
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Aushev, Alexander; Tran, Thong; Pesonen, Henri; Howes, Andrew; Kaski, Samuel
Likelihood-free inference in state-space models with unknown dynamics

Published in:
STATISTICS AND COMPUTING

DOI:
[10.1007/s11222-023-10339-8](https://doi.org/10.1007/s11222-023-10339-8)

Published: 01/02/2024

Document Version
Publisher's PDF, also known as Version of record

Published under the following license:
CC BY

Please cite the original version:
Aushev, A., Tran, T., Pesonen, H., Howes, A., & Kaski, S. (2024). Likelihood-free inference in state-space models with unknown dynamics. *STATISTICS AND COMPUTING*, 34(1), Article 27.
<https://doi.org/10.1007/s11222-023-10339-8>

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.



Likelihood-free inference in state-space models with unknown dynamics

Alexander Aushev¹ · Thong Tran¹ · Henri Pesonen² · Andrew Howes³ · Samuel Kaski^{1,4}

Received: 23 May 2023 / Accepted: 11 October 2023
© The Author(s) 2023

Abstract

Likelihood-free inference (LFI) has been successfully applied to state-space models, where the likelihood of observations is not available but synthetic observations generated by a black-box simulator can be used for inference instead. However, much of the research up to now has been restricted to cases in which a model of state transition dynamics can be formulated in advance and the simulation budget is unrestricted. These methods fail to address the problem of state inference when simulations are computationally expensive and the Markovian state transition dynamics are undefined. The approach proposed in this manuscript enables LFI of states with a limited number of simulations by estimating the transition dynamics and using state predictions as proposals for simulations. In the experiments with non-stationary user models, the proposed method demonstrates significant improvement in accuracy for both state inference and prediction, where a multi-output Gaussian process is used for LFI of states and a Bayesian neural network as a surrogate model of transition dynamics.

Keywords Likelihood-free inference · Simulator-based inference · Bayesian optimisation · Multi-objective optimisation · State-space models · Non-linear dynamics

1 Introduction

Likelihood-free inference (LFI) methods (Sunnåker et al. 2013; Sisson et al. 2018; Cranmer et al. 2020) estimate the parameters θ of a statistical model, given an observed measurement \mathbf{x}_* and a black-box simulator g_θ . These methods use synthetic observations $\mathbf{x}_\theta \sim g_\theta(\mathbf{x} | \theta)$ produced by the simulator to assist the inference without requiring an analytical formulation of the likelihood $p(\mathbf{x} | \theta)$. LFI has been successfully applied to identifying parameters of complex real-world systems, such as financial markets (Peters

et al. 2012; Barthelmé and Chopin 2014; Ong et al. 2018), species populations (Beaumont et al. 2002b; Beaumont 2010; Bertorelle et al. 2010) and cosmology models (Schafer and Freeman 2012; Alsing et al. 2018; Jeffrey et al. 2021). A special type of application of LFI is time-dependent systems, which can be described using state-space models (SSMs) (Kalman et al. 1960; Koller and Friedman 2009) where observed measurements $\mathbf{x}_t \in \mathbb{R}^n$ are emitted given a series of latent variables, the states $\theta_t \in \mathbb{R}^m$, as illustrated in Fig. 1.

Compared to traditional Bayesian estimation, in a simulator-based setting, our primary aim is to understand how latent states evolve in relation to both the logic of the simulator and real-world observed data. Typically, state-space inference methods (Kalman et al. 1960; Anderson and Moore 2012; Zerdali and Barut 2017) require an observation model g_θ in the form of the likelihood $p(\mathbf{x}_t | \theta_t)$ to find the posterior distribution $p(\theta_{1:T} | \mathbf{x}_{1:T})$. When the observation model is unavailable, state-space learning methods (Frigola et al. 2014; Melchior et al. 2019) are commonly used to infer g_θ from the observed time-series data. However, when g_θ is inferred, the states become very difficult to interpret for domain experts since the states are no longer informed by a known model. An alternative solution to this problem is to use a simulator in place of g_θ . LFI methods are able to infer

✉ Alexander Aushev
alexander.aushev@aalto.fi

Samuel Kaski
samuel.kaski@aalto.fi

¹ Department of Computer Science, Aalto University, Otakaari 24, Espoo 02150, Finland

² Oslo Centre for Biostatistics and Epidemiology, Oslo University Hospital, Oslo, Norway

³ Department of Computer Science, University of Exeter, Exeter, UK

⁴ Department of Computer Science, University of Manchester, Manchester, UK

the states and avoid learning g_θ by using a simulator as the observation model. Simulators are widespread in SSM settings (Ghassemi et al. 2017; Shafi et al. 2018; Georgiou and Demiris 2017) since they enable the incorporation of additional prior knowledge about data-generating mechanisms without the need for a tractable likelihood $p(\mathbf{x}_t | \theta_t)$. In this paper, we focus on LFI for SSMs, which fall under the category of approximate methods in the broader context of SSM inference.

An essential aspect of SSMs that is often overlooked in the LFI literature is the complexity of transition dynamics h_{θ_t} . Current LFI methods for SSMs (Toni et al. 2009; Dean et al. 2014) proceed by assuming dynamics to be either too simplistic (e.g., linear) or readily available for sampling. In contrast, our approach stands out as especially valuable when the transition dynamics are complex, non-linear, and not immediately known, especially under a limited simulation budget of the observation model. Such complexities in state transitions, which deviate from simple linear or Gaussian norms, are frequently observed in diverse domains like meteorology (Errico et al. 2013; Zeng et al. 2020), cosmology (Lange et al. 2019; He et al. 2019) or behavioural sciences (Gimenez et al. 2007; Georgiou and Demiris 2017). In meteorology, for example, intricate dynamics (Kalnay 2003) are driven by a vast web of interconnected factors shaping weather patterns. In the realm of behavioural sciences (Kahneman and Tversky 2013; Fiske and Taylor 2013), human decision-making stands as a testament to complexity. Choices are shaped not only by an individual's past experiences but also by their current emotional states and cognitive biases. An instance of this is how new information can sway subsequent decisions, a phenomenon we delve deeper into in our later experiments. Traditional LFI methods, when not tailored to address these non-linear and non-Gaussian dynamics, frequently result in less-than-optimal state estimates and predictions. While there have been commendable advancements in LFI, such as the creation of more efficient sampling-based methods (Jasra et al. 2012), innovative statistic-matching generation mechanisms (Martin et al. 2019), and theoretical convergence confirmations (Dean et al. 2014; Martin et al. 2014; Calvet and Czellar 2015), they still fall short in addressing this core challenge.

