \in \{1, \dots, N_K\}$. For the Markov regression, the interpolated version of (34) is

$$\begin{aligned} & \mathbf{L}(\Gamma(t, j), P(u(t, j))) \\ &= \sum_{j=1}^{N_J} \sum_{t=1}^{N_T} \sum_{k=1}^{N_K} \gamma_k(t, j) \|\pi(t+1, j)^\top - \pi(t, j)^\top P^k(u(t, j))\|_2^2 \rightarrow \min_{\Gamma(t, j), P(u(t, j))}, \end{aligned} \quad (40)$$

where the local Markovian transition operators $P(u(t, j)) = [P^1(u(t, j)), \dots, P^{N_K}(u(t, j))] \in \mathbb{R}^{N_S \times N_S N_K}$ for fixed t and j are defined in a linear approximation:

$$P^k(u(t, j)) = P_0^k + \sum_{e=1}^{N_E} P_e^k u_e(t, j) \quad \forall k \in \{1, \dots, N_K\}. \quad (41)$$

To ensure that the stochasticity of the Markov transition operator remains preserved, the optimization problem is subject to a number of constraints. Since the transition matrices $P^k(u(t, j))$ are stochastic matrices, the matrices P_e^k are required to satisfy the equalities

$$P_0^k \mathbf{1} = \mathbf{1} \quad \forall k \in \{1, \dots, N_K\}, \quad (42)$$

$$P_e^k \mathbf{1} = \mathbf{0} \quad \forall e \in \{1, \dots, N_E\}, k \in \{1, \dots, N_K\}, \quad (43)$$

whereas $\mathbf{1} \in \mathbb{R}^{N_S \times 1}$ is a column vector with all entries equal to one and analogously $\mathbf{0} \in \mathbb{R}^{N_S \times 1}$ refers to the corresponding vector with all entries equal to zero. Furthermore, the entries of $P^k(u(t, j))$ need to be greater than or equal to zero. In the case of a rectangular domain \mathcal{Q} , the feasible number of 2^{N_E} inequality constraints

(consisting of all possible combinations of suprema and infima of the N_E explicit external factors $u_e(t, j)$)

$$\{P_0^k\}_{n,m} + \sum_{e=1}^{N_E} \{P_e^k\}_{n,m} \left[\begin{array}{c} \sup_{t,j} u_e(t, j) \\ \inf_{t,j} u_e(t, j) \end{array} \right] \geq 0 \quad \forall k, n, m \quad (44)$$

is sufficient to satisfy this condition. See [30] for more details and a proof of (44) for the purely temporal case; extension to the spatiotemporal case given in equations (41)–(44) above is straightforward.

3C. Spatial and temporal persistence. The problem formulation is still ill-posed since its solution may not be unique due to many possibilities to choose the switching process Γ . Therefore, we need to make further assumptions/restrictions on the function space that contains the switching process and add another constraint to the optimization problem. More precisely, to approach this issue, we limit the number of transitions of $\gamma_k(\cdot, j)$, introducing a persistency constraint on the time interval

$$|\gamma_k(\cdot, j)|_{\text{BV}(1, N_T)} = \sum_{t=1}^{N_T-1} |\gamma_k(t+1, j) - \gamma_k(t, j)| \leq N_C \quad (45)$$

that holds for every location $j \in \{1, \dots, N_J\}$. Without an additional spatial regularization, the constraints for parameter $\Gamma(t, j)$ are still independent for every location. This structural advantage allows us to compute each $\Gamma(\cdot, j)$ separately if the value of the parameter $\Theta(u(t, j))$ is kept fixed. In some situations, it might be reasonable to limit the variation along the locations as well (e.g., a limitation concerning only the neighboring cells of a location), but constraints on the switching process Γ would result in a global coupling (in j) for different optimization problems L_j from (39), leading to immense numerical costs. Furthermore, an identification of the best model in terms of parameter choice, discussed in the next paragraph, would have to be pursued for all possible combinations of choices for $N_C(j)$, $j \in \{1, \dots, N_J\}$, as well, leading to a computationally expensive analysis. This additional regularization over spatial locations is an aspect of further research.

3D. Numerical approach and computational complexity. The inverse problem posed in (38) has no general analytic solution and is not convex (i.e., it is not possible to obtain a unique global minimum with standard approaches, e.g., gradient descent or Newton methods). But the global optimizers $\Gamma^*(t, j)$ and $\Theta^*(u(t, j))$ can be approximated combining a *subspace algorithm* and simulated annealing [22]. The main idea of the subspace algorithm is to exploit the above-mentioned structural property of the optimization problem (38), i.e., that the simple convex optimization problems can be stated for Γ and Θ separately, i.e., for (i) an optimization with respect to $\Gamma(\cdot, j)$ for a fixed Θ and (ii) an optimization with respect to Θ for

a fixed Γ . Dividing the optimization problem with two sets of unknowns into two minimizations over just one set of parameters reduces the originally high-dimensional and nonconvex problem to two manageable problems that can be approached with standard optimization techniques, e.g., simplex method for the above subspace step (i) and quadratic minimization with linear equality and inequality constraints for the above subspace step (ii). It is straightforward to demonstrate that the subsequent repetition of steps (i) and (ii) leads to a strict minimization of the original functional \mathbf{L} , and since the average model distance functional is bounded with zero from below, this procedure will converge to a local minimum of \mathbf{L} . Iterations over the subproblems only converge to local minima, and simulated annealing approaches [22; 25] can be deployed in combination with the subspace-iteration algorithm to avoid getting trapped in the local minimum. The details of the algorithm are now given in the pseudocode in [Algorithm 1](#).

In contrast to the time-dependent algorithm introduced in [30], the additional spatial dimension j is involved in the above scheme. Since a spatial regularization is not included, the affiliations $\Gamma(\cdot, j)$ are determined for each location j separately (see the for-loop on line 6), i.e., the problem of optimizing \mathbf{L} with respect to Γ is equivalent to separate optimization of N_J suboptimization problems given by functionals \mathbf{L}_j defined in (39). The local stationary and homogeneous model parameters θ_i , on the other hand, are computed for all t and j simultaneously (line 9). A separate computation for every spatial component is not possible here since different \mathbf{L}_j are coupled through Θ .

In order to obtain a global minimizer of (38), the subspace-iteration algorithm is repeated $N_{\text{anneal}}^{\text{FEM}}$ times with different randomly sampled initial parameters $\Gamma^{[0]}$ (see lines 2 and 3). This form of simulated annealing helps to avoid local minima by trying to consider the entire parameter space. Since the annealing steps can be run independently, it is possible to reduce the corresponding computational complexity via an “embarrassingly parallel” implementation. The necessary memory capacity as well as the computing time can be further decreased by using a time-discretized (with finite elements) version of the full process Γ [30]. This form of dimension reduction is especially beneficial when modeling time-persistent dynamical systems with few transitions between the local models (i.e., systems where a comparatively small number of finite element functions ($N_{\text{basis}}^{\text{FEM}} \ll N_T$) is sufficient for qualitative results).

Computational cost of the proposed technique is dependent on the number of locations N_J and the number $N_{\text{basis}}^{\text{FEM}}$ of finite elements for the time discretization. The run time for the Γ calculation is proportional to $\mathcal{O}(N_J N_K (2N_{\text{basis}}^{\text{FEM}} - 1)^\kappa)$, where $\kappa \geq 1$ is the parameter dependent on the choice of the numerical scheme for the $\Gamma(\cdot, j)$ -optimization (Step 1 of [Algorithm 1](#)). As already indicated above, the computational complexity of the determination of Θ varies for different model classes and the spatial component can be regarded as an additional dimension in the

input : Set number of different regimes N_K , value for time-wise transition boundary N_C , number of simulated annealing steps $N_{\text{anneal}}^{\text{FEM}}$ and optimization tolerance value τol (optional: number of finite-element functions $N_{\text{basis}}^{\text{FEM}}$).

output : Global optimizers $\Gamma^*(t, j)$ and $\Theta^*(u(t, j))$

- 1 $\mathbf{L}_{\min} = 1000000$
- 2 **for** $r = 1 : N_{\text{anneal}}$ **do**
- 3 Generate random initial $\Gamma_r^{[0]}$ and compute $\Theta_r^{[0]}$.
- 4 **while** $|\mathbf{L}(\Gamma_r^{[s]}, \Theta_r^{[s]}) - \mathbf{L}(\Gamma_r^{[s-1]}, \Theta_r^{[s-1]})| \geq \tau ol$ **do**
- 5 Step 1:
- 6 **for** $j = 1 : N_J$ **do**
- 7 Determine $\Gamma_r^{[s+1]}(:, j) = \arg \min \mathbf{L}_j(\Gamma(:, j), \Theta_r^{[s]})$ subject to constraints (36), (37) and (45), whereas $\Theta_r^{[s-1]}$ denotes the current fixed approximation of the optimal Θ^* . Apply standard methods of linear minimization with linear equality and inequality constraints (e.g., simplex method).
- 8 Step 2:
- 9 Compute $\Theta_r^{[s+1]} = \arg \min \mathbf{L}(\Gamma_r^{[s+1]}, \Theta)$ (additional constraints depend on the model, e.g., constraints (42)–(44) in case of the Markovian process and no constraints in the logistic regression case). Apply standard methods of quadratic optimization with linear equality and inequality constraints.
- 10 $s := s + 1$.
- 11 **if** $\mathbf{L}_{\min} \geq \mathbf{L}(\Gamma_r^*(t, j), \Theta_r^*(u(t, j)))$ **then**
- 12 $\mathbf{L}_{\min} = \mathbf{L}(\Gamma_r^*(t, j), \Theta_r^*(u(t, j)))$
- 13 $\Gamma^* = \Gamma_r^*$
- 14 $\Theta^* = \Theta_r^*$
- 15 **Return** Γ^* and Θ^* .

Algorithm 1: Subspace algorithm with annealing steps.

problem. In Step 2 of the algorithm, one needs to solve a quadratic minimization problem subject to linear constraints (equalities and inequalities) to compute the matrices P_e^k considering the nonstationary nonhomogeneous Markov regression (see (40)). Such problems are known to be NP-complete [37]. For the logistic model (see (27)), the computational complexity of the Step 2 can be expressed as $\mathcal{O}(N_K N_T N_J)$ [31]. The overall resulting numerical cost of the proposed method is in the range of the average complexity of standard approaches such as artificial neural networks ($\mathcal{O}((N_{\text{weights}}^{\text{ANN}})^3)$ where $N_{\text{weights}}^{\text{ANN}}$ is the number of ANN parameters, i.e., neural biases and weights⁵) and support vector machines ($\mathcal{O}(N_T^2 N_E)$ with N_E

⁵This number is directly proportional to the number of neurons and depends on the type of the transfer functions and network architecture.

referring to the number of explicit external factors). Details of the techniques and their computational time complexity will be discussed in [Section 5](#).

3E. Information criterion. A further issue originates from the selection of the parameters N_K and N_C , which can lead to a variety of models differing in terms of quality and complexity. This problem is addressed by applying a *modified* formulation of *Akaike's information criterion* (mAIC). The main idea of the method is based on approximating the time series of the obtained model errors $g(\pi(t+1, j), \dots, \pi(t-N_M, j), \theta_k(u(t, j)))$ through an optimal nonparametric scalar-valued stochastic process, followed by the comparison of the mAIC values for the obtained processes from different models. A detailed description of the method can be found in [\[30\]](#). The main advantage of this approach is that no a priori parametric probabilistic assumptions about the analyzed data are necessary.⁶

The main idea of an information criterion is that the quality of the determined model is weighted against the total number of parameters involved in the calculation of the model [\[1\]](#). In other words, the aim is to identify the model that fits best with the fewest number of necessary model parameters, e.g.,

$$\text{mAIC}(N_K, N_C) = -2 \log(\mathcal{L}(N_K, N_C)) + 2|M(N_K, N_C)|. \quad (46)$$

Here the likelihood $\mathcal{L}(N_K, N_C)$ corresponds to the underlying model characterized by N_K different regimes with a maximum of N_C transitions between them and is defined as

$$\begin{aligned} & \mathcal{L}(N_K, N_C) \\ &= \prod_{j=1}^{N_J} \prod_{t=1}^{N_T} \prod_{k=1}^{N_K} \sum \gamma_k(t, j) \phi_k(g(\pi(t+1, j), \dots, \pi(t-N_M, j), \theta_k(u(t, j))) | N_{\phi_k}). \end{aligned} \quad (47)$$

A detailed derivation of the likelihood function for the nonstationary case can also be found in [\[30\]](#). The expression above is its straightforward extension to the nonstationary and nonhomogeneous case. The functional $M(N_K, N_C)$ describes the total number of involved model parameters, which in the case of the logistic regression consists of

$$|M^{\text{logit}}(N_K, N_C)| = |\Gamma| + N_K(N_E + 1) \quad (48)$$

and for the Markov regression is

$$|M^{\text{Markov}}(N_K, N_C)| = |\Gamma| + N_K N_S (N_S - 1) (N_E + 1). \quad (49)$$

This modified version of Akaike's information criterion coupled with the nonstationary and nonhomogeneous logistic and Markov regression (introduced above)

⁶Such parametric a priori assumptions are needed to compute the log-likelihood of the data in context of standard information criteria like AIC.

allows us to simultaneously identify the optimal model and the optimal values of the parameters N_K and N_C .

In practice, mAIC values for different cluster values $N_K \in \mathcal{S}_1$ and persistency parameter values $N_C \in \mathcal{S}_2$ might not vary substantially. By appointing only one model, other suitable ones are discarded, resulting in an unnecessary information loss [5]. In this case, the mAIC values of the possible models are ranked via the deviation from the lowest mAIC value, i.e.,

$$\Delta(N_K, N_C) = \exp \left[\frac{\min_{(N'_K, N'_C) \in \mathcal{S}_1 \times \mathcal{S}_2} (\text{mAIC}(N'_K, N'_C)) - \text{mAIC}(N_K, N_C)}{2} \right]. \quad (50)$$

If there is more than one probable model, then the overall model can be considered as a multimodel, i.e., a weighted linear combination of individual models with the model weights [5] given by

$$w(N_K, N_C) := \frac{\Delta(N_K, N_C)}{\sum_{(N'_K, N'_C) \in \mathcal{S}_1 \times \mathcal{S}_2} \Delta(N'_K, N'_C)}. \quad (51)$$

Besides determining the optimal model with respect to the parameters N_K and N_C , the criterion can also be used to determine the better model in terms of the prior assumptions. Since different models are compared with respect to the same observation data and the same form of the nonparametric likelihood-estimation procedure described in [30] (based on fitting the optimal stochastic process to the time series of the model residuals), resulting mAIC values can be used to identify the statistically optimal model from a given class of models (e.g., Markov, logit, ANN and SVM). Practical examples of this data-based model-discrimination procedure will be given in the last sections of this manuscript.

4. Prediction and assimilation of additional information

Suppose the global optimal model parameters $\Gamma^*(t, j)$ and $\Theta^*(u(t, j))$ with respect to the average model distance functional $\mathbf{L}(\Gamma(t, j), \Theta(u(t, j)))$ introduced in (38) can be determined with the proposed numerical scheme (see Algorithm 1); then it is possible to approximate the observed time series

$$\pi(t+1, j) \approx f \left(\pi(t, j), \dots, \pi(t - N_M, j), \sum_{k=1}^{N_K} \gamma_k^*(t, j) \theta_k^*(u(t, j)) \right) \quad (52)$$

on the basis of the formal definition of the direct model function. This ansatz, used to approximate the vector of state probabilities, is discussed in detail in [30] and allows

us to directly concatenate the two model parameters $\Gamma^*(t, j)$ and $\Theta^*(u(t, j))$.⁷ In most of the practical applications, a further aspect of interest is a prediction $\hat{\pi}(N_T + N_{\text{pred}}, j)$ of the probability distribution $\pi(N_T + N_{\text{pred}}, j)$ outside of the observed time sequence $\{1, \dots, N_T\}$. The quantity N_{pred} denotes the prediction depth, i.e., the total number of prediction steps in time. The difficulty lies in the nonstationarity and nonhomogeneity of the model formulation; i.e., any prediction crucially depends on $\Gamma^*(t, j)$, which is only defined for the observed time sequence $\{1, \dots, N_T\}$. In order to predict future affiliations $\hat{\Gamma}(N_T + N_{\text{pred}}, j)$, the process $\Gamma^*(t, j)$ can be regarded as an observed time series of probabilities to be in N_K different discrete states. Subsequently, the proposed Markov regression framework (given in (40)) can be applied to determine the model parameters describing $\Gamma^*(t, j)$. To avoid an infinite sequence of prediction problems caused by nonstationarity and nonhomogeneity, the model of the affiliation process $\Gamma^*(t, j)$ is assumed to have only one regime (i.e., $N_K = 1$). Although this is a strong restriction, it is important to note that stationarity as well as homogeneity are common assumptions in time-series analysis. This self-contained strategy to determine

$$\hat{\Gamma}(N_T + N_{\text{pred}}, j) = \Gamma^*(N_T, j) \prod_{\tau=0}^{N_{\text{pred}}-1} \left(\left[P_0^\Gamma + \sum_{e=1}^{N_E} P_e^\Gamma u_e(N_T + \tau, j) \right] \right) \quad (53)$$

has been introduced in [16] (in the context of purely time-dependent data) and further discussed and deployed in [30]. The model transition matrix, characterizing the dynamics of the affiliation $\Gamma^*(t, j)$, is denoted $P^\Gamma(u(t, j))$ and is a linear combination of explicit external factors $u(t, j)$ and matrices $P_0^\Gamma, \dots, P_{N_E}^\Gamma$ (see (41) for $N_K = 1$). In a case when the data $\pi(N_T + 1, j)$ for the next time step can be obtained, the new information can be used to update the $\hat{\Gamma}(t, j)$ -predictor. A strategy for updating the prediction $\hat{\Gamma}(N_T + 1, j)$ conditioned on the additional information $\pi(N_T + 1, j)$ has recently been introduced in [16; 30] and is based on the maximum-likelihood principle, i.e.,

$$\begin{aligned} \gamma_k^*(N_T + 1, j) &= \begin{cases} 1 & \text{if } k = \arg \min_h g(\pi(t + 1, j), \dots, \pi(t - N_M, j), \theta_h(u(t, j))), \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \quad (54)$$

The update $\gamma_k^*(N_T + 1, j)$ is assumed to be optimal (hence the superscript $*$). In detail, this means that it is possible to identify all local regimes θ_k describing the dynamical process $\sigma(t, j, l)$ on the basis of the available data measured in the time sequence $\{1, \dots, N_T\}$. Further, it is necessary to assume the affiliation process Γ^*

⁷Note that the model function f needs to be linear in its parameters and the model distance functional g has to be strictly convex to pursue the equation given in (52) (for a detailed derivation see [30]). This is the case for the proposed Markov as well as for the logistic model.

is deterministic (i.e., takes only values in the set $\{0, 1\}$). In the following, a new update method is proposed on the basis of Bayes' theorem that allows for a fuzzy affiliation. We denote $\Gamma(N_T + 1, j)$ to be the true but unknown cluster affiliation of the dynamical system under observation and $\widehat{\Gamma}(N_T + 1, j)$ the (prior) prediction, calculated only with the information from the previous time steps $t \in \{1, \dots, N_T\}$, and as $\dot{\Gamma}(N_T + 1, j)$, we denote the posterior estimate based on the new observation $\pi(N_T + 1, j)$. The following proposition gives an analytical form of the posterior estimate of the hidden model affiliation function and shows how the implicit impact of the unresolved external factors can be assimilated into the model:

Proposition 4.1. *Let the entries of $\gamma_k(t, j)$ for all j, t and k only assume values zero or one and the predictor $\widehat{\Gamma}(N_T + 1, j)$ be a prior probability distribution for $\Gamma(N_T + 1, j)$ in the sense that*

$$\mathbb{P}[\gamma_k(N_T + 1, j) = 1] = \hat{\gamma}_k(N_T + 1, j). \quad (55)$$

Moreover, let the distribution of the observation $\pi(N_T + 1, j)$ given the information about the affiliation γ_k at time t be independent of the prediction $\widehat{\Gamma}(t, j)$. Then the posterior distribution of the regime assigning process $\Gamma(N_T + 1, j)$ is of the form

$$\dot{\gamma}_k(N_T + 1, j) = \frac{\mathbb{P}[\pi(N_T + 1, j) | \gamma_k(N_T + 1, j) = 1] \hat{\gamma}_k(N_T + 1, j)}{\sum_{h=1}^{N_K} \mathbb{P}[\pi(N_T + 1, j) | \gamma_h(N_T + 1, j) = 1] \hat{\gamma}_h(N_T + 1, j)}. \quad (56)$$

Proof. Using the above assumptions and Bayes' theorem, the following holds:

$$\begin{aligned} & \mathbb{P}[\gamma_k(N_T + 1, j) = 1 | \pi(N_T + 1, j)] \\ &= \frac{\mathbb{P}[\gamma_k(N_T + 1, j) = 1; \pi(N_T + 1, j)]}{\mathbb{P}[\pi(N_T + 1, j)]} \\ &= \frac{\mathbb{P}[\pi(N_T + 1, j) | \gamma_k(N_T + 1, j) = 1] \mathbb{P}[\gamma_k(N_T + 1, j) = 1]}{\mathbb{P}[\pi(N_T + 1, j)]} \\ &= \frac{\mathbb{P}[\pi(N_T + 1, j) | \gamma_k(N_T + 1, j) = 1] \mathbb{P}[\gamma_k(N_T + 1, j) = 1]}{\sum_{h=1}^{N_K} \mathbb{P}[\pi(N_T + 1, j) | \gamma_h(N_T + 1, j) = 1] \mathbb{P}[\gamma_h(N_T + 1, j) = 1]}. \quad \square \end{aligned}$$

As will be demonstrated by numerical examples in the next section, formula (56) improves an estimation of the new affiliations in comparison to the maximum-likelihood approach (54) deployed in [16; 30]. Although the affiliation is “fuzzy” (i.e., resulting affiliations may take values between zero and one), it is (as demonstrated by the numerical tests) less prone with respect to introducing unjustified switches between the local models.

5. Numerical investigation

To explore the characteristic properties of the introduced nonstationary and nonhomogeneous regression framework, we apply it to three different synthetic data sets. Note that we actively chose to work with artificial rather than real-life examples due to the specific settings necessary to analyze the proposed framework. In a real-life observation, for example, there is no reliable information about the influencing factors $u^{\text{unres}}(t, j)$ that are not available in form of measurements.

Different model functions (e.g., Markov and logit) for the framework proposed in [Section 3](#) as well as other standard techniques of time-series analysis (e.g., SVM and ANN) are considered in the following. It is necessary to distinguish between the different resulting model parameters via additional superscript tags (e.g., $\Gamma^{\text{Markov}}(t, j)$ or $\Gamma^{\text{logit}}(t, j)$). The same labeling system is employed for approximations of the actual observations $\pi(t, j)$ determined with model parameters computed with various methodologies (e.g., $\pi^{\text{Markov}}(t, j)$ or $\pi^{\text{logit}}(t, j)$ or $\pi^{\text{ANN}}(t, j)$). Due to the fact that the considered observations are artificial, all parameters and variables used to generate the synthetic data are tagged with the superscript *syn* (e.g., $\Gamma^{\text{logit}}(t, j)$). Some tags are specifying the settings used for a specific algorithm such as the number of annealing steps (e.g., $N_{\text{anneal}}^{\text{ANN}}$ or $N_{\text{anneal}}^{\text{FEM}}$) or the regularization factor (e.g., N_C^{FEM} or N_C^{SVM}). Note that the regularization factor N_C can have superscripts FEM as well as Markov or logit although all of those labels correspond to the technique proposed in [Section 3](#). This further distinction is necessary as the abbreviation FEM is a general reference to the framework introduced in the current manuscript. Resulting parameters derived with any technique, which are considered to be optimal in the sense that the corresponding model has the lowest AIC, have a superscript $*$.

A few variables remain free of labels as they are independent of the parameter-identification process and are assumed to be the same for all the employed techniques, e.g., the number of explicit external factors N_E , the number of discrete states N_S or the number of considered locations N_J .

One aspect of the numerical investigation includes testing of the various considered parameter-identification techniques with respect to predictions, i.e., approximate data that was not given for the computation of corresponding model parameters. Thus, it is necessary to divide the time sequence $\{1, \dots, N_T\}$ describing the time-wise length of the observations $\pi(t, j)$ into two components $\{1, \dots, N_{T_{\text{train}}}\}$ and $\{N_{T_{\text{train}}} + 1, \dots, N_T\}$. The first sequence will be referred to as *training sequence* and the second one will be known as *test sequence*.

The first data set is discussed in [Section 5A](#) and is employed to demonstrate the general feasibility of the proposed nonstationary and nonhomogeneous Markov regression as well as the logistic regression frameworks under “good conditions” (all relevant data is given for the computations, i.e., no unresolved external factors

$u^{\text{unres}}(t, j)$). In [Section 5B](#), the focus is on the key attribute of the Markov regression technique presented in this paper, which allows us to take missing/unavailable external factors into account. To numerically investigate this theoretical incorporation of unobserved information, a synthetic data set is generated with $N_F = 101$ external factors $\bar{u}(t, j)$ and only one of these 101 factors is made available for the calculation of the model parameters (i.e., $N_E = 1$ and $N_I = 100$).

The last example data set is chosen to numerically investigate (again considering the nonstationary and nonhomogeneous Markov regression) the newly proposed update of the prediction $\widehat{\Gamma}(N_T + 1, j)$ (see [Proposition 4.1](#)). The quality of the determined model is analyzed and compared to the results of two standard frameworks from machine learning (namely artificial neural networks [[2](#); [24](#); [18](#); [3](#)] and support vector machines [[8](#); [35](#)]).

5A. Nonstationary example. Under ideal conditions, the regarded dynamical process $\sigma(t, j, l)$ has the Markov property and all external influences are available in the form of observation data. The toy example considered in this section allows us to check the basic feasibility of the proposed technique and also serves as a reference for an example under “bad conditions”, investigated in [Section 5B](#). The data is generated using the proposed Markov model structure (see [\(32\)](#)) and pseudorandom numbers generated by the computer. In the following, two algorithms are outlined in order to explain the synthetic data. At first, the affiliation process $\Gamma^{\text{syn}}(t, j)$ subject to constraints [\(45\)](#), [\(36\)](#) and [\(37\)](#) is generated.