In this paper, we introduce a method capable of likelihood-free state inference and state prediction in discrete-time SSMs. Our method operates in a LFI setting, where a time-series of observations \mathbf{x}_t and a simulator g_θ capable of replicating these observations are provided. The goal of the method is to infer the states $\theta_{1:T} = \{\theta_1, \dots, \theta_T\}$ that can produce the observed time-series $\mathbf{x}_{1:T} = \{\mathbf{x}_1, \dots, \mathbf{x}_T\}$, using as few simulations as possible to reduce their potentially high computational cost. This setting is broader than is typically assumed by traditional LFI methods since we do not assume the transition dynamics h_{θ_t} to be known (neither in its closed-

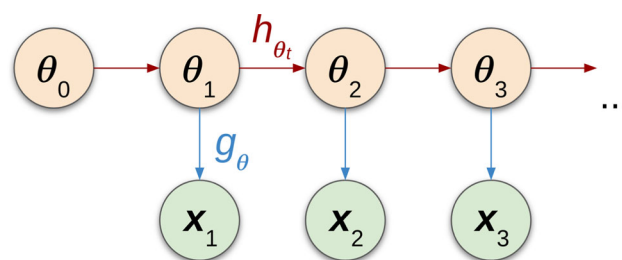


Fig. 1 Graphical representation of an SSM. Latent states θ_t (orange) produce observations \mathbf{x}_t through the observation simulator g_θ (blue) and follow the Markovian transition dynamics h_{θ_t} (red)

form nor its function family) or available for sampling, and also because the number of simulations can be limited to a small number. Instead of assuming the transition dynamics, we learn a non-parametric model and use it as their *surrogate* (or replacement) in state approximation and prediction.

This paper contains three main contributions. First, we propose a solution to the previously unaddressed problem of state prediction in SSMs with unknown transition dynamics and a limited simulation budget. We use samples from LFI approximations of state posteriors $p(\theta_t | \mathbf{x}_t)$ to accurately model the state transition dynamics, with accuracy shown by empirical comparisons with state-of-the-art SSM inference techniques. Second, focusing on problems where LFI has to be *sample-efficient*, i.e., the number of simulations needs to be reduced as much as possible, we improve upon the current LFI methods for the state inference task by leveraging time-series information. This is done by using a multi-objective surrogate for the consecutive states (e.g., for time-steps j and $j + 1$) and sampling from a transition dynamics model to determine where to next run simulations. Lastly, we demonstrate that the proposed method is needed to tackle the crucial case of user modelling, where user models are non-stationary because users' beliefs, preferences, and abilities change over time.

2 Background

Approximate Bayesian computation (ABC) (Beaumont et al. 2002a; Csilléry et al. 2010; Sunnåker et al. 2013) is arguably the most popular family of LFI methods. In its simplest variant, ABC with rejection sampling (Tavaré et al. 1997; Pritchard et al. 1999), the simulator parameters are repeatedly sampled from the prior $p(\theta)$ to generate synthetic observations \mathbf{x}_θ . These synthetic observations are then compared to the observed measurement \mathbf{x}_* using the so-called *discrepancy* measure $\delta(\theta) = \rho(\mathbf{x}_*, \mathbf{x}_\theta)$, where $\rho(\cdot, \cdot)$ is a distance function, e.g. Euclidean. If synthetic observations \mathbf{x}_θ have a discrepancy smaller than a user-defined threshold ϵ , then they are considered to be produced by simulator parameters

θ that could plausibly replicate the observed measurement \mathbf{x}_* . This common assumption in ABC approaches results in the following approximations of the likelihood function $\mathcal{L}(\cdot)$ and the posterior $p(\theta | \mathbf{x}_*)$:

$$\mathcal{L}(\theta) \approx \mathbb{E}[\kappa_\epsilon(\delta(\theta))], \quad p(\theta | \mathbf{x}_*) \propto \mathcal{L}(\theta) \cdot p(\theta). \quad (1)$$

Here $\kappa_\epsilon(\cdot)$ is a kernel with its maximum at zero and whose bandwidth ϵ acts as an acceptance/rejection threshold. For instance, in ABC with rejection sampling, $\kappa_\epsilon(\delta(\theta)) = \xi_{[0, \epsilon)}(\delta(\theta))$, where $\xi_{[0, \epsilon)}(\delta(\theta))$ equals to one if $\delta(\theta) \in [0, \epsilon)$ and zero otherwise. Unfortunately, ABC approaches need many simulations of synthetic observations to accurately approximate the posterior, making them unsuitable for inference with computationally intensive simulators.

2.1 Bayesian optimisation for LFI

Since many applications, including those considered in this paper, aim to minimise the number of simulations, other methodologies have emerged, such as Bayesian optimisation for LFI (BOLFI) (Gutmann and Corander 2016). In BOLFI, a Gaussian process (GP) surrogate is used for a discrepancy measure $\delta(\theta)$, where the minimum of the GP surrogate mean function $\mu(\theta)$ can be used as ϵ and a Gaussian CDF $F((\epsilon - \mu(\theta))/\sqrt{v(\theta) + \sigma^2})$ with mean 0 and variance 1 as $\mathbb{E}[\kappa_\epsilon(\cdot)]$ in Eq. (1). Here, $v(\theta) + \sigma^2$ is the posterior variance of the GP surrogate.

A main advantage of modelling the discrepancy with a GP is the ability to estimate uncertainty. The GP’s predictive mean $\mu(\theta^{(i)})$ and variance $v(\theta^{(i)})$ are used to calculate the utility (e.g., expected improvement, Brochu et al. 2010) of sampling the objective function at the next candidate point $\theta^{(i+1)}$, where i denotes the number of a simulation. Maximising this so-called *acquisition* function $\mathcal{A}(\cdot)$ with respect to θ helps determine where to run simulations next. Because BOLFI actively chooses where to run simulations, its posterior approximation requires much fewer synthetic observations than other LFI methods that do not use active learning. However, BOLFI was not specifically designed for SSMs and hence does not make use of any temporal information typical for SSMs to enhance inference quality.

2.2 Sequential neural estimation

An alternative approach to sample-efficient LFI is global sequential neural estimation (SNE), which learns the statistical relationship between observations and simulator parameters directly through a neural network surrogate. If trained with a sufficiently large sample set, this surrogate does not need retraining when the observation changes, making SNE methods particularly suitable for a sequence of related inference tasks, such as those required in time-series

prediction. Although there exist amortised versions of neural approximation methods, the specific sequential variants highlighted here are not naturally amortised. In our dynamic framework, these methods are employed to address separate LFI problems across different time-steps, ensuring the use of consistent priors throughout. The SNE neural network can be used as a surrogate for the posterior, likelihood, or likelihood ratio, resulting in SNPE (Papamakarios and Murray 2016; Goncalves et al. 2018; Greenberg et al. 2019), SNLE (Papamakarios et al. 2019), and SNRE (Durkan et al. 2020; Hermans et al. 2020) methods respectively. These SNE methods address a more difficult problem than we do: learning a model across all possible tasks (i.e., observed datasets). The price is that they require significantly more simulations than Bayesian optimisation (BO) approaches, as seen in Sect. 4.3 of Aushev et al. (2020).