The synthetic parameter Γ^{syn} is generated with pseudorandom numbers that, for simplicity, are restricted to the set $\{0, 1\}$. Furthermore, a certain level of persistency is forced on $\Gamma^{\text{syn}}(t, j)$, meaning that the total number of transitions is limited to N_C^{syn} (see lines 3–12 of [Algorithm 2](#)). As the weights $\gamma_k^{\text{syn}}(t, j)$, generated with [Algorithm 2](#), only take values in $\{0, 1\}$, it is possible to directly assume⁸

$$P^{\text{syn}}(t, j, u(t, j)) \approx \sum_{k=1}^{N_C^{\text{syn}}} \gamma_k^{\text{syn}}(t, j) P^{k \text{ syn}}(u(t, j)), \quad (57)$$

whereas the definition of $P^{k \text{ syn}}(u(t, j))$ is given in [\(41\)](#). Then a synthetic time series $\pi^{\text{syn}}(t, j)$ can be computed on the basis of the definition of the ensemble data by generating an ensemble of N_{ens} Markov chain realizations $\sigma^{\text{syn}}(t, j, l) \in \{s_1, \dots, s_{N_S}\}$ given the transition matrix $P^{\text{syn}}(t, j, u(t, j))$ (see [Algorithm 3](#)). The transition matrix $P^{\text{syn}}(t, j, u(t, j))$ is calculated using the assumed model structure given in [\(41\)](#) and [\(57\)](#) (see line 5). Further, it is assumed that $P^{\text{syn}}(t, j, u(t, j))$ also depends linearly on the implicit external factors $u^{\text{unres}}(t, j)$, given for the

⁸For more information on this approximation of the transition matrix $P^{\text{syn}}(t, j, u(t, j))$, see elucidations in [Section 4](#), or for a more detailed discussion on the matter (for purely time-dependent model parameters), the reader is referred to [\[30\]](#).

```

input : Choose values for  $N_K^{\text{syn}}, N_C^{\text{syn}}, N_T$  and  $N_J$ .
output :  $\Gamma^{\text{syn}}(t, j)$ 
1 for  $j = 1 : N_J$  do
2    $\gamma_k^{\text{syn}}(:, j) = [] \forall k \in \{1, \dots, N_K\}$ 
3   for  $c = 1 : N_C^{\text{syn}}$  do
4      $N_{\text{dummy}} = \text{round}(2N_T / (N_C^{\text{syn}} * \text{rand}([0, 1])))$ 
5      $\text{dummy0} = (0, \dots, 0) \in \mathbb{R}^{1 \times N_{\text{dummy}}}$ 
6      $\text{dummy1} = (1, \dots, 1) \in \mathbb{R}^{1 \times N_{\text{dummy}}}$ 
7      $r = \text{rand}(\{1, \dots, N_K^{\text{syn}}\})$ 
8     for  $k = 1 : N_K^{\text{syn}}$  do
9       if  $r == k$  then
10         $\gamma_k^{\text{syn}}(:, j) = [\gamma_k^{\text{syn}}(:, j) \text{ dummy1}]$ 
11       else
12         $\gamma_k^{\text{syn}}(:, j) = [\gamma_k^{\text{syn}}(:, j) \text{ dummy0}]$ 
13     if  $\text{length}(\gamma_1^{\text{syn}}(:, j)) \geq N_T$  then
14        $\gamma_k^{\text{syn}}(:, j) = \gamma_k^{\text{syn}}(1 : N_T, j) \forall k \in \{1, \dots, N_K^{\text{syn}}\}$ 
15     else
16        $N_{\text{dummy}} = N_T - \text{length}(\gamma_1^{\text{syn}}(:, j))$ 
17        $\text{dummy0} = (0, \dots, 0) \in \mathbb{R}^{1 \times N_{\text{dummy}}}$ 
18        $\text{dummy1} = (1, \dots, 1) \in \mathbb{R}^{1 \times N_{\text{dummy}}}$ 
19        $\gamma_1^{\text{syn}}(:, j) = [\gamma_1^{\text{syn}}(:, j) \text{ dummy1}]$ 
20        $\gamma_k^{\text{syn}}(:, j) = [\gamma_k^{\text{syn}}(:, j) \text{ dummy0}] \forall k \in \{2, \dots, N_K^{\text{syn}}\}$ 
21    $\Gamma^{\text{syn}}(:, j) = [\gamma_1^{\text{syn}}(:, j), \dots, \gamma_{N_K^{\text{syn}}}^{\text{syn}}(:, j)]$ 

```

Algorithm 2: Generate synthetic affiliation $\Gamma^{\text{syn}}(t, j)$.

generation of artificial data. Hence, analogously to the synthetic model matrices $P_1^{k \text{ syn}}, \dots, P_{N_E}^{k \text{ syn}}$, corresponding to the explicit external factors $u(t, j)$, a set of matrices $P_{N_E+1}^{k \text{ syn}}, \dots, P_{N_E+N_I}^{k \text{ syn}}$, related to the unresolved factors $u^{\text{unres}}(t, j)$, is chosen for $k \in \{1, \dots, N_K^{\text{syn}}\}$.

In order to generate samples from a distribution, as necessary in lines 7–9 of [Algorithm 3](#), one can employ standard techniques such as rejection sampling (also known as the acceptance-rejection method) [6; 33; 38]. Finally, the artificial data $\pi^{\text{syn}}(t, j)$ can be computed considering the quotients $N_{s_i}(t, j)/N_{\text{ens}}$ first introduced in (1), which are assumed to be a good approximation of the probability $\pi^{\text{syn}}(t, j)$ for large N_{ens} . The affiliation $\gamma_k^{\text{syn}}(t, j)$ is generated with the following setting: $N_C^{\text{syn}} = 10$, $N_K^{\text{syn}} = 2$, $N_T = 400$, $N_J = 24$, $N_S = 2$, $N_E = 2$ and $N_I = 0$. The first explicit external influence $u_1(t, j)$ is set to be a time- and location-dependent sinus function. As the second factor, we use the average of the neighboring cell

input : Choose values for N_K^{syn} , $\Gamma^{\text{syn}}(t, j)$ for all t and j (already generated), N_T , N_J , N_E , N_I , N_S and N_{ens} . Define synthetic model matrices $P_0^{k \text{ syn}}, \dots, P_{N_E}^{k \text{ syn}}, P_{N_E+1}^{k \text{ syn}}, \dots, P_{N_E+N_I}^{k \text{ syn}}$ with $k \in \{1, \dots, N_K^{\text{syn}}\}$, a finite set of discrete states $\{s_1, \dots, s_{N_S}\}$ and explicit as well as implicit external factors, i.e., $u(t, j)$ and $u^{\text{unres}}(t, j)$.

output: $\pi^{\text{syn}}(t, j)$

```

1 Initialize  $\sigma^{\text{syn}}(0, j, l) = \mathbf{rand}\{s_1, \dots, s_{N_S}\} \forall j \in \{1, \dots, N_J\}, l \in \{1, \dots, N_{\text{ens}}\}$ 
2 for  $t = 1 : N_T$  do
3   for  $j = 1 : N_J$  do
4      $P^{\text{syn}}(t, j, u(t, j)) =$ 
        $\sum_{k=1}^{N_K} \gamma_k(t, j) (P_0^{k \text{ syn}} + \sum_{e=1}^{N_E} P_e^{k \text{ syn}} u_e(t, j) + \sum_{e=1}^{N_I} P_{N_E+e}^{k \text{ syn}} u_e^{\text{unres}}(t, j))$ 
5     for  $l = 1 : N_{\text{ens}}$  do
6        $h = \mathbf{index}(\sigma^{\text{syn}}(t-1, j, l))$ 
7        $\sigma^{\text{syn}}(t, j, l) = \begin{cases} s_1 & \text{with probability } \{P^{\text{syn}}(t, j, u(t, j))\}_{h1}, \\ \vdots \\ s_{N_S} & \text{with probability } \{P^{\text{syn}}(t, j, u(t, j))\}_{hN_S} \end{cases}$ 
8       (see rejection sampling [6; 33; 38])
9     for  $i = 1 : N_S$  do
10       $\pi_i^{\text{syn}}(t, j) = \mathbf{counter}(\sigma^{\text{syn}}(t, j, l) = s_i) / N_{\text{ens}}$ 

```

Algorithm 3: Generate synthetic data $\pi^{\text{syn}}(t, j)$.

states at the previous time step, i.e.,

$$u_2(t, j) := \mathbf{average}_{r \in \text{neigh}(j)}(\pi(t, r)). \quad (58)$$

It allows us to model the spatial relations and to evaluate the statistical impact of adjacent location states. To be able to speak of neighbors in the spatial sense, a honeycomb lattice is assumed and each hexagon is assigned to one location. The choice of this lattice allows us to work with six neighbors for every location, each sharing an edge with the considered cell. To generate the data, we define matrices

$$P_0^{1 \text{ syn}} = \begin{bmatrix} 0.7 & 0.3 \\ 0.7 & 0.3 \end{bmatrix}, \quad P_1^{1 \text{ syn}} = \begin{bmatrix} 0.28 & -0.28 \\ 0.28 & -0.28 \end{bmatrix}, \quad P_2^{1 \text{ syn}} = \begin{bmatrix} -0.01 & 0.01 \\ -0.01 & 0.01 \end{bmatrix} \quad (59)$$

and

$$P_0^{2 \text{ syn}} = \begin{bmatrix} 0.3 & 0.7 \\ 0.3 & 0.7 \end{bmatrix}, \quad P_1^{2 \text{ syn}} = \begin{bmatrix} 0.24 & -0.24 \\ 0.24 & -0.24 \end{bmatrix}, \quad P_2^{2 \text{ syn}} = \begin{bmatrix} 0.05 & -0.05 \\ 0.05 & -0.05 \end{bmatrix}. \quad (60)$$

The primary focus of this example lies on checking the techniques' attributes. This includes the ability to infer good (i.e., unbiased) approximations of the model

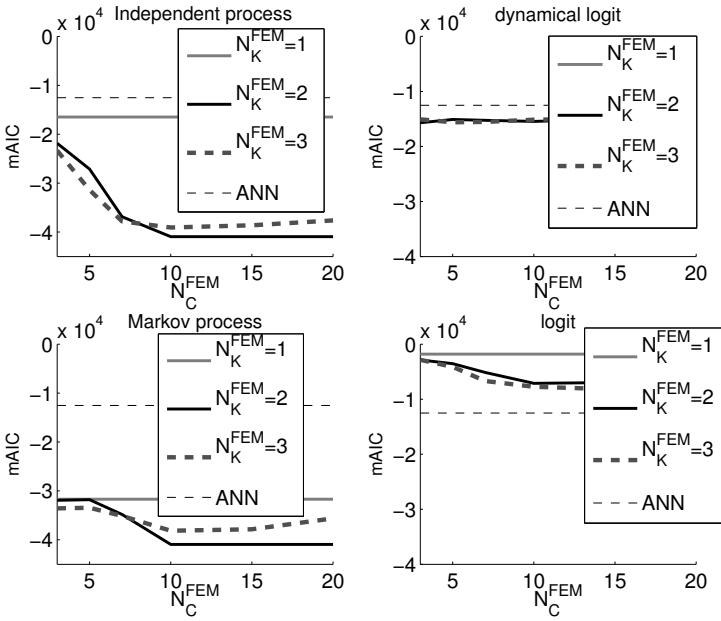


Figure 2. The four panels display the mAIC values for different parameters $N_K^{\text{FEM}} \in \{1, 2, 3\}$ and $N_C^{\text{FEM}} \in \{3, 5, 7, 10, 15, 20\}$ whereas each panel corresponds to a different model ansatz: Markov model, independent process, logistic model and dynamical logistic model. Additionally, the mAIC value calculated for the ANN results is shown.

parameters (i.e., $\Gamma^{\text{syn}}(t, j)$, $P^{k \text{ syn}}(u(t, j))$, N_K^{syn} and N_C^{syn}) as well as to generate a qualitative estimate of the distribution $\pi^{\text{syn}}(t, j)$. The proposed framework (four different direct model functions are considered, i.e., Markov and logit both with and without memory) is applied to the training sequence $\{1, \dots, 360\}$ (i.e., $N_{T_{\text{train}}} = 360$) of the synthetic data $\pi^{\text{syn}}(t, j)$ and the subspace algorithm is iterated $N_{\text{anneal}}^{\text{FEM}} = 10$ (for all four model assumptions) times in order to find a global minimum.⁹ The calculation is done for different parameters values $N_K^{\text{FEM}} \in \{1, 2, 3\}$ and $N_C^{\text{FEM}} \in \{3, 5, 7, 10, 15, 20\}$. Further, the corresponding mAIC values are computed with the proposed adapted information criterion. The resulting values are displayed in Figure 2. As can be seen in the panels on the left side of Figure 2, the mAIC values for the originally chosen maximal number of transitions N_C^{syn} and number of regimes N_K^{syn} are the lowest for the Markov framework with and without memory (i.e., the variables $N_C^{* \text{ Markov}} = N_C^{\text{syn}}$ and $N_K^{* \text{ Markov}} = N_K^{\text{syn}}$ are correctly identified). The results for the runs with logistic model assumptions (again with and without memory) have much bigger mAIC values (displayed in the panels on the right-hand side of Figure 2). Moreover, the mAIC value corresponding to the results of a neural network run

⁹For the remainder of the paper, we denote the AIC-optimal parameters computed by the framework with a superscripted *.

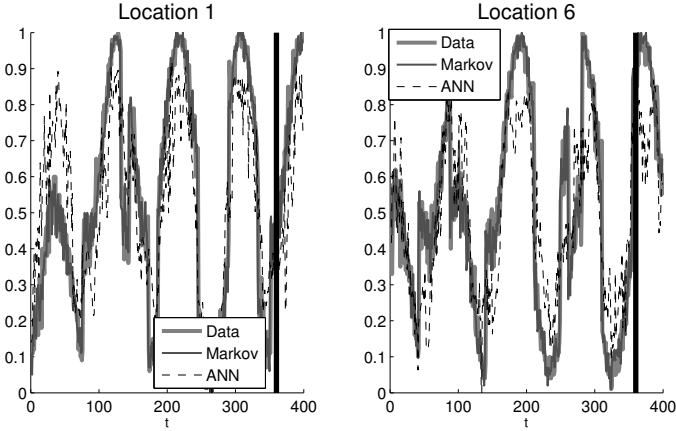


Figure 3. Approximations of the synthetic data $\pi_1^{\text{syn}}(t, j)$ retrieved with two different techniques: ANN (settings: $N_{\text{neurons}}^{\text{ANN}} = 20$, Levenberg–Marquardt backpropagation and $N_{\text{anneal}}^{\text{ANN}} = 10$) and nonstationary Markov regression (settings: no memory, $N_{\text{anneal}}^{\text{Markov}} = 10$, $N_C^{\text{Markov}} = 10$ and $N_K^{\text{Markov}} = 2$) are presented. Each of the panels corresponds to a location. The vertical black line at time $N_{\text{train}} = 360$ marks the last data point of the training data and the beginning of prediction sequence. The ANN approximation $\pi_1^{\text{ANN}}(t, j)$ is shown as a thin dashed line, and the approximation $\pi_1^{\text{Markov}}(t, j)$ obtained with the Markov model is displayed as a thin solid line.

(details below) is also presented in each of the four panels. The calculated model parameters of the Markov process without memory applied to the synthetic data for $N_K^{\text{Markov}} = 2$ and $N_C^{\text{Markov}} = 10$ are used to simulate $\pi^{\text{Markov}}(t, j)$ employing Algorithm 3 with parameters $\Gamma^*(t, j)$ and $P^*(u(t, j))$ (see Figure 3). It is compared to results obtained with artificial neural networks (ANN) [2; 24; 18; 3] and support vector machines (SVM) [8; 35]. These techniques are popular pattern-recognition algorithms and can both be used to model spatiotemporal data. As a representative ANN, we consider a feedforward network, more specifically a multilayer perceptron (MLP) [3]. According to the theory, a network of this particular architecture with two hidden layers can be used to approximate an arbitrary nonlinear function [23]. For many cases, a single-layer network (with an arbitrary depth, i.e., number of neurons) is enough and can already describe most of the practically relevant functions [18]. Typically used transfer function classes are linear-, step- or sigmoid-functions. Multilayer feedforward networks with logistic sigmoid transfer functions are universal approximators [18], and therefore, we will deploy this type of ANN in the numerical tests below. We train networks with different numbers of hidden neurons and continue with the network that has the smallest residuals ($N_{\text{neurons}}^{\text{ANN}} = 20$). This means that a particular ANN with $N_{\text{neurons}}^{\text{ANN}} = w$ neurons is considered to be the best fit when $\sum_{t,j} \|\pi^{\text{syn}}(t, j) - \pi^{\text{ANN}(w)}(t, j)\|_2^2 \leq \sum_{t,j} \|\pi^{\text{syn}}(t, j) - \pi^{\text{ANN}(v)}(t, j)\|_2^2$ for all of the regarded neuron numbers $v, w \in [5, 10, 15, 20, 25, 30, 40, 50]$. The

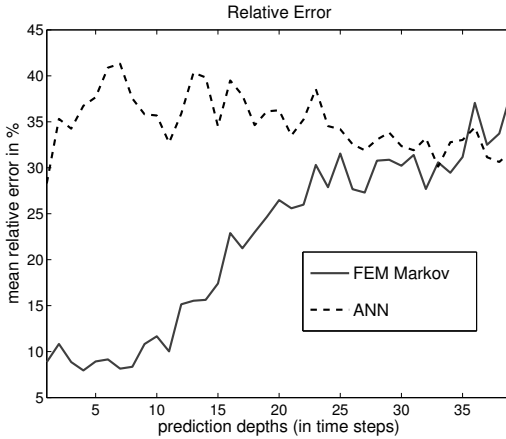


Figure 4. The mean relative error in % (in Euclidean metric) is shown dependent on the prediction depth $\tau \in \{1, \dots, N_{\text{pred}}\}$ (note that $N_{\text{pred}} = 39$). More specifically, the shown prediction error is computed as follows: $\text{mean}_j(\varpi(j, \tau) / \|\pi^{\text{syn}}(N_{T_{\text{train}}} + \tau, j)\|_2^2) * 100$ with $j \in \{1, \dots, N_J\}$ (details can be found in [Algorithm 4](#)).

Levenberg–Marquardt backpropagation is employed to optimize the network, and since it only converges to a local minimum, we also use annealing steps ($N_{\text{anneal}}^{\text{ANN}} = 10$) to approach a global solution.

An attempt to reconstruct the synthetic data $\pi_1^{\text{syn}}(t, j)$ for the entire time sequence, i.e., $t \in \{1, \dots, 400\}$, with the two different techniques, namely ANN and the Markov regression proposed in [Section 3](#), is shown in [Figure 3](#). Regarding the test sequence $\{1, \dots, N_{T_{\text{train}}}\}$, the approximation $\pi_1^{\text{ANN}}(t, j)$ (see the thin dashed line in the panels of [Figure 3](#)), corresponding to ANN, mostly follows the original path $\pi_1^{\text{syn}}(t, j)$. The performance of the ANN framework is also satisfactory when confronted with the test data (i.e., external factors $u(t, j)$ with $t \in \{361, \dots, 400\}$, starting from the thick black vertical line in both panels of [Figure 3](#)). The Markov regression technique (see the thin solid line in panels of [Figure 3](#)) restores the original series in the training sequence, i.e., in the first 360 time steps, more accurately than the ANN. In pursuance of approximating $\Gamma^{\text{syn}}(t, j)$ for $t \in \{N_{T_{\text{train}}}, \dots, N_T\}$, the self-contained strategy outlined in [Section 4](#) is employed. Details of the procedure to obtain $\hat{\pi}^{\text{Markov}}(t, j)$, i.e., approximating the synthetic data for the test sequence, can be found in the pseudocode of [Algorithm 4](#).

As can be seen in [Figure 3](#) (right from the vertical black line), the nonstationary nonhomogeneous Markov regression provides a high quality approximation $\hat{\pi}^{\text{Markov}}(t, j)$ of the artificial time series $\pi^{\text{syn}}(t, j)$ in the test sequence. The quality of the calculated model can also be accessed comparing the estimated local Markov parameters matrices with the synthetic ones $P_0^{k \text{ syn}}, \dots, P_{N_E}^{k \text{ syn}}$ with

input : $\Gamma^*(t, j)$ for $t \in \{1, \dots, N_{T_{\text{train}}}\}$, set maximal prediction depth N_{pred} and $u(t, j)$ for $t \in \{1, \dots, N_T\}$
output: $\varpi(j, \tau)$ with $\tau \in \{1, \dots, N_{\text{pred}}\}$ and $\hat{\pi}^{\text{Markov}}(t, j)$ with $t \in \{N_{T_{\text{train}}} + 1, \dots, N_T\}$

- 1 **for** $j = 1 : N_J$ **do**
- 2 Determine model parameter $P^\Gamma(u(t, j))$ characterizing the underlying model of $\Gamma^*(t, j)$ via stationary Markov regression.
- 3 **for** $\tau = 1 : N_{\text{pred}}$ **do**
- 4 $\hat{\Gamma}(N_{T_{\text{train}}} + \tau, j) = \Gamma^*(N_{T_{\text{train}}}, j) \prod_{h=0}^{\tau-1} P^\Gamma(u(N_{T_{\text{train}}} + h, j))$ (see (53))
- 5 Generate $\hat{\pi}(N_{T_{\text{train}}} + \tau, j)$ employing [Algorithm 3](#) (lines 3 to 10) using $\hat{\Gamma}(N_{T_{\text{train}}} + \tau, j)$.
- 6 $\varpi(j, \tau) = \|\pi^{\text{syn}}(N_{T_{\text{train}}} + \tau, j) - \hat{\pi}^{\text{Markov}}(N_{T_{\text{train}}} + \tau, j)\|_2^2$

Algorithm 4: Prediction.

$k \in \{1, \dots, N_K^{\text{syn}}\}$ (given in (59) and (60)) that have been used to generate the data

$$P_0^{1 \text{ Markov}} = \begin{bmatrix} 0.6999 & 0.3001 \\ 0.3001 & 0.6999 \end{bmatrix}, \quad P_1^{1 \text{ Markov}} = \begin{bmatrix} 0.2801 & -0.2801 \\ 0.2801 & -0.2801 \end{bmatrix}, \quad (61)$$

$$P_2^{1 \text{ Markov}} = \begin{bmatrix} -0.0125 & -0.0515 \\ -0.0125 & -0.0515 \end{bmatrix}$$

and

$$P_0^{2 \text{ Markov}} = \begin{bmatrix} 0.3003 & 0.69971 \\ 0.3003 & 0.69971 \end{bmatrix}, \quad P_1^{2 \text{ Markov}} = \begin{bmatrix} 0.24 & -0.24 \\ 0.24 & -0.24 \end{bmatrix}, \quad (62)$$

$$P_2^{2 \text{ Markov}} = \begin{bmatrix} 0.0515 & -0.0515 \\ 0.0515 & -0.0515 \end{bmatrix}.$$

Furthermore, the error plot of [Figure 4](#) also indicates the superiority of the Markov model in terms of relative prediction error

$$\varpi_{\text{rel}}(\tau) = \underset{j}{\text{mean}}(\varpi(j, \tau) / \|\pi^{\text{syn}}(N_{T_{\text{train}}} + \tau, j)\|_2^2) * 100 \quad (63)$$

up to a prediction depth of approximately 23 time steps ahead. The computation of the error term $\varpi(j, \tau)$ is explained in [Algorithm 4](#). An alternative possibility to model the discrete/categorical processes is provided by the support vector machines. SVMs are used for the classification of a given data set $u(t, j)$ with $t \in \{1, \dots, N_T\}$ and $j \in \{1, \dots, N_J\}$ with respect to a set of different *classes* (or states) $\{s_1, \dots, s_{N_S}\}$. This is achieved via geometrical separation, i.e., appropriate placing of hyperplanes in \mathcal{U} , dividing the values $u(t, j)$ in N_S different segments, thus associating $u(t, j)$ for each t and j with one class/state s_i . In the training phase, the assignment of the data values $u(t, j)$ to the classes is computed according to the values of the discrete process $\sigma^{\text{syn}}(t, j, l)$. As the microscopic information about the discrete states of the

process is unavailable, a threshold of 0.5 is set and $\pi^{\text{syn}}(t, j)$ is rounded accordingly so that the data has two categories, i.e., two classes. The optimization problem corresponding to the SVMs can be formulated as a quadratic minimization procedure resulting in a unique robust solution. In contrast, the ANNs (that are fitted through a nonconvex gradient-based optimization procedure) do not provide a unique solution of the inverse problem and therefore are in general less robust than SVMs. Different kernel functions are considered (specifically linear, quadratic, polynomial and radial basis functions), and the best fit (again regarding the residuals) was obtained for the radial basis function. The SVM run takes less computing time than the MLP run but needs a lot of support vectors to characterize the process. This overfitting is reflected in the very big $\text{mAIC} = 3.5193 \times 10^4$ value. In general, the computational complexity of SVMs with Gaussian radial basis function kernel (in the worst case) is $\mathcal{O}(N_T^2 N_E)$ for the training of each location [4]. But in most of the cases, it is possible to considerably reduce the computation time, e.g., by working with small values of the regularization parameter N_C^{SVM} for a faster convergence or, alternatively, increasing the number of training samples [34]. Determination of an optimal feedforward network with a nonlinear transfer function for a set of considered training data also requires solving a sequence of quadratic optimization problems. For the ANN calculations in this paper, we employed the Levenberg–Marquardt backpropagation algorithm, which is known to be very efficient [14]. Note, however, that the technique scales badly with the number of involved weights $N_{\text{weights}}^{\text{ANN}} = N_{\text{neurons}}^{\text{ANN}} (N_E + N_E^2 + \text{biases})$ (it is necessary to compute the inverse of the $N_{\text{weights}}^{\text{ANN}} \times N_{\text{weights}}^{\text{ANN}}$ Hessian matrix in each iteration step, which has a complexity $\mathcal{O}((N_{\text{weights}}^{\text{ANN}})^3)$). It is advisable to switch to a different gradient-descent algorithm for high-dimensional systems (i.e., systems that require more than a couple hundred weights) [39]. Further, the ANN fitting requires a longer run time due to the necessary annealing steps.

The SVM results are visualized in Figure 5 along with the approximations determined with the nonstationary Markov regression and the neural network (settings like in Figure 3). The assignment calculated with the SVM in general corresponds to the original data. Wrong categorization in the form of single outliers is mostly caused by data values too close to the threshold 0.5. Longer periods of wrong association especially in the test time sequence suggest that support vector machines are not feasible for prediction of spatiotemporal data of this particular nature.

Summing up, the proposed regression framework provides feasible and qualitative results. Nevertheless, it is important to mention that the considered synthetic data in this section is inherently designed to suit the model technique. The aim here was not to prove the overall superiority of the proposed algorithm in comparison to standard methods like ANN and SVM but to give the reader an idea of its capabilities under “good” conditions and as a contrast to the ill-posed example with missing external factors outlined in the next section.