2.3 Likelihood approximation networks

Likelihood approximation networks (LANs), introduced by Fengler et al. (2021), share similarities with SNE approaches. LANs approximate the likelihood for time-dependent generative models in dynamical systems within cognitive neuroscience. Their key distinction is the assumption that the time component is one of the inputs of the observation model, allowing them to learn the observation model at an arbitrary time-step. This assumption shifts the role of the dynamics onto the observation model, which is often beneficial for diffusion models (Reynolds and Rhodes 2009; Wieschen et al. 2020), but not for models of human behaviour (Schall 2019; Futrell et al. 2020; Pothos and Chater 2002). In contrast, our approach does not rely on the explicit dependency of the observation model on time, enabling state predictions when the transition dynamics are unknown at the cost of amortisation.

2.4 Non-linear dynamics in non-LFI methods

The issue of handling non-linear transition dynamics, in general, has been primarily addressed outside of the LFI literature. This large and growing set of methods includes extended Kalman filters (Anderson and Moore 2012; Zerdali and Barut 2017), GP-SSMs (Frigola et al. 2014; Melchior et al. 2019), sequential Monte Carlo (Doucet et al. 2001; Smith 2013; Septier et al. 2013) and Bayes filtering (Smidl and Quinn 2008; Karl et al. 2016). Although they are not directly applicable to the LFI setting considered in this paper, we summarise them in Table 1 alongside relevant LFI literature to highlight important connections.

Table 1 Comparison of inference methods in SSMs with references to selected representative works

Reference	Method	LFI	Amortized	# Simulations	Dynamics
Rubin (1984)	ABC	✓	✗	≈10k	✗
Gutmann and Corander (2016)	BOLFI	✓	✗	≈50	✗
Papamakarios and Murray (2016)	SNEs	✓	✓	≈1k	✗
Fengler et al. (2021)	LANs	✓	✓	≈1k	✗
Jasra et al. (2012)	ABC/Filt.	✓	✗	≈10k	Linear
Izenman (1988)	Kalman	✗	✗	–	Linear
Anderson and Moore (2012)	Ext. Kalman	✗	✗	–	Non-linear
Doerr et al. (2018)	PR-SSM	✗	✗	–	Non-linear
Ialongo et al. (2019)	GP-SSM	✗	✗	–	Non-linear
This work	LMC-BNN	✓	✗	≈50	Non-linear

LFI methods use simulators to infer states (or simulator parameters) and can be either amortised (do not need to be retrained when the observations change) or non-amortised (valid only for the observed data). These methods also vary in the number of simulations required for inference and the type of dynamics they assume. We report the single observation budget (per time-step) for non-amortised methods and the total budget for amortised methods

3 Likelihood-free inference in state-space models

In this section, we introduce a multi-objective approach to LFI in SSMs, which improves the sample-efficiency of existing methods by using the model for discrepancy shared across consecutive states while also learning the model of the transition dynamics. The main elements of the solution are presented in Fig. 2. To estimate state points θ_t , given \mathbf{x}_t , we employ a multi-objective surrogate $\tilde{\delta}_\theta$ for discrepancies and then approximate the posterior over states $p(\theta_t | \mathbf{x}_t)$ with Eq. (1). At the same time, we randomly pair consecutive posterior samples (θ_j, θ_{j+1}) and train a non-parametric surrogate for the state transition \tilde{h}_{θ_t} , whose predictive posterior $p(\theta_{t+1} | \mathbf{x}_t)$ proposes candidates for future simulations. We summarise our approach in Algorithm 1, where θ_* denotes simulator parameter points shared across all time-steps. For in-depth details, please refer to “Appendix C (Section C.3.2)”.

3.1 Multi-objective state inference

As an extension to BOLFI, we employ a multi-objective surrogate model for the discrepancies $\delta_t(\theta_*) = \rho(\mathbf{x}_t, \mathbf{x}_\theta)$ at different t , considering multiple discrepancy objectives simultaneously and leveraging information between consecutive states. More specifically, we pass discrepancies of the consecutive states to the surrogate separately (e.g., $\delta_{t-1}(\cdot), \delta_t(\cdot)$), but through the use of shared parameters of the multi-objective surrogate, they become associated. This approach allows using a discrepancy model of the previous state to infer the current state instead of simply discarding it. Moreover, it allows for a much more flexible surrogate for

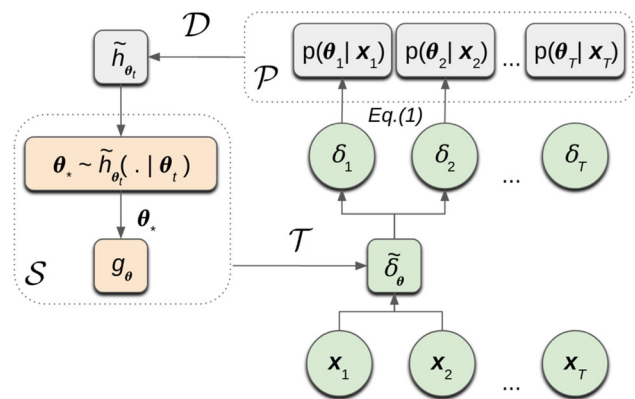


Fig. 2 An overview of our approach, in which the $\tilde{\delta}_\theta$ surrogate is used for LFI of states and \tilde{h}_{θ_t} for the unknown transition dynamics. The $\tilde{\delta}_\theta$ models the corresponding discrepancies $\delta_t \equiv \delta_t(\theta_*)$ of several observations (green) inside a moving window (here, with the size of two), from which posteriors are extracted according to Eq. (1) in \mathcal{P} . \tilde{h}_{θ_t} is trained with paired samples \mathcal{D} from posteriors of consecutive states (grey); its predictive samples are used as proposals (orange) for simulations \mathcal{S}

LFI of states than the traditional GP used in BOLFI. These changes do not need any additional data to fit the surrogates because all synthetic observations \mathbf{x}_θ for discrepancy objectives can be shared across all states (therefore, we use θ_* instead of θ_t in the context of simulations). When we consider a new observation \mathbf{x}_{t+1} , we simply need to recalculate the discrepancy values for all synthetic observations. Once we have a trained surrogate for discrepancy objectives, we infer state posteriors $p(\theta_t | \mathbf{x}_t)$, similarly as in BOLFI. This can be achieved, for example, through importance resampling, where prior samples are weighted according to the likelihood function $\mathcal{L}(\theta)$ from Eq. (1).