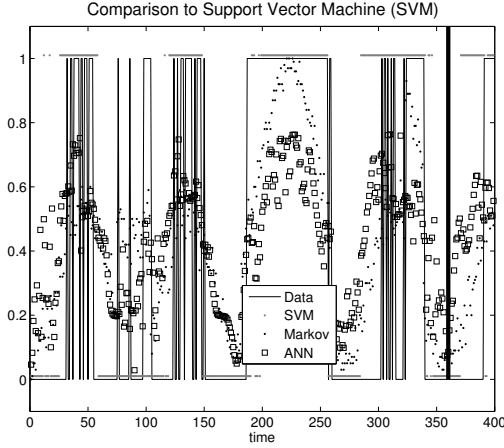


Figure 5. Dotted approximations of $\pi_1^{\text{syn}}(t, j)$ (for one fixed location) determined with a nonstationary Markov regression and a feedforward neural network and an output of support vector machines are shown. The beginning of the predicted time series is marked with a vertical black line.

5B. Example with missing (implicit) external factors. A key conceptual advantage of the proposed Markov regression framework is that implicit external factors, influencing the data, can be reflected in the nonstationary and nonhomogeneous formulation of the model. In order to numerically investigate this property, the framework is applied to a synthetic time series $\pi^{\text{syn}}(t, j)$ ($N_S = 2$) generated employing [Algorithm 3](#) with the number of implicit external factors set to $N_I = 100$ and the number of regimes fixed to be one ($N_K^{\text{syn}} = 1$), i.e., the artificial system is stationary and homogeneous and influenced by forces $u^{\text{unres}}(t, j)$ not available as observations.¹⁰ For the construction, we choose one explicit external factor (computed as a mean of neighboring states of the previous time step) and 100 implicit influences in the form of sinus functions (randomly chosen between: $u_e^{\text{unres}}(t, j) := \sin^2((2\pi te)/360 + j/20)$ and $u_e^{\text{unres}}(t, j) := \cos^2((2\pi te)/360 + j/20)$) depending on time t ($N_T := 400$), location j ($N_J := 24$) and the index of the particular external factor $e \in \{1, \dots, N_I\}$. Further, the model matrices for the one considered regime are defined:

$$P_0^{1 \text{ syn}} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}, \quad P_1^{1 \text{ syn}} = \begin{bmatrix} 0.05 & -0.05 \\ 0.05 & -0.05 \end{bmatrix}, \quad P_2^{1 \text{ syn}} = \begin{bmatrix} 0.42 & -0.42 \\ 0.42 & -0.42 \end{bmatrix} \quad (64)$$

and

$$P_{e+2}^{1 \text{ syn}} = \begin{bmatrix} 0.0002 & -0.0002 \\ 0.0002 & -0.0002 \end{bmatrix} \quad \forall e \in \{1, \dots, N_I - 2\}. \quad (65)$$

¹⁰Note that it is not necessary to use [Algorithm 2](#) since $\Gamma^{\text{syn}}(t, j) := \mathbf{ones}(1, N_T, N_J)$.

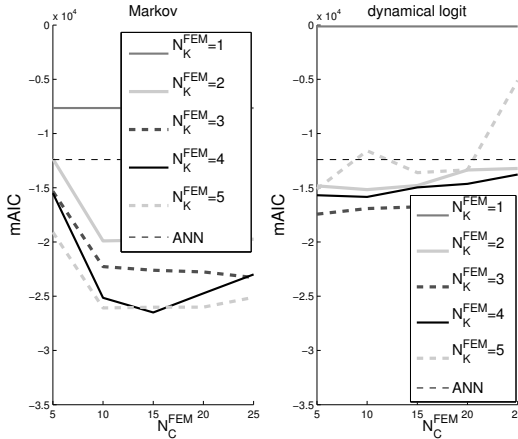


Figure 6. The mAIC values for runs of the Markov regression and the dynamical logistic regression applied to the second synthetic data series run for different values of N_C^{FEM} and N_K^{FEM} are displayed in this graph. Moreover, the value for the ANN result is shown.

The first implicit external factor $\bar{u}_2(t, j) = u_1^{\text{unres}}(t, j)$ thereby has the most significant influence, and all the other external factors have a much smaller impact. The proposed nonstationary nonhomogeneous Markov regression is applied to part of the generated data (i.e., $\pi^{\text{syn}}(t, j)$ with $t \in \{1, \dots, 360\}$ and $j \in \{1, \dots, 24\}$) for $N_K^{\text{FEM}} \in \{1, 2, 3, 4, 5\}$ and $N_C^{\text{FEM}} \in \{5, 10, 15, 20, 25\}$ with $N_{\text{anneal}}^{\text{FEM}} = 10$. Note that the implicit external factors $u^{\text{unres}}(t, j)$ are not made available for the regression procedure. The optimal model fit is determined via the modified information criterion (46). The resulting graphs can be seen in the left panel of Figure 6.

The lowest mAIC value has a model with up to 15 transitions between four regimes, i.e., $N_C^{\text{Markov}} = 15$ and $N_K^{\text{Markov}} = 4$. Thus, the synthetic stationary homogeneous model is described with a nonstationary and nonhomogeneous model capturing the original process and reflecting the implicit external factors $u^{\text{unres}}(t, j)$. In contrast, the dynamical logistic regression, applied to the data set, has bigger mAIC values and hence represents a worse description for the analyzed data. Two approximations of the ensemble distribution $\pi^{\text{syn}}(t, j)$ for different locations are shown in Figure 7. The plots illustrate that the nonstationary nonhomogeneous Markov regression is feasible even for observations where the biggest part of the relevant information is not provided in the form of measurements. The data $\pi^{\text{syn}}(t, j)$ in the test sequence, i.e., $t \in \{361, \dots, 400\} \forall j$, is approximated by computing a one-step prediction $\hat{\Gamma}(361, j)$ (see (53)) and using Algorithm 3 to determine $\hat{\pi}^{\text{Markov}}(361, j) \forall j$. Then the proposed Bayesian-update scheme is employed to update $\hat{\Gamma}^{\text{Markov}}(361, j)$ (see (56) in Proposition 4.1) using new data information $\pi^{\text{syn}}(361, j)$. These steps are iterated until $\hat{\pi}^{\text{Markov}}(N_T, j)$ can be calculated (note that the updated $\hat{\Gamma}(t, j)$ is used as the affiliation parameter $\Gamma^*(t, j)$

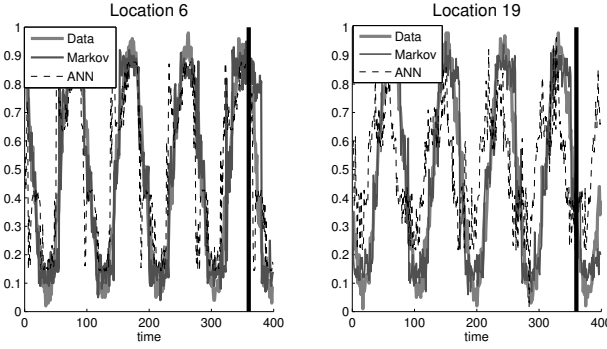


Figure 7. Each graph displays approximations of the data $\pi_1^{\text{syn}}(t, j)$ generated by means of different models, i.e., Markov regression (settings: no memory, $N_{\text{anneal}}^{\text{FEM}} = 10$, $N_{\text{C}}^* \text{Markov} = 15$, $N_{\text{K}}^* \text{Markov} = 4$) and an optimal ANN (settings: $N_{\text{neurons}}^{\text{ANN}} = 10$, Levenberg–Marquardt backpropagation, $N_{\text{anneal}}^{\text{ANN}} = 10$). The artificial time series $\pi_1^{\text{syn}}(t, j)$ is shown as a thick gray line. The start of the prediction is marked with a black vertical line at $N_{T_{\text{train}}} = 360$.

for all previous time steps t in the prediction sequence $\{361, \dots, 400\}$). The resulting prediction has a good quality as can be seen from the right-hand side of the vertical black line in the two panels of Figure 7.

In order to give an impression on the feasibility of standard techniques under “bad” conditions, such as artificially generated for this example, ANNs are applied to $\pi^{\text{syn}}(t, j)$ (settings: $N_{\text{neurons}}^{\text{ANN}} = 10$, Levenberg–Marquardt backpropagation and $N_{\text{anneal}}^{\text{ANN}} = 10$). The quality of the ANN results strongly depends on the location. This is caused by the dependence of the implicit external factors on the location; i.e., the implicit impact on the data is differing for each cell. In other words, the ANN framework does not allow restoring the devolution of the data without the additional information of the implicit external factors for location 19 and all other locations that are strongly influenced by the unresolved quantities. This is due to the fact that, in contrast to the nonstationary and nonhomogeneous Markov regression model presented above, the parameters (such as neuron weights and biases) of the standard ANN are time- and location-independent. In other words, ANN as well as SVM represent intrinsically stationary and homogeneous models. Because of this reason, both ANN and SVM as model classes have difficulties in capturing the effects of unobserved external factors. Concluding, it is possible to obtain qualitative results with ANN for the constructed dynamical system when enough information is provided in the form of data (see Section 5A) but is not a reliable option for realistic systems with data availability problems.

5C. Assimilation of additional information. The purpose of this example is to demonstrate the application of the Bayesian-update scheme (see Proposition 2.1

in Section 4) when compared to a simple maximum-likelihood allocation of new data (see (54)) or machine-learning algorithms like SVMs or ANNs. To this end, a transition path $\Gamma^{\text{syn}}(t, j)$ (employing Algorithm 2) of length $N_T = 10000$, switching between $N_K^{\text{syn}} = 2$ local models and $N_C^{\text{syn}} = 5$ transitions, was generated for $N_J = 10$ different locations. This path was then used to generate a time series, switching between $N_S = 2$ discrete values s_1 and s_2 without external influences (i.e., $N_F = 0$ and $N_{\text{ens}} = 1$) according to the following rules:

- (1) In the first model θ_1 , the process at time t is modeled by a Bernoulli-random variable with a probability 0.6 to be in the state s_1 .
- (2) For the second model θ_2 , a Markov chain is used to obtain the value of $\sigma(t, j, l)$; here the probability for the next value to be in different state than the previous value is 0.3.

For the training of the model, the natural choice for this example is the nonstationary nonhomogeneous Markov regression model (as introduced in (34)); the first 9000 time steps are chosen as a training set (i.e., $N_{T_{\text{train}}} = 9000$). To obtain a statistically significant result, the analysis is done not only for one but for 200 different time series (as already mentioned, $N_{\text{ens}} = 1$), all sharing the same transition path $\Gamma^{\text{syn}}(t, j)$. This allows us to draw first statistical conclusions and make the comparison of different methods independent of a single stochastic realization of the process. Since the focus is on the statistical significance rather than on the size of the ensemble, it is necessary to interpret the outcome of a single observation as the corresponding ensemble data, i.e.,

$$\pi_i^{\text{syn}}(t, j) = \delta_{s_i}(\sigma^{\text{syn}}(t, j, l)) \quad \text{for } i \in \{1, 2\}, \quad (66)$$

where δ_{s_i} is the Kronecker delta for the value s_i (i.e, being one if s_i is observed, else zero). To predict the incoming values of the time series ($t > N_{T_{\text{train}}} = 9000$), one needs to predict the affiliation vector $\Gamma^*(t, j)$ first. To this end, a self-contained strategy, proposed in Section 4, is employed. In other words, a transition matrix P^Γ is fitted to the 9000 elements of the transition path. This Markov chain is then used to propagate the current distribution of the affiliation to the next step and so forth. Of course, this makes the prediction very sensitive to finding the correct affiliation of data points [30] not included in the initial analysis of the time series. To demonstrate this sensitivity, the updating procedure as in Section 4 is compared to an SVM, an ANN and the maximum-likelihood affiliation (defined in (54)) of the data points. The SVM and ANN are additionally provided with the previous observation as this is used in the other two assimilation methods as well; thus, all four methods can make use of the same input information. To this end, the dimension of the data is doubled by creating the vectors $[\pi^{\text{syn}}(t, j) \pi^{\text{syn}}(t - 1, j)]$. Additionally, different kernel functions were tried for SVM and different transfer

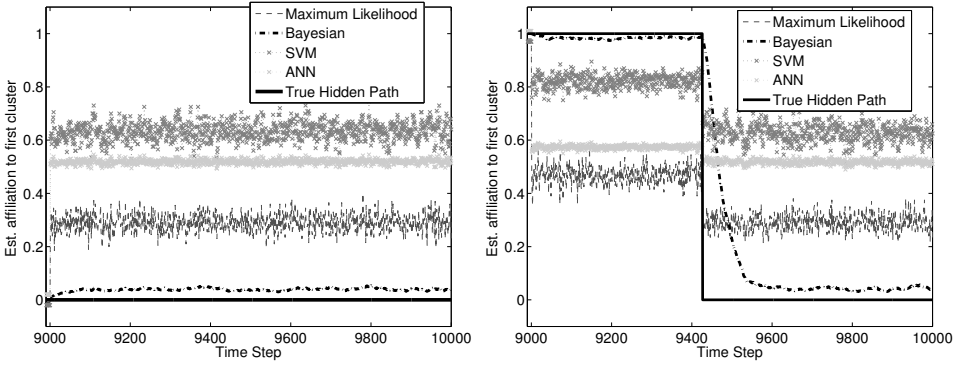


Figure 8. Average assimilation of 1000 untrained data points to the clusters for two different transition paths (in two different locations). The sample consisted of 200 different realizations of the time series with 9000 training points. To improve visibility, the allocations are shifted by up to 0.02. The beginning of the prediction is time step 9001.