Algorithm 1 Multi-Objective LFI with Transition Model

Input: observations $\{\mathbf{x}_1, \dots, \mathbf{x}_T\}$, observation simulator g_θ , moving window size L , number of initial simulations B_0 , simulation budget B_{sim} , number of posterior samples I for states and K for state predictions, prior over states $p(\theta_*)$;
Output: state posteriors \mathcal{P} , transition model \tilde{h}_{θ_t} ;

Initialise an empty set for state posteriors $\mathcal{P} := \emptyset$;
 Initialise the transition model \tilde{h}_{θ_t} from Section 3.2;
 Simulate B_0 observations with parameters sampled from the prior $p(\theta_*)$;
 Form the set \mathcal{S} with the resulting data $\{(\theta_*, \mathbf{x}_\theta)\}_{b=1}^{B_0}$;
 Initialise the start and end indexes for the moving window: $t_0 := 0, t := L$;
while $t < T$ **do**
 Calculate discrepancies for \mathcal{S} inside the moving window $w := t_0:t$;
 Form the training set \mathcal{T} with the resulting data $\{(\theta_*, \delta_w(\theta_*))\}_{s=1}^{|\mathcal{S}|}$;
 Train the multi-objective model $\tilde{\delta}_\theta$ from Section 3.1 with \mathcal{T} ;
 Extract I posterior samples over states from $\tilde{\delta}_\theta$ with Equation (1);
 Replace the old posterior estimates in \mathcal{P} with the resulting samples $\{\theta_w\}_{i=1}^I$;
 Randomly sample K consecutive (some j and $j + 1$) state samples from \mathcal{P} ;
 Form the training set \mathcal{D} with the resulting data $\{\theta_j, \theta_{j+1}\}_{k=1}^K$;
 Update parameters of \tilde{h}_{θ_t} with data from \mathcal{D} through training;
 Simulate B_{sim} parameters proposed by \tilde{h}_{θ_t} with Equation (2);
 Augment \mathcal{S} with the resulting new data $\{(\theta_*, \mathbf{x}_\theta)\}_{b=1}^{B_{\text{sim}}}$;
 Move the moving window by adjusting its indexes: $t_0 := t_0 + 1, t := t + 1$;
end while

3.1.1 Moving window approach

There is an additional challenge in adapting multi-objective surrogates in SSMs: the high computational cost associated with considering too many objectives. Time-series can potentially have hundreds of time-points, and expanding the number of considered objectives may be detrimental to the performance of the surrogate. We avoid this problem by limiting the number of objectives the surrogate can have. Instead of considering all available time-steps as objectives, we propose to consider only L recent objectives by gradually including new ones and discarding old ones that have little impact on current states. The size of this *moving window* depends on how rapidly the transition dynamics change. As the size of the window L grows, the model becomes less sensitive to the noise from the dynamics, at the cost of increased computations and decreased adaptability to the most recent state transitions. Overall, the moving window reduces the number of objectives L considered at a time, making multi-objective modelling in the SSM setting feasible. In ‘‘Appendix A’’, we further investigate the influence of the moving window size hyperparameter on state inference and prediction and show that having only two objectives ($L = 2$) is the most beneficial choice in terms of the quality of posterior approximations and low computational time.

3.2 Learning state transition dynamics

While we progressively improve LFI posterior approximations $p(\theta_t | \mathbf{x}_t)$ by acquiring new simulations, we use empirical samples from the latest available approximations to learn a stochastic model of transition dynamics. This model

should be able to learn from noisy samples of LFI posterior approximations $p(\theta_t | \mathbf{x}_t)$, and be flexible enough to fit arbitrary function families the dynamics may follow. In addition, it should be able to handle uncertainty associated with samples outside the training distribution, as samples from posterior approximations tend to be concentrated around the main mode of the learning data. For these reasons, the appropriate transition model should be Bayesian and non-parametric (or semi-parametric). Such a model would account for the uncertainty associated with posterior approximations and be flexible enough to follow possibly non-linear transition dynamics.

We propose to train this model in an autoregressive fashion by forming a training set of K randomly paired sample points from posteriors (e.g., $p(\theta_{t-1} | \mathbf{x}_{t-1}), p(\theta_t | \mathbf{x}_t)$). More specifically, we assume the Markov property in the transition dynamics and use pairs of states instead of their whole trajectories. For each SSM time interval, we group consecutive state posterior samples in a training set, and expand it when new state posteriors become available (as we move forward in time). Thus, the transition model does not need to be retrained when new observations present themselves and can be actively used throughout state inference to determine where to run simulations next. This can be done by sampling the predictive posterior $p(\theta_{T+1} | \mathbf{x}_T)$ from the trained model \tilde{h}_{θ_T} :

$$p(\theta_{T+1} | \mathbf{x}_T) \approx \int \tilde{h}_{\theta_T}(\theta_{T+1} | \theta_T) \cdot p(\theta_T | \mathbf{x}_T) d\theta_T. \quad (2)$$

The posterior described above should be recognized as an approximate representation, informed by the data and

model, rather than an exact reflection of the true posterior. All later mentions of the posterior pertain to this approximation. Within this framework, the state transition model \tilde{h}_{θ_t} influences state posteriors indirectly, primarily serving as a source of simulation candidates for the LFI surrogate. Ultimately, accumulating more simulations improves the discrepancy surrogate for the LFI of states and, by extension, the quality of posterior samples, while higher-quality posterior samples allow for more accurate learning of state transition dynamics.

3.3 Computational complexity and model choices

In this section, we discuss the model choices and the resulting complexity analysis for the proposed multi-objective approach to LFI, as illustrated in Algorithm 1.

3.3.1 Model choices for surrogates

To meet the requirements for the surrogates as stated in Sects. 3.1 and 3.2, we have chosen a linear model of coregionalization (LMC) (Fanshawe and Diggle 2012) for discrepancies and a Bayesian neural network (BNN) (Kononenko 1989; Esposito 2020) for state transition dynamics.

- Linear model of coregionalization.** LMC is one of the simplest multi-objective models. It expresses each of its L outputs f_l as a linear combination $f_l(\theta_*) = \sum_{q=1}^Q a_{l,q} u_q$, as shown in Fig. 3, where the $u_q \sim GP(0, v(\theta_*))$ are latent GPs and the $a_{l,q}$ are linear coefficients that need to be solved.
- Bayesian neural network.** BNN can be represented as an ensemble of neural networks, where each has its own weights $\omega^{(h)}$ drawn from a shared, learned probability distribution (Blundell et al. 2015) with $\omega^{(h)} \sim \mathcal{N}(\mu^{(h)}, \log(1 + \chi^{(h)}))$, where $\mu^{(h)}$ and $\chi^{(h)}$ are the hyperparameters that need to be learned. Previously, neural networks have been successfully applied in SSM settings for either modelling the transition dynamics or the observation model (Rivals and Personnaz 1996; Bonatti and Mohr 2021).

3.3.2 Complexity analysis

Given the aforementioned model choices, the resulting computational complexity of Algorithm 1 is primarily influenced by three main stages: training the multi-objective surrogate $\tilde{\delta}_{\theta}$, extracting the posterior from discrepancy surrogates (Eq. 1) and training the transition dynamics model \tilde{h}_{θ_t} . Both LMC and BNN are trained by minimising the variational evidence lower bound (see more details in ‘‘Appendix C’’).

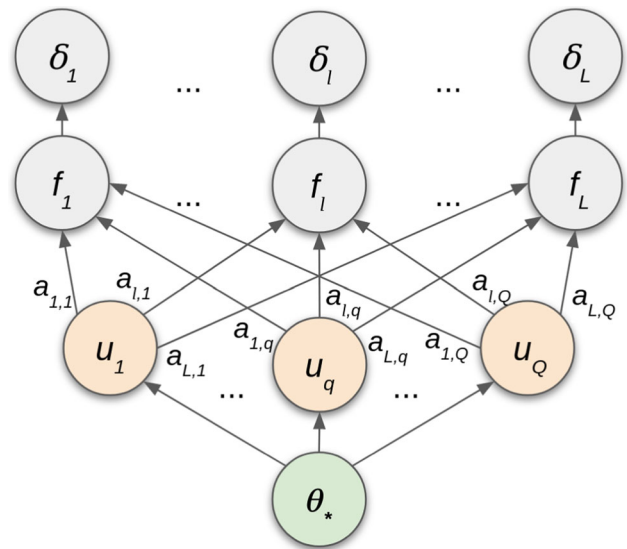


Fig. 3 Graphical representation of the LMC. The discrepancy outputs $\delta_l \equiv \delta_l(\theta_*)$ are modelled as a linear combination of latent functions u_q . The model shares the same parameter values θ_* between all objectives

- Training the multi-objective surrogate.** The cost of training $\tilde{\delta}_{\theta}$ depends on the number of synthetic observations $|\mathcal{S}|$ (the cardinality of \mathcal{S}), on the size of the moving window L and on the user-specified number M of inducing points (Alvarez and Lawrence 2011) for the LMC. This results in a complexity $\mathcal{O}(|\mathcal{S}|LM^2)$, compared to $\mathcal{O}(|\mathcal{S}|M^2)$ for traditional GPs used in BOLFI.
- Posterior extraction.** This stage consists of finding the appropriate ϵ (e.g., by minimising the GP mean function) and then applying Eq. (1). The complexity of this step is bounded by the calculation of the variance of the surrogate for each of the I samples from the posterior over states, resulting in $\mathcal{O}(LM^2I)$.
- Training the transition dynamics model.** When employing variational inference (Zhang et al. 2018) to train the transition dynamics model \tilde{h}_{θ_t} , the computational cost is linear in the number W of BNN parameters, resulting in $\mathcal{O}(WKES_p)$. Here, K represents the overall amount of training data for \tilde{h}_{θ_t} , E is the number of epochs, and S_p is the number of parameter samples from the posterior distribution that is required to obtain the distribution of outputs.

Depending on the choice of hyperparameters, the computational complexity of Algorithm 1 is bounded by either $\mathcal{O}(|\mathcal{S}|LM^2)$, $\mathcal{O}(LM^2I)$ or $\mathcal{O}(WKES_p)$. Most of these parameters are common in LFI (e.g., $|\mathcal{S}|$, I), and the rest are specific to surrogate choices, which can be replaced with fewer-parameter alternatives if needed. We provide recommendations for choosing these hyperparameters in ‘‘Appendix C’’.

3.4 Theoretical properties

In this section, we analyse the convergence properties and limitations of our LFI method for state-space models. We discuss how our method approximates states and transition dynamics, and outline the restrictions imposed by our choice of models on the class of systems that can be effectively modelled using our method. While the approach discussed in this section provides a robust framework for state inference in SSMs, it is vital to note that the results are approximate in nature, owing to inherent limitations such as the finite moving window. This section primarily aims to lay out the conditions under which our method can be seen as offering a good approximation rather than an exact solution.

3.4.1 Convergence

In the convergence analysis, we examine the ability of our method to learn a suitable approximation of states and transition dynamics when provided with sufficient data. The state approximations for $p(\theta_t | \mathbf{x}_t)$ are obtained through the likelihood function in Eq. (1), which Proposition 1 of Gutmann and Corander (2016) identifies as a non-parametric approximation of the true likelihood:

Proposition 1 *Maximising the synthetic log-likelihood $\log \mathcal{L}(\theta_*)$ in Eq. (1) corresponds to maximising a lower bound of a non-parametric approximation of the log likelihood when the kernel function $\kappa_\epsilon(\cdot)$ is convex.*

$$\log \mathcal{L}(\theta_*) \geq \log \kappa_\epsilon(\mathbb{E}[\delta_t(\theta_*)])$$

For our LMC model, we can demonstrate that Proposition 1 holds when the kernel is a Gaussian CDF, as specified below and ϵ is the minimum of the GP surrogate mean function:

Corollary 1 *Assuming the Gaussian CDF kernel $F((\epsilon - \mu(\theta_*))/\sqrt{v(\theta_*) + \sigma^2})$ from Sect. 2 and $\epsilon = \min_{\theta_*} \mu(\theta_*)$, Proposition 1 holds for the LMC model of discrepancy.*

Proof The Gaussian CDF kernel $F(\cdot)$ is known to be convex on the interval $(-\infty, 0]$. By setting ϵ as the minimum of the GP surrogate mean function, the argument of $F(\cdot)$ is restricted to the range $(-\infty, 0]$ with the maximum at 0 (note that since $\mu(\cdot)$ models discrepancy, it is always non-negative). Consequently, the inequality expression in Proposition 1 is preserved, while Jensen’s inequality ensures a lower bound for both $\mathcal{L}(\theta_*)$ and its logarithm when the functions are convex. \square

As for the approximations of state transitions $p(\theta_{t+1} | \theta_t)$, their convergence follows from the universal approximation theorem of neural networks Hornik et al. (1989). This theorem states that every continuous function can be approximated by a neural network with a single hidden layer of

neurons whose transfer function is bounded. Our use of the BNN model for transition dynamics compiles with this theorem. Under certain conditions, such as the availability of sufficient parameters and data, the central limit theorem guarantees that the expectation of our approximation converges to the target distribution.

3.4.2 Restrictions on modelling classes

Our choice of models imposes additional limitations on the class of systems that may be challenging to model using our method. The first limitation concerns high predictive variance when learning systems with long-term dependencies. While our method is robust across a variety of applications, it encounters challenges when dealing with time series that possess long memory. If the size of the moving window is shorter than the memory inherent in the time series, our method may fail to capture crucial long-term dependencies. Although it can adeptly handle abrupt changes to a certain extent, effectively addressing the complexities presented by long-memory dynamics in time series is a topic for future development. Given that the training of the BNN involves a single trajectory consisting of a limited number of observations (50 in our experiments from Sect. 4), the flexibility offered by BNNs might be insufficient to model systems characterised by both non-linear dynamics and significant long-term dependencies accurately. It is important to note, however, that BNNs do not introduce additional theoretical restrictions on the class of systems that can be modelled.

The second limitation concerns the type of observation distribution that our method models through the LMC. Although LMCs offer greater flexibility than vanilla GPs, they may have difficulty modelling asymmetric, skewed, or multimodal noise in the observation model when the simulation budget is constrained. This issue is prevalent among various LFI methods, as they often rely on models, such as GP-based surrogates, that make simplifying assumptions, such as assuming Gaussian noise. These assumptions can compromise the reliability of state posterior approximations when they are violated.