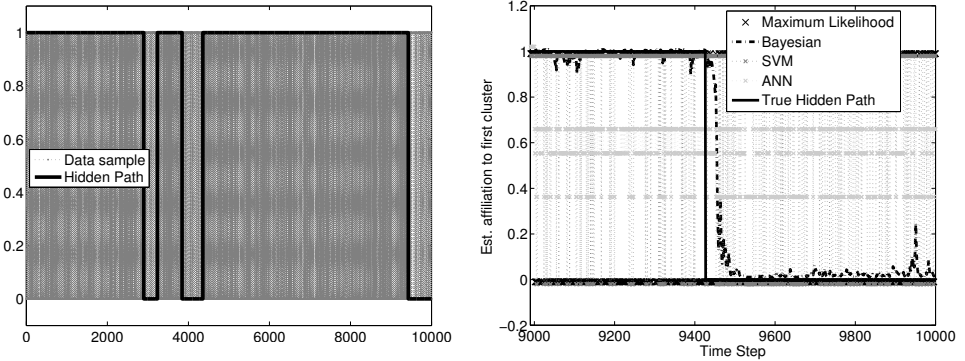


Figure 9. Left panel: Typical data set and the transition path used for the creation of the data. Right panel: Result of the assimilation schemes; only the relevant points are shown. The beginning of the assimilation is time step 9001. As can be seen, only the Bayesian assimilation scheme (black dashed line) based on Proposition 2.1 completely recovers the true persistent structure of the original hidden process (black solid line).

functions and numbers of neurons for the ANN; an optimal configuration in each model class was obtained applying the standard AIC procedure. Out of the 10 locations, two are shown here, one with constant original allocation in the prediction time frame (Figure 8, left panel) and one with a jump in the allocation (Figure 8, right panel). Additionally, a typical data set is shown in Figure 9 (left panel) and the affiliation functions resulting from the different assimilation methods are depicted in the right panel of Figure 9.

All four updating procedures generate affiliations that are not free of errors. To measure the quality of an allocation, the average distance

$$\frac{1}{(N_T - N_{T_{\text{train}}}) * N_J} \sum_{t=N_{\text{pred}}+1}^{N_T} \sum_{j=1}^{N_J} |\hat{\gamma}_1(t, j) - \gamma_1(t, j)|$$

of the estimated affiliation and the original path is averaged over all 200 realizations. Resulting error estimates are shown in [Table 1](#).

Algorithm	Error
maximum-likelihood affiliation	0.3142
Bayesian update (see Section 4)	0.0384
SVM-based affiliation	0.6188
ANN-based affiliation	0.4948

Table 1. Average distance of the affiliation of new data to the true path.

All estimators are then used to predict the next 10 time steps, i.e., $N_{\text{pred}} = 10$, according to the following algorithm:

input : data $\pi^{\text{syn}}(t, j)$, maximal prediction depth N_{pred} and the affiliation $\Gamma^*(t, j)$ for $t \in \{1, \dots, N_{T_{\text{train}}}\}$
output : $\hat{\pi}^{\text{Markov}}(t, j)$ and $\varpi(t, j, \tau)$ for $t \in \{N_{T_{\text{train}}} + 1, \dots, N_T\}$, $j \in \{1, \dots, N_J\}$ and $\tau \in \{1, \dots, N_{\text{pred}}\}$

- 1 Set start of test data $N_{T_{\text{train}}} = 9000$.
- 2 $\hat{\Gamma}(N_{T_{\text{train}}}, j) := \Gamma^*(N_{T_{\text{train}}}, j) \forall j$
- 3 **for** $j = 1 : N_J$ **do**
- 4 **for** $t = N_{T_{\text{train}}} : N_T - N_{\text{pred}}$ **do**
- 5 **for** $\tau = 1 : N_{\text{pred}}$ **do**
- 6 $\hat{\Gamma}(t + \tau, j) = \hat{\Gamma}(t, j) \prod_{h=0}^{\tau-1} P^\Gamma(u(t+h, j))$ (see (53))
- 7 Generate $\hat{\pi}(t + \tau, j)$ employing [Algorithm 3](#) (lines 3 to 10) using $\hat{\Gamma}(t + \tau, j)$.
- 8 $\varpi(t, j, \tau) = \|\pi_1^{\text{syn}}(t + \tau, j) - \hat{\pi}_1^{\text{Markov}}(t + \tau, j)\|_2^2$
- 9 Incorporate the observation $\pi^{\text{syn}}(t + 1, j)$ into the data set, and estimate its affiliation $\hat{\Gamma}(t + 1, j)$ for all j .

Algorithm 5: Prediction.

The quality of the prediction is measured by $\varpi(t, j, \tau)$, the squared distance of the synthetic data and the predicted probability for observing one (see line 8, [Algorithm 5](#)). These errors are then averaged for every τ over the 200 different realizations, the 10 locations and the prediction period.

As can be seen from [Table 1](#) and [Figure 10](#), the posterior estimators based on [Proposition 4.1](#) significantly outperforms other considered methods. Yet it should be noted that the process is rapidly mixing and thus hard to predict in the first place.

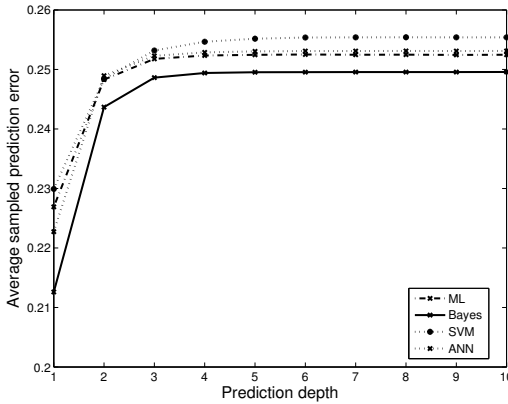


Figure 10. Mean of the sampled prediction errors for up to 10 time steps and four different assimilation schemes. The sample consisted of 200 different realizations of sets of 10 time series with length 1000.

This property increases the challenge all the assimilation methods have to face as the two different model-states are hard to separate even visually (see, e.g., the gray line in the left panel of [Figure 9](#)). Additionally, it should be noted that the average errors of the predictions for all four assimilation methods are rather similar; this is again a result of the low persistency of the rapidly mixing observed process. Nevertheless, the better assimilation of the missing information in the form of the affiliation function Γ (introduced in the current manuscript) leads to a reduction of the prediction error even for this very tough case, raising hope for better predictive models and better assimilation of the effects induced by the unresolved external factors as captured by the affiliation functions Γ .

6. Conclusion

The proposed nonstationary and nonhomogeneous regression framework represents a very promising way for modeling of spatiotemporal discrete jump processes under the presence of unobserved external impacts. As demonstrated in the current manuscript, it can capture the most significant impacts of the unobserved external factors described by [Proposition 2.1](#).

This was demonstrated by means of an example with additionally incorporated implicit external factors that were not made available for the calculation of the model parameters. Since incomplete data sets represent one of the central challenges in the field of time-series analysis, this property makes the presented methodology potentially useful in many areas of multiscale modeling and simulation, where discrete processes (e.g., associated with the phase transitions in physics) are subject to unresolved subgrid-scale effects.

Along the lines of traditional data assimilation, a new Bayesian algorithm to assimilate the model affiliation function $\widehat{\Gamma}(t, j)$ (capturing an impact of the unresolved external factors) was introduced and shows promising results. The proposed Bayesian algorithm for discrete data assimilation provides considerably better results than the currently available standard methods (i.e., maximum-likelihood assimilation, ANN and SVM) for the considered “tough” example of a rapidly switching nonstationary and nonhomogeneous discrete process.

It should be stressed that the adequacy of the presented models is largely relying on the validity of the underlying assumptions in [Proposition 2.1](#) as well as on the validity of the stationary homogeneous Markov assumption for the model-affiliation process (capturing an impact of unresolved external factors).

Because of this reason, in some situations, it might be necessary to use a nonstationary model formulation for the affiliation process and to include the additional necessary variables in the validation of the modified information criterion. In other words, in such situations, the optimal fit given by the nonstationary discrete regression model parameters and parametrization of the optimal spatiotemporal model for the hidden process Γ (beyond the stationary approximation deployed in this work) should be approached simultaneously. Although this new direction will allow constructing more realistic models with less a priori assumptions, it would also require many more computational resources than the proposed numerical framework. Numerical complexity estimates presented in this paper demonstrate that the deployment of concepts from high-performance computing and supercomputing computational facilities will also be necessary to extend all of the considered methods to realistic numbers of spatial locations and lengths of the time series. This issue is also a matter of the ongoing research.

Appendix A: Proof of [Proposition 2.1\(3\)](#)

Proof. Without loss of generality, we can assume that the external factors are ordered such that the explicit factors are the first N_E entries of $\bar{u}(t, j)$. By performing a Taylor expansion on the transition matrix $P(\bar{u}(t, j))$ around the means $\mu(t, j) = [\mathbf{E}(\bar{u}_1(t, j)), \dots, \mathbf{E}(\bar{u}_{N_E+N_I}(t, j))] \in \mathbb{R}^{(N_E+N_I) \times 1}$, we obtain

$$\begin{aligned}
 P(\bar{u}(t, j)) &= P(\mu(t, j)) + \sum_{e=1}^{N_E+N_I} \frac{\partial P(\mu(t, j))}{\partial \bar{u}_e(t, j)} (\bar{u}_e(t, j) - \mu_e(t, j)) \\
 &+ \sum_{e,h=1}^{N_E+N_I} \frac{\partial^2 P}{\partial \bar{u}_e(t, j) \partial \bar{u}_h(t, j)} (\bar{u}_e(t, j) - \mu_e(t, j)) (\bar{u}_h(t, j) - \mu_h(t, j)) \\
 &+ \sum_{|\alpha|=3} R_\alpha(\bar{u}(t, j)) (\bar{u}(t, j) - \mu(t, j))^\alpha, \tag{1}
 \end{aligned}$$

where α is a multi-index and

$$R_\alpha(\bar{u}(t, j)) = \frac{3}{\alpha!} \int_0^1 (1-x) D^\alpha P(\mu(t, j) + x(\bar{u}(t, j) - \mu(t, j))) dx. \quad (2)$$

Note that $R_\alpha(\bar{u}(t, j))$ is bounded as the third derivative of P is assumed to be bounded. Resorting the terms and defining

$$\rho_h(t, j) = 2 \sum_{e=N_E+1}^{N_E+N_I} \frac{\partial^2 P}{\partial \bar{u}_h(t, j) \bar{u}_e(t, j)} (u_e(t, j) - \mu_e(t, j)), \quad h = 1, \dots, N_E, \quad (3)$$

$$\begin{aligned} \check{\varepsilon}(t, j) &= \sum_{e=N_E+1}^{N_E+N_I} \frac{\partial P(\mu(t, j))}{\partial \bar{u}_e(t, j)} (\bar{u}_e(t, j) - \mu_e(t, j)) \\ &\quad + \sum_{|\alpha|=3} R_\alpha(\bar{u}(t, j)) (\bar{u}(t, j) - \mu(t, j))^\alpha \\ &\quad + \sum_{e,h=1}^{N_E} \frac{\partial^2 P}{\partial \bar{u}_h(t, j) \bar{u}_e(t, j)} (u_e(t, j) - \mu_e(t, j)) (u_h(t, j) - \mu_h(t, j)) \\ &\quad - \sum_{e=1}^{N_E} \mu_e(t, j) \rho_h(t, j) + \sum_{e,h=N_E+1}^{N_E+N_I} \frac{\partial^2 P}{\partial \bar{u}_h(t, j) \bar{u}_e(t, j)} \\ &\quad \quad \quad * (u_h(t, j) - \mu_h(t, j)) (u_e(t, j) - \mu_e(t, j)), \end{aligned} \quad (4)$$

$$\varepsilon(t, j) = \check{\varepsilon}(t, j) - \mathbf{E}[\check{\varepsilon}(t, j)] \quad (5)$$

yields (11) whereas the definition of matrices $P_h(t, j)$ is given in (14) for all t and j and $h \geq 1$ and $P_0(t, j)$ is defined as in (16), and replacing the expectation in the formula by the expectation of $\check{\varepsilon}(t, j)$. Moreover, $\mathbf{E}[\varepsilon(t, j)] = 0$ and $\mathbf{E}[\rho_h(t, j)] = 0$. \square

Appendix B: Notation

The notation index is organized as follows. The numbers and sizes are listed separately as their notation is very similar. The remaining notations are listed in order of appearance in the manuscript. To improve readability, the titles of sections and subsections are indicated. Moreover, the abbreviations used in the manuscript are listed at the end of the notation index.

Numbers and sizes.

- N_S total number of states s_i (associated index i , p. 3).
- N_{ens} (associated index l , p. 3).
- N_J space dimension of observations $\pi(t, j)$ for all time steps t (associated index j , p. 3).
- N_T length of observed time series $\pi(t, j)$ for fixed location j (associated index t , p. 3).
- $N_{s_i}(t, j)$ number of cells j currently (at time t) in state s_i (p. 3).