The third limitation stems from using GPs in LFI, which are subject to the curse of dimensionality, restricting the observation model’s dimensionality to fewer than 10. This constraint, however, is intrinsically connected to our method’s sample-efficiency, a significant advantage, as it requires only a few synthetic observations to approximate the likelihood. If the simulation budget for the observation model is not limited to the order of a hundred simulations, we recommend using more complex surrogates, such as SNEs or LANs from Table 1, alongside our approach to modelling state transitions.

4 Experiments

We assess the quality of our method for state inference and prediction tasks in a series of SSM experiments, where a simulator serves as the observation model g_{θ} . In the experiments, our method uses the surrogate choices of LMC and BNN, as described in Sect. 3.3. We demonstrate that it can accurately learn state transition dynamics and improve upon existing LFI methods for the state inference task. Moreover, we investigate the sample-efficiency of the proposed method and demonstrate its effectiveness in non-stationary user modelling case studies. We compare our method against traditional SSM methods in cases with available closed-form likelihoods and against LFI methods when only a simulator is available and traditional methods cannot be applied.

4.1 Experimental setup

We simulated time series of observations based on single-sampled trajectories from ground-truth transition dynamics (available for evaluation purposes but unknown to the methods) of five SSMs, described in Sect. 4.2. Our goal was to estimate the simulator parameters that likely produced these observations, and learn the model of transition dynamics for state prediction based on the sampled trajectory.

4.1.1 Comparison methods

For the state inference task, we compare the quality of state estimates by our approach against other LFI methods: BOLFI (Gutmann and Corander 2016), SNPE (Papamakarios and Murray 2016), SNLE (Papamakarios et al. 2019), and SNRE (Durkan et al. 2020). We use a fixed simulation budget for all these methods, with 20 simulations to initialise the models and then two additional simulations for each new time-step. For the SNE approaches (SNPE, SNLE, and SNRE), we provided all simulations at once since that is their intended mode of operation. As for the prediction task, we sampled state trajectories from the transition model and evaluated them against trajectories from ground-truth dynamics. We performed these experiments in SSMs with simulators that have tractable likelihoods $p(\mathbf{x}_t | \theta_t)$, providing the closed-form of the ground-truth likelihoods to the state-of-the-art SSM inference methods GP-SSM (Frigola et al. 2014; Ialongo et al. 2019) and PR-SSM (Doerr et al. 2018), while our method was still doing LFI. For all methods in the prediction task, we provided 50 observations and then sampled trajectories that had the same length of 50 time-steps.

We also compared two variants of our method that differ only in the way the next simulations are sampled: LMC-BLR, where samples were taken from Bayesian linear regression (BLR) models that linearized the transition dynamics along 50 observed time-steps; and LMC-qEHVI, where a popular

acquisition function for multitask BO, q-expected hypervolume improvement (qEHVI) (Daulton et al. 2020), was used to provide samples. The role of these variants was to evaluate how the choice of future simulations impacts the quality of state inference and prediction.

All models were assessed in terms of the root mean squared error (RMSE) between the state estimates and their ground-truths. The experiments were repeated 30 times with different random seeds. Additional details on the implementation of the methods can be found in “Appendix C”; all code for replicating the experiments is included in the Supplement.

4.2 The state-space models

In this section, we present two case studies with non-stationary user models and three SSMs with tractable likelihoods. In user modelling experiments, we simulated behavioural data from humans who completed a certain task in two different experiments, described in Sects. 4.2.1 and 4.2.2. It is worth noting that unlike typical experiments, where the underlying dynamics can be assumed to be stationary, user preferences and behaviour can change over time, making them harder to model with traditional approaches. For the first task, the user evaluated dataset embeddings for a classification problem, and the evaluation score was used as behavioural data. During the second task, the user searched for a target on a display, and the search time was measured. Our task in the experiments was to track the changing parameters of user models and learn their dynamics.

In addition to non-stationary user models, we also experimented with three models with tractable likelihoods, common in SSM literature: linear Gaussian (LG), non-linear non-Gaussian (NN), and stochastic volatility (SV) models. In the LG model, the state transition dynamics and observation model are both linear, with high observational white noise. The NN model is a popular non-linear SSM Kitagawa (1996), where each observation has two unique solutions. Lastly, we used the SV model Barndorff-Nielsen and Shephard (2002), which is used for predicting the volatility of asset prices in stock markets (Taylor 1994; Shephard 1996). For in-depth insights into these models and a detailed report on the *auto-correlation function* (Parzen 1963; Brockwell and Davis 2009) across all five SSMs, shedding light on the intricacies of their transition dynamics, please refer to “Appendix B”.

4.2.1 UMAP parameterisation

In this study, we utilise the first non-stationary user model to observe and analyse how an individual’s preferences evolve over time while interacting with data. Consider a scenario where an individual is engaged in data categorization without prior familiarity with the data. Initially, their primary focus

might be on exploring and understanding the data. However, as they become acquainted with the data, their attention shifts towards enhancing the accuracy of their categorizations. By employing uniform manifold approximation and projection (UMAP) (McInnes et al. 2018a), we adapt the data presentation to align with the user’s shifting needs. The ultimate objective is to accurately predict and apply the most suitable UMAP settings for the user at different times.

User modelling

Drawing insights from cognitive science (Slovic et al. 2002; Lichtenstein and Slovic 2006), it is evident that individuals can possess preferences over various data presentations, even if they are unable to articulate their specific desires explicitly.

We quantify these preferences through an evaluation function, which assigns scores to different presentations of handwritten digit data (Alpaydin and Kaynak 1998). The assigned scores are influenced by two primary metrics: the density-based cluster validity (DBCv) score ($\mathcal{U}(\cdot)$) (Moulavi et al. 2014) and the c-support vector classification (SVC) accuracy ($\mathcal{P}(\cdot)$) (Boser et al. 1992; Cortes and Vapnik 1995). A weight parameter, w_t , adjusts the balance between these metrics, thereby allowing emphasis on either data exploration (\mathcal{U}) or classification accuracy (\mathcal{P}):

$$\delta_t(\theta_{*,t}) = (1 - w_t) \cdot \mathcal{U}(\theta_{*,t}) + w_t \cdot \mathcal{P}(\theta_{*,t}), \tag{3}$$

$$w_t = \frac{1}{1 + e^{-0.1 \cdot (t-25)}}. \tag{4}$$

Both $\mathcal{U}(\cdot)$ and $\mathcal{P}(\cdot)$ are dependent on the time-variant UMAP settings, denoted as $\theta_{*,t}$.