- N_E total number of explicit external factors (associated index e , p. 5).
- N_I total number of implicit external factors (associated index e , p. 5).
- N_F total number external factors (associated index e , p. 5).
- N_M memory depth (p. 8).
- N_K total number of local stationary homogeneous models θ_k (associated index k , p. 13).
- N_C maximal number of allowed transitions of the affiliation processes $\gamma_k(t, j)$ for fixed j .
- $N_{\text{anneal}}^{\text{FEM}}$ total number of annealing steps used for the FEM framework (p. 16).
- N_{ϕ_k} degree of a polynomial of parametric (conditional) probability density function ϕ_k (p. 18).
- N_{pred} prediction depth (p. 20).
- $N_{T_{\text{train}}}$ time-wise length of training data (p. 22).
- N_C^{syn} artificially chosen maximal number of transitions of the synthetic affiliation processes $\gamma_k^{\text{syn}}(t, j)$ (p. 24).
- N_K^{syn} artificially chosen total number of local stationary homogeneous models θ_k^{syn} (p. 24).
- N_K^{FEM} number of local regimes considered for the general FEM framework (p. 26).
- N_C^{FEM} number of maximal transitions considered for the general FEM framework (p. 26).
- N_{dummy} auxiliary quantity of [Algorithm 2](#) (p. 24).
- $N_C^*{}^{\text{Markov}}$ optimal in terms of the mAIC values (with respect to the data) maximal number of transitions for the parameters computed with the Markov regression framework (p. 26).
- $N_K^*{}^{\text{Markov}}$ optimal in terms of the mAIC values (with respect to the data) maximal number of local stationary models computed with the Markov regression framework (p. 26).
- $N_{\text{neurons}}^{\text{ANN}}$ total number of employed neurons for an ANN run (p. 27).
- $N_{\text{anneal}}^{\text{ANN}}$ total number of annealing steps used for an ANN run (p. 28).
- $N_{\text{anneal}}^{\text{Markov}}$ total number of annealing steps used for the Markov regression (p. 27).
- $N_{\text{weights}}^{\text{ANN}}$ is the number of ANN parameters (p. 30).
- $N_{\text{basis}}^{\text{FEM}}$ number of finite elements used for the discretization (p. 16).
- N_C^{SVM} regularization parameter of SVM (p. 30).

Ensemble data and exterior quantities.

- s_i discrete state (p. 3).
- $\omega(j, l)$ microscopic cell (p. 3).
- $\sigma(t, j, l)$ with $j \in \{1, \dots, N_J\}$ and $l \in \{1, \dots, N_{\text{ens}}\}$ dynamical process of a microscopic cell $\omega(j, l)$ (p. 3).
- $\tilde{\pi}_i(t, j)$ empirical probability for process $\sigma(t, j, l)$ to be in state s_i in location $\omega(j, l)$ at time t (Definition (1), p. 3).
- $N_{s_i}(t, j)$ total number of microscopic cells $\omega(j, t)$ in state s_i for fixed t and j (Definition (2), p. 3).
- $\delta_{s_i}(\cdot)$ the Kronecker delta for the value s_i (p. 4).
- $\tilde{\pi}(t, j) \in [0, 1]^{N_s \times 1}$ vector of empirical probabilities (Definition (3), p. 4).
- $\pi_i(t, j)$ probability for process $\sigma(t, j, l)$ to be in state s_i in location $\omega(j, l)$ at time t (Definition (4), p. 4).

- $\mathbb{P}[\cdot]$ probability function.
- $\pi(t, j) \in [0, 1]^{N_S \times 1}$ vector of states probabilities (Definition (5), p. 4).

Implicit external factors.

- $P(\bar{u}(t, j)) \in [0, 1]^{N_S \times N_S}$ transition matrix dependent on all external factors (p. 4).
- $\bar{u}(t, j) \in \mathbb{R}^{(N_E + N_I) \times 1}$ all influencing external factors (Definition (7), p. 5).
- \mathbb{R} real numbers.
- $u_e(t, j) \in \mathbb{R}$ explicit external factor.
- $u(t, j) \in \mathbb{R}^{N_E \times 1}$ vector of explicit external factors (Definition (8), p. 5).
- $\mathcal{U} \subset \mathbb{R}^{N_E \times 1}$ vector space of explicit external factors $u(t, l)$.
- $u_e^{\text{unres}}(t, j) \in \mathbb{R}$ implicit external factor.
- $u^{\text{unres}}(t, j) \in \mathbb{R}^{N_I \times 1}$ vector of implicit external factors (Definition (9), p. 5).
- $\varepsilon(t, j)$ error term associated with decomposition of transition matrix $P(\bar{u}(t, j))$ (Definition (15), p. 7).
- $\mathbf{E}(\cdot)$ expected value.
- $\rho_e(t, j)$ second noise process for decomposition of $P(\bar{u}(t, j))$ with second derivatives. (Definition (3), Appendix A, p. 39).
- $\mu(t, j) \in \mathbb{R}^{N_I \times 1}$ vector of expected values for each of the entries of vector $\bar{u}(t, l)$ (p. 7).
- $R_\alpha(\bar{u}(t, j))$ Taylor-expansion error component (Definition (13), p. 7).
- α a multi-index (p. 7).
- $P_e(t, j)$ matrix used in the linear combination equal to $P(t, l, u(t, l))$ corresponding to $u_e(t, j)$ for $e \in \{1, \dots, N_S\}$ (Definition (14), p. 7).
- $P_0(t, j)$ matrix used in the linear combination equal to $P(t, l, u(t, l))$ (Definition (16), p. 7).
- $P(t, j, u(t, j)) \in [0, 1]^{N_S \times N_S}$ equal to $P(\bar{u}(t, j))$ assuming the conditions of Proposition 2.1 are fulfilled.

Inverse problem formulation.

- $f(\cdot)$ a general direct mathematical model (Definition (17), p. 8).
- $\theta(\bar{u}(t, j))$ unknown model parameter dependent on all external factors (p. 8).
- Ω parameter space containing $\theta(\bar{u}(t, j))$ (p. 8).
- $\lambda(t, j)$ error term of simple model example (p. 9).
- $g(\cdot)$ model distance function (Definition (19), p. 9).
- $\mathbf{L}(\cdot)$ averaged clustering functional (Definition (20), p. 9).
- $d(\cdot, \cdot)$ metric (p. 9).
- $d_2(\cdot, \cdot)$ Euclidean metric (p. 9).
- f^{logit} logistic direct mathematical model function (p. 9).
- f^{Markov} Markov direct mathematical model function (p. 9).

Logistic regression.

- $\mathcal{C}_i[u(t, j), B^i(t, j)]$ utility measure (Definition (23), p. 10).

- $B^i(t, j) \in \mathbb{R}^{(N_E+1) \times 1}$ logistic model parameter corresponding to state s_i for $i \in \{1, \dots, N_S\}$ (Definition (24), p. 10).
- $\beta_e^i(t, j)$ e -th entry of vector $B^i(t, j)$.
- $\xi^i(t, j)$ error process of utility measure (p. 10).
- $B(t, j) \in \mathbb{R}^{(N_E+1) \times N_S}$ nonstationary nonhomogeneous logistic model parameter (p. 11).
- $\theta^{\text{logit}}(B(t, j), u(t, j))$ logistic model parameter (Definition (28), p. 11).
- $\zeta(t, j)$ error term of logistic model distance function (p. 11).

Interpolation.

- $\theta_k(u(t, j))$ stationary homogeneous model parameter (p. 13).
- $\gamma_k(t, j)$ weighting process corresponding to local model $\theta_k(u(t, j))$ (p. 13).
- $\Theta(u(t, j))$ vector of stationary homogeneous model parameters (p. 13).
- $\Gamma(t, j) \in [0, 1]^{1 \times N_K}$ vector of affiliation processes (p. 13).
- $\mathbf{L}(\cdot, \cdot)$ interpolated version of averaged clustering functional $\mathbf{L}(\cdot)$ (Definition (38), p. 13).
- $\mathbf{L}_j(\cdot, \cdot)$ one summand for a fixed location j of interpolated average clustering functional (Definition (39), p. 14).
- $B_k \in \mathbb{R}^{(N_E+1) \times N_S}$ local logit model parameter (p. 14).
- B_k^i i -th entry of stationary and homogeneous logit model parameter vector B_k (p. 14).
- $P^k(u(t, j))$ local Markov model parameter matrix (Definition (41), p. 14).
- $P(u(t, j)) \in \mathbb{R}^{N_S \times N_S N_K}$ vector of model matrices $P^k(u(t, j))$ (p. 14).
- $P_0^k, \dots, P_{N_E}^k$ for all k matrices used in the linear combination equal to $P^k(u(t, l))$ (p. 14).
- $\mathbf{1}$ auxiliary column vector containing only entries equal to one (p. 14).
- $\mathbf{0}$ auxiliary column vector containing only entries equal to zero (p. 14).
- $\{P_e^k\}_{n,m}$ entry of matrix P_e^k in n -th row and m -th column (p. 15).

Spatial and temporal persistence.

- $|\cdot|_{\text{BV}(1, N_T)}$ bounded variation (BV) half-norm (Definition (45), p. 15).

Numerical approach and computational complexity.

- $\Gamma^*(t, j) = [\gamma_1^*(t, j), \dots, \gamma_{N_K}^*(t, j)] \in [0, 1]^{1 \times N_K}$ global optimizer with respect to $\mathbf{L}(\Gamma(t, j), \Theta(u(t, j)))$ (p. 15).
- $\Theta^*(u(t, j))$ global optimizer with respect to $\mathbf{L}(\Gamma(t, j), \Theta(u(t, j)))$ (p. 15).
- Γ_r computed Γ process dependent on annealing index (p. 17).
- $\Gamma_r^{[h]}$ determined Γ process dependent on annealing index and optimization iteration index (p. 17).
- Θ_r computed model parameter Θ dependent on annealing index (p. 17).
- $\Theta_r^{[h]}$ determined model parameter Θ dependent on annealing index and optimization iteration index (p. 17).
- \mathbf{L}_{\min} auxiliary variable of Algorithm 1 (p. 17).
- κ auxiliary variable used to describe the order of the computational costs (p. 16).

Information criterion.

- $\text{mAIC}(\cdot, \cdot)$ modified version of Akaike information criterion for presented framework (Definition (46), p. 18).
- $\mathcal{L}(\cdot, \cdot)$ log-likelihood (Definition (47), p. 18).
- $\phi_k(\cdot, \dots, \cdot | N_{\phi_k})$ parametric (conditional) probability density function (PDF) (p. 18).
- $M(\cdot, \cdot)$ function computing total number of involved parameters (p. 18).
- $M^{\text{logit}}(\cdot, \cdot)$ function computing total number of involved parameters for a logistic model (Definition (48), p. 18).
- $M^{\text{Markov}}(\cdot, \cdot)$ function computing total number of involved parameters for Markov model (Definition (49), p. 18).
- \mathcal{S}_1 finite discrete set of different values for variable N_K (p. 19).
- \mathcal{S}_2 finite discrete set of different values for variable N_C (p. 19).
- $\Delta(\cdot, \cdot)$ mAIC model ranking (Definition (50), p. 19).
- $w(\cdot, \cdot)$ mAIC model weights (Definition (51), p. 19).

Prediction and assimilation of additional information.

- $\hat{\pi}(t, j)$ prediction of observation $\pi(t, j)$ (p. 20).
- $\hat{\Gamma}(t, j) = [\hat{\gamma}_1(t, j), \dots, \hat{\gamma}_{N_K}(t, j)] \in [0, 1]^{1 \times N_K}$ prediction of future affiliations (p. 20).
- $P^\Gamma(t, j)$ transition matrix characterizing $\Gamma^*(t, j)$ (p. 20).
- $P_0^\Gamma, \dots, P_{N_E}^\Gamma$ matrices used in linear combination equal to $P^\Gamma(t, j)$ (p. 20).
- $\dot{\gamma}(N_T + 1, j)$ posterior estimate based on the new observation $\pi(N_T + 1, j)$ (p. 21).
- $\dot{y}_k(N_T + 1, j)$ updated affiliation associated with local model θ_k (Definition (56), p. 21).

Numerical investigation.

- $\Gamma^{\text{syn}}(t, j)$ synthetic affiliation process (p. 23).
- $\gamma_k^{\text{syn}}(t, j)$ synthetic affiliation associated with θ_k^{syn} (p. 24).
- dummy1 auxiliary vector of Algorithm 2 containing only ones (p. 24).
- dummy0 auxiliary vector of Algorithm 2 containing only zeros (p. 24).
- $P^{\text{syn}}(t, j, u(t, j))$ synthetic transition matrix (Definition (57), p. 23).
- $P^{k \text{ syn}}(u(t, j))$ synthetic model parameter matrix associated with affiliation $\gamma_k^{\text{syn}}(t, j)$ (p. 23).
- $\sigma^{\text{syn}}(t, j, l)$ synthetic dynamical process (p. 23).
- $\pi^{\text{syn}}(t, j)$ synthetic data (p. 25).
- $P_0^{k \text{ syn}}, \dots, P_{N_E}^{k \text{ syn}}$ synthetic model matrices corresponding to explicit external factors $u(t, j)$ (p. 25).
- $P_{N_E+1}^{k \text{ syn}}, \dots, P_{N_E+N_I}^{k \text{ syn}}$ synthetic model matrices corresponding to implicit external factors $u^{\text{unres}}(t, j)$ (p. 25).
- $\pi_i^{\text{syn}}(t, j)$ i -th vector entry of synthetic data $\pi^{\text{syn}}(t, j)$ (p. 25).
- $\pi^{\text{ANN}(w)}(t, j)$ approximation of $\pi^{\text{syn}}(t, j)$ computed with an ANN based on a network with w neurons (p. 27).
- $\pi^{\text{ANN}}(t, j)$ approximation of $\pi^{\text{syn}}(t, j)$ computed with an ANN (p. 28).