State-space modelling framework

- **Observation model:** The observation model is constituted by the combination of the UMAP algorithm and the evaluation function (as defined in Eq. (3)). While the UMAP algorithm provides a low-dimensional data representation (or embedding), the parameters of which are dictated by the latent states, the evaluation function serves as a subjective lens through which data presentations are assessed. Subsequently, the scores of the evaluation function serve as observations. To cater to evolving human preferences, adjustments to the UMAP settings are needed.
- **Transition dynamics:** In our framework, transition dynamics are unknown; we lack precise knowledge regarding the necessary adjustments to the UMAP parameters since the evaluation function is beyond our direct control.
- **Latent states:** The latent states within our model are represented by the time-dependent UMAP parameters: $\theta_{*,t} = \{\theta_{d,t}, \theta_{dist,t}, \theta_{n,t}\}$. These include the dimension of the reduced space (θ_d), the point density-dictating parameter (θ_{dist}), and the neighbourhood size parameter

for local metric approximation (θ_n). The priors for these parameters are:

$$\theta_{d,t} \sim \text{Unif}(1, 64) \in \mathbb{Z},$$

$$\theta_{dist,t} \sim \text{Unif}(0, 0.99) \in \mathbb{R},$$

$$\theta_{n,t} \sim \text{Unif}(2, 200) \in \mathbb{Z}.$$

It is crucial to note that the ideal UMAP settings (referred to as *ground truth states*) remain elusive for our task. To address this, we generated 1,500,000 embeddings using parameter settings sampled from the prior, calculated their corresponding \mathcal{U} and \mathcal{P} values, and retained only a very small number of parameter settings (0.06%) exhibiting the best preference score values at each time step. Subsequently, we applied a Gaussian kernel density estimator to these parameter settings, allowing us to derive estimates of the ground truth for evaluating the performance of the methods. More implementation details can be found in ‘‘Appendix C’’.

4.2.2 Eye movement control for gaze-based selection

In our next study, we seek to delve deeper into the mechanics of human eye movement during target-search tasks on a two-dimensional (2D) screen, as previously explored by (Zhang et al. 2010; Schuetz et al. 2019). The study observes individuals as they engage in repeated tasks, noticing an improvement in their task performance as they develop beliefs about the target location. However, fatigue sets in over time, leading to increased latency between eye movements. This latency masks the true individual characteristics of the human gaze, like ocular motor noise and peripheral vision’s spatial noise, making it challenging to identify consistent gaze features as fatigue develops. Our goal is to create a robust model that accurately depicts eye movement latency, helping identify consistent gaze features amidst signs of fatigue.

User modelling

The user model serves as a surrogate for human behaviour within a simulated environment, aiming to locate a target on a 2D screen. This environment comprises:

- **Reinforcement learning agent:** Acting as a surrogate for the human subject, the agent learns to locate and focus on a target on a 2D display. The agent’s training uses ground truth values for ocular motor noise ($\theta_{om,t}$), spatial noise of peripheral vision ($\theta_{s,t}$), and varied values for eye movement latency ($\theta_{l,t}$). Given that both actions and observations experience noise from $\theta_{om,t}$ and $\theta_{s,t}$, the agent needs multiple attempts to locate the target accurately:

$$x_t = \sum_e (2.7 \cdot \hat{A}^{(e)}(\theta_{om}, \theta_s) + \theta_{l,t}). \tag{5}$$

where $\hat{A}^{(e)}$ represents the eye movement amplitude function, e is the eye movement index, and x_t represents the total time recorded for the agent to locate the target. The training process, spanning 10,000 episodes, utilises a multilayer perceptron policy derived from the proximal policy optimisation (PPO) algorithm (Schulman et al. 2017).

- **Virtual environment:** Within this simulated space, the agent acts, observes, and updates its beliefs, with all elements (including the target's location) represented by coordinates c_1, c_2 in the range of -1 to 1 . As the agent assimilates noisy target observations, it adjusts its gaze and updates its belief system through a predefined mechanism. Once the agent's gaze aligns with the target location, the task is deemed complete. This virtual environment was created by Chen et al. (2021) using the Open AI gym framework (Brockman et al. 2016).

State-space modelling framework

- **Observation model:** The simulation that drives the eye movement task functions as the observation model, detailed in Eq. (5). However, due to the inherent complexity and opaque characteristics of the reinforcement learning components used within, it remains ambiguous how the parameters of the reinforcement learning policy (or the latent states) influence the time it takes for the agent to locate the target x_t . These timings x_t subsequently form the observations for the SSM.
- **Transition dynamics:** Our experiments assume a specific form for eye movement latency's transition dynamics, which remains unknown to the methods:

$$\theta_{l,t} = 12 \cdot \log(t + 1) + 37. \quad (6)$$

It is crucial to note that while latency changes, the way it influences the user model's policy remains elusive, making the inference of other latent states non-trivial.

- **Latent states:** The properties of human gaze behaviour serve as the latent states in our SSM, denoted by $\theta_{*,t} = \{\theta_{om,t}, \theta_{s,t}, \theta_{l,t}\}$. These parameters, which also shape the reinforcement learning policy, have the following prior distributions:

$$\begin{aligned} \theta_{om,t} &\sim \text{Unif}(0, 0.2) \in \mathbb{R}, \\ \theta_{s,t} &\sim \text{Unif}(0, 0.2) \in \mathbb{R}, \\ \theta_{l,t} &\sim \text{Unif}(30, 60) \in \mathbb{R}. \end{aligned}$$

For generating observed data, the observation model utilised ground truth values of 0.01 for $\theta_{om,t}$ and 0.09 for $\theta_{s,t}$ across all t . The $\theta_{l,t}$ value, however, was changing according to the aforementioned transition dynamics. Detailed infor-

mation about the implementation is available in "Appendix C".

4.3 Results and analysis

The results for the inference and prediction tasks are presented in Tables 2 and 3, respectively. The lower the RMSE, the better the quality of estimation. In the inference task, the proposed LMC-based methods clearly outperformed the BOLFI and SNE approaches. This indicates that considering multiple objectives at the same time was beneficial for state inference and that the model actually leverages information from consecutive states without hindering performance. Additionally, it can be seen that all LMC-based variants performed differently, which can only be attributed to how the next simulations were chosen since the surrogate was exactly the same in all three methods. As the results show, having BNN as a model for state transition was beneficial for experiments with non-stationary user models, while having BLR was more preferable for simpler models. This suggests that BLR is expressive enough to replicate simple transitions but struggles with more complex ones, for which BNN is more suitable.

4.3.1 Learning transition dynamics

The comparisons with GP-SSMs and PR-SSMs for learning transition dynamics show that our method learns accurate dynamics, or, at least, relative to the SSM method baselines. The SSM methods showed worse results than the BLR and BNN approaches. This can be explained by the lack of observations for learning state transitions by the SSM methods, which also explains the high variance in the sampled trajectories from these methods. As for comparisons between BLR and BNN, BLR performs better only in LG and SV models, while BNN performs better in more complex case studies. Moreover, it should be noted that trajectory sampling from BLR is possible only by retaining all local linearizations of the dynamics, which is a far more limiting approach than having one single model. Therefore, BNN is a more preferable transition dynamics model.