- $\pi^{\text{Markov}}(t, j)$ approximation of $\pi^{\text{syn}}(t, j)$ computed with Markov regression framework (p. 27).
- $\varpi(j, \tau)$ prediction error term dependent on location j and prediction depth τ (p. 29).
- $\varpi_{\text{rel}}(\tau)$ relative mean prediction error (p. 29).
- $\varpi(t, j, \tau)$ prediction error term dependent on time t , location j and prediction depth τ (p. 36).
- $\check{\epsilon}(t, j)$ auxiliary process used in the proof of [Proposition 2.1](#) (p. 39).

Abbreviations.

- SVM support vector machines.
- ANN artificial neural networks.
- AIC Akaike information criterion.
- mAIC modified Akaike information criterion.
- GLM generalized linear models.
- PDEs partial differential equations.
- ODEs ordinary differential equations.
- FEM finite-element method.
- IIA independence of irrelevant alternatives.
- i.i.d. independent and identically distributed.

Acknowledgements

The authors thank the DFG SPP 1276 MetStroem “Meteorology and Turbulence Mechanics”, the Swiss National Science Foundation (project 131845 “AnaGraph”), the Center for Scientific Simulation (Freie Universität Berlin) and the graduate research school GEOSIM (GFZ Potsdam, Universität Potsdam, Freie Universität Berlin) for funding. Further, the authors thank Professor Rupert Klein (Freie Universität Berlin) for stating the question about spatial heterogeneity of nonstationary models that led to a formulation of the problem for discrete processes considered in this manuscript. Special thanks to the unknown referees for their many helpful comments and hints regarding the deployed notation.

References

- [1] H. Akaike, *A new look at the statistical model identification*, IEEE Trans. Automatic Control **19** (1974), no. 6, 716–723. [MR 54 #11691](#) [Zbl 0314.62039](#)
- [2] C. M. Bishop, *Neural networks for pattern recognition*, Clarendon Press, New York, 1995. [MR 97m:68172](#)
- [3] C. Blume, K. Matthes, and I. Horenko, *Supervised learning approaches to classify stratospheric warming events*, J. Atmos. Sci. **69** (2012), no. 6, 1824–1840.
- [4] A. Bordes, S. Ertekin, J. Weston, and L. Bottou, *Fast kernel classifiers with online and active learning*, J. Mach. Learn. Res. **6** (2005), 1579–1619. [MR 2249866](#) [Zbl 1222.68152](#)

- [5] K. P. Burnham and D. R. Anderson, *Model selection and multimodel inference: a practical information-theoretic approach*, 2nd ed., Springer, New York, 2002. MR 1919620 Zbl 1005.62007
- [6] S. Chib and E. Greenberg, *Understanding the Metropolis–Hastings algorithm*, Am. Stat. **49** (1995), no. 4, 327–335.
- [7] J. S. Chipman, *The foundations of utility*, Econometrica **28** (1960), no. 2, 193–224. MR 22 #9284 Zbl 0173.48001
- [8] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*, Cambridge University Press, 2000. Zbl 0994.68074
- [9] J. de Wiljes, A. Majda, and I. Horenko, *An adaptive Markov chain Monte Carlo approach to time series clustering of processes with regime transition behavior*, Multiscale Model. Simul. **11** (2013), no. 2, 415–441. MR 3047436
- [10] A. J. Dobson and A. G. Barnett, *An introduction to generalized linear models*, 3rd ed., CRC Press, Boca Raton, 2008. MR 2010a:62003 Zbl 1165.62049
- [11] J. Fox, *Applied regression analysis, linear models, and related methods*, SAGE Publications, Thousand Oaks, 1997.
- [12] J. Gill, *Generalized linear models: a unified approach*, Quantitative Applications in the Social Sciences, no. 134, SAGE Publications, Thousand Oaks, 2001.
- [13] J. Hadamard, *Sur les problèmes aux dérivées partielles et leur signification physique*, Princeton University Bulletin **13** (1902), no. 4, 49–52.
- [14] M. T. Hagan and M. B. Menhaj, *Training feedforward networks with the Marquardt algorithm*, IEEE Trans. Neural Networks **5** (1994), no. 6, 989–993.
- [15] L. D. Haugh and G. E. P. Box, *Identification of dynamic regression (distributed lag) models connecting two time series*, J. Amer. Statist. Assoc. **72** (1977), no. 357, 121–130. MR 56 #4084
- [16] I. Horenko, *Nonstationarity in multifactor models of discrete jump processes, memory, and application to cloud modeling*, J. Atmos. Sci. **68** (2011), no. 7, 1493–1506.
- [17] ———, *On analysis of nonstationary categorical data time series: dynamical dimension reduction, model selection, and applications to computational sociology*, Multiscale Model. Simul. **9** (2011), no. 4, 1700–1726. MR 2012j:60196 Zbl 1244.60070
- [18] K. Hornik, M. Stinchcombe, and H. White, *Multilayer feedforward networks are universal approximators*, Neural Networks **2** (1989), no. 5, 359–366.
- [19] M. A. Katsoulakis, A. J. Majda, and A. Sopsakis, *Multiscale couplings in prototype hybrid deterministic/stochastic systems, I: Deterministic closures*, Commun. Math. Sci. **2** (2004), no. 2, 255–294. MR 2005m:76144 Zbl 1103.93013
- [20] ———, *Multiscale couplings in prototype hybrid deterministic/stochastic systems, II: Stochastic closures*, Commun. Math. Sci. **3** (2005), no. 3, 453–478. MR 2006j:34135 Zbl 1101.34042
- [21] ———, *Hybrid deterministic stochastic systems with microscopic look-ahead dynamics*, Commun. Math. Sci. **8** (2010), no. 2, 409–437. MR 2012c:60116 Zbl 1197.35336
- [22] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, *Optimization by simulated annealing*, Science **220** (1983), no. 4598, 671–680. MR 85f:90091 Zbl 1225.90162
- [23] V. Kurkova, *Kolmogorov’s theorem and multilayer neural networks*, Neural Networks **5** (1992), no. 3, 501–506.
- [24] J. Lawrence, *Introduction to neural networks: design, theory, and applications*, 6th ed., California Scientific Software Press, Nevada City, CA, 1994.

- [25] W. D. Li and C. A. McMahon, *A simulated annealing-based optimization approach for integrated process planning and scheduling*, Int. J. Comp. Integ. M. **20** (2007), no. 1, 80–95.
- [26] T. F. Liao, *Interpreting probability models: logit, probit, and other generalized linear models*, SAGE Publications, Thousand Oaks, 1994.
- [27] R. D. Luce, *Individual choice behavior: a theoretical analysis*, Wiley, New York, 1959. MR 21 #7127 Zbl 0093.31708
- [28] C. F. Manski and D. McFadden (eds.), *Structural analysis of discrete data with econometric applications*, MIT Press, Cambridge, MA, 1981. MR 83f:62158 Zbl 0504.00023
- [29] D. McFadden, *Conditional logit analysis of qualitative choice behaviour*, Frontiers in econometrics (P. Zarembka, ed.), Academic Press, New York, 1974, pp. 105–142.
- [30] P. Metzner, L. Putzig, and I. Horenko, *Analysis of persistent nonstationary time series and applications*, Commun. Appl. Math. Comput. Sci. **7** (2012), no. 2, 175–229. MR 3005737 Zbl 1275.62067
- [31] A. Y. Ng and M. I. Jordan, *On discriminative vs. generative classifiers: a comparison of logistic regression and naive Bayes*, Adv. Neural Inf. Process. Syst. (2002), 841–848.
- [32] A. Pankratz, *Forecasting with dynamic regression models*, Wiley, Hoboken, NJ, 1991.
- [33] C. P. Robert and G. Casella, *Monte Carlo statistical methods*, 2nd ed., Springer, New York, 2004. MR 2005d:62006 Zbl 1096.62003
- [34] S. Shalev-Shwartz and N. Srebro, *SVM optimization: inverse dependence on training set size*, Proceedings of the 25th International Conference on Machine Learning (A. McCallum and S. Roweis, eds.), 2008, pp. 928–935.
- [35] J. Shawe-Taylor and N. Cristianini, *Kernel methods for pattern analysis*, Cambridge University Press, 2004.
- [36] K. E. Train, *Discrete choice methods with simulation*, 2nd ed., Cambridge University Press, 2009. MR 2010m:91055 Zbl 1269.62073
- [37] S. A. Vavasis, *Quadratic programming is in NP*, Inform. Process. Lett. **36** (1990), no. 2, 73–77. MR 91m:68095 Zbl 0719.90052
- [38] J. von Neumann, *Various techniques used in connection with random digits*, National Bureau of Standards Applied Math Series **11** (1951), 36–38.
- [39] H. Yu and B. M. Wilamowski, *Levenberg–Marquardt training*, Intelligent systems (B. M. Wilamowski and J. D. Irwin, eds.), Industrial Electronics Handbook, no. 5, CRC Press, Boca Raton, 2nd ed., 2011.

Received November 29, 2012. Revised October 22, 2013.

JANA DE WILJES: jana.dewiljes@math.fu-berlin.de

Institute of Mathematics, Freie Universität Berlin, Arnimallee 6, D-14195 Berlin, Germany

LARS PUTZIG: lars.putzig@usi.ch

Institute of Computational Science, Università della Svizzera Italiana, Via Giuseppe Buffi 13, CH-6900 Lugano, Switzerland

ILLIA HORENKO: horenkoi@usi.ch

Institute of Computational Science, Università della Svizzera Italiana, Via Giuseppe Buffi 13, CH-6904 Lugano, Switzerland

Communications in Applied Mathematics and Computational Science

msp.org/camcos

EDITORS

MANAGING EDITOR

John B. Bell
Lawrence Berkeley National Laboratory, USA
jbbell@lbl.gov

BOARD OF EDITORS

Marsha Berger	New York University berger@cs.nyu.edu	Ahmed Ghoniem	Massachusetts Inst. of Technology, USA ghoniem@mit.edu
Alexandre Chorin	University of California, Berkeley, USA chorin@math.berkeley.edu	Raz Kupferman	The Hebrew University, Israel raz@math.huji.ac.il
Phil Colella	Lawrence Berkeley Nat. Lab., USA pcolella@lbl.gov	Randall J. LeVeque	University of Washington, USA rjl@amath.washington.edu
Peter Constantin	University of Chicago, USA const@cs.uchicago.edu	Mitchell Luskin	University of Minnesota, USA luskin@umn.edu
Maksymilian Dryja	Warsaw University, Poland maksymilian.dryja@acn.waw.pl	Yvon Maday	Université Pierre et Marie Curie, France maday@ann.jussieu.fr
M. Gregory Forest	University of North Carolina, USA forest@amath.unc.edu	James Sethian	University of California, Berkeley, USA sethian@math.berkeley.edu
Leslie Greengard	New York University, USA greengard@cims.nyu.edu	Juan Luis Vázquez	Universidad Autónoma de Madrid, Spain juanluis.vazquez@uam.es
Rupert Klein	Freie Universität Berlin, Germany rupert.klein@pik-potsdam.de	Alfio Quarteroni	Ecole Polytech. Féd. Lausanne, Switzerland alfio.quarteroni@epfl.ch
Nigel Goldenfeld	University of Illinois, USA nigel@uiuc.edu	Eitan Tadmor	University of Maryland, USA etadmor@cscamm.umd.edu
		Denis Talay	INRIA, France denis.talay@inria.fr

PRODUCTION

production@msp.org

Silvio Levy, Scientific Editor


See inside back cover or msp.org/camcos for submission instructions.

The subscription price for 2014 is US \$75/year for the electronic version, and \$105/year (+\$15, if shipping outside the US) for print and electronic. Subscriptions, requests for back issues from the last three years and changes of subscribers address should be sent to MSP.

Communications in Applied Mathematics and Computational Science (ISSN 2157-5452 electronic, 1559-3940 printed) at Mathematical Sciences Publishers, 798 Evans Hall #3840, c/o University of California, Berkeley, CA 94720-3840, is published continuously online. Periodical rate postage paid at Berkeley, CA 94704, and additional mailing offices.

CAMCoS peer review and production are managed by EditFLOW® from Mathematical Sciences Publishers.

PUBLISHED BY

 **mathematical sciences publishers**
nonprofit scientific publishing

<http://msp.org/>

© 2014 Mathematical Sciences Publishers

Communications in Applied Mathematics and Computational Science

vol. 9

no. 1

2014

- Discrete nonhomogeneous and nonstationary logistic and Markov regression models for spatiotemporal data with unresolved external influences 1
JANA DE WILJES, LARS PUTZIG and ILLIA HORENKO
- Low Mach number fluctuating hydrodynamics of diffusively mixing fluids 47
ALEKSANDAR DONEV, ANDY NONAKA, YIFEI SUN, THOMAS G. FAI,
ALEJANDRO L. GARCIA AND JOHN B. BELL
- High-order methods for computing distances to implicitly defined surfaces 107
ROBERT I. SAYE
- On inference of statistical regression models for extreme events based on incomplete observation data 143
OLGA KAISER AND ILLIA HORENKO