4.3.2 Empirical time costs

The empirical time costs for running the LFI methods are shown in Table 4. It can be seen that the SNPE method was the fastest for the computationally cheap simulators (of SSMs with tractable likelihoods), while the LMC-qEHVI required the least amount of time for the non-stationary user models. This is expected since the SNEs learn the model only once and then simply use it for all observations, which is suitable for the computationally cheap simulations with simple LFI solutions. However, for non-stationary user models, where

Table 2 Comparison of LFI methods (rows) in different SSMs (columns) for the state inference task

Methods	LG	NN	SV	UMAP	Gaze
LMC-BNN	1.77 ± 0.12	6.92 ± 0.51	16.14 ± 3.27	58.24 ± 3.62	58.7 ± 5.4
LMC-BLR	1.3 ± 0.1	6.86 ± 0.54	13.15 ± 3.25	59.19 ± 3.31	60.6 ± 5.8
LMC-qEHVI	1.5 ± 0.1	7.03 ± 0.55	24.4 ± 2.5	64.96 ± 3.72	56.9 ± 4.5
BOLFI	1.55 ± 0.15	7 ± 0.6	27.35 ± 3.45	84.31 ± 3.54	73.45 ± 5.75
SNPE	7.15 ± 0.65	18.2 ± 0.93	77.4 ± 3.1	74.13 ± 3.21	68.1 ± 7.8
SNLE	6 ± 0.5	10.35 ± 0.64	54.25 ± 2.45	71.45 ± 3.44	77.25 ± 4.05
SNRE	10.4 ± 1.7	17.93 ± 1.34	57.15 ± 2.35	75.85 ± 1.26	80.75 ± 1.35

The performance was measured with 95% confidence interval (CI) of the RMSE between estimated and ground truth state points for 50 time-steps. The best results in each column are highlighted in bold

Table 3 Comparison of transition dynamics models (rows) in different SSMs (columns)

Methods	LG	NN	SV	UMAP	Gaze
LMC-BNN	210 ± 4	148 ± 2	117 ± 21	1394 ± 27	1365 ± 3
LMC-BLR	64 ± 7	154 ± 4	100 ± 37	1409 ± 49	1372 ± 3
BNN base	103 ± 22	51 ± 14	24 ± 6	–	–
GP-SSM	284 ± 71	2204 ± 1111	3206 ± 1175	–	–
PR-SSM	253 ± 68	610 ± 510	1378 ± 740	–	–

The performance was measured with 95% CI of the RMSE between sampled and ground truth trajectories of length 50. The best results in each column are highlighted in bold. The *BNN base* row evaluates the BNN transition model that was trained using samples from the true posterior, and it is used as a baseline for the transition dynamics performance

Table 4 Time comparison of LFI methods (rows) in different SSMs (columns) for training 50 time-steps

Methods	LG	NN	SV	UMAP	Gaze
LMC-BNN	87.6 ± 2.6	79.2 ± 1	81 ± 2.9	171.2 ± 5	408 ± 8
LMC-BLR	82.1 ± 5.5	48.2 ± 0.8	93.5 ± 4	149.4 ± 4.5	442.5 ± 8.8
LMC-qEHVI	25.5 ± 1.5	23.9 ± 0.5	24.7 ± 0.7	116 ± 4.6	347.2 ± 7.3
BOLFI	1.1 ± 0	1 ± 0	1.7 ± 0.1	129.3 ± 11.7	369.5 ± 16.5
SNPE	0.1 ± 0	0.1 ± 0	0.4 ± 0	168.6 ± 4.4	710.9 ± 415
SNLE	119.4 ± 2.9	137.3 ± 5.8	355.3 ± 23.1	582.7 ± 23.8	1003.4 ± 92.4
SNRE	34 ± 1.5	35.4 ± 0.3	110.2 ± 6.1	309.6 ± 8.6	446.4 ± 14

The running time is shown in minutes along with 95% CI. The best results in each column are highlighted in bold

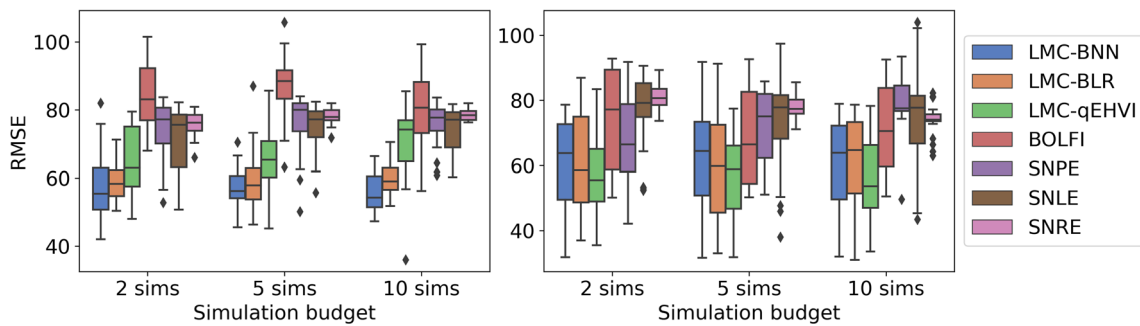


Fig. 4 The performance of LFI methods for state inference tasks with various simulation budgets in two non-stationary user modelling experiments. The box plots were computed from 30 repetitions with different

random seeds. The horizontal line on box plots shows the median; the bar shows the upper and lower quartiles; and the whiskers indicate the rest of the quartiles. The diamond points indicate outliers

there are no closed-form likelihoods available, learning a single model actually requires much more time. To summarise, the LMC variants are clearly preferable for the computationally heavy simulators, which dominate the cost of training a transition dynamics model and a multi-objective surrogate.

4.3.3 Simulation budget impact

Finally, Fig. 4 shows how the performance of the LFI models changes with different simulation budgets: 2, 5, and 10 simulations per time-step. As expected, in general, all methods improved their performance with increased budgets. However, there is little difference in how these methods compare with respect to each other. This indicates that the results are not sensitive to the precise simulation budget.

4.3.4 Key findings of experiments

In all experiments, we attribute the success of the proposed LMC-BNN method to a more flexible multi-output surrogate and a more efficient way of choosing simulation candidates. The LMC allows multi-fidelity modelling (e.g., decomposing a stochastic process into processes with different length-scales), which allows leveraging information from multiple consecutive time-steps, unlike standard GPs. At the same time, samples from the transition model provide better candidates for simulations than the alternatives. The flexible surrogate along with adaptive acquisition make our method particularly suitable for online settings, where only a handful of samples are possible per time-step.

While the experiments underline the strengths of our LMC-BNN method, it is equally important to be candid about its drawbacks. As highlighted earlier in this section, for more basic models, opting for BLR proves to be a more streamlined and beneficial approach than BNN. Moreover, as detailed in Sect. 4.3.2, in certain contexts, when the simulations are computationally cheap, SNPE methods outpace our method in terms of performance speed. For a comprehensive dive into the specific constraints and limitations of our modelling technique, please consult Sect. 3.4.2.

5 Discussion

We proposed an approach for state inference and prediction in the challenging SSM setting, where the transition dynamics are unknown and observations can only be simulated. Importantly, our model of transition dynamics was obtained with few simulations, making it suitable for cases with computationally expensive simulators. This is important because typically sample-efficient LFI approaches discard any temporal information from observed time-series and cannot do state prediction, which is necessary for choosing the next

simulation when the simulation budget is limited. We proposed a solution for both of these challenges: we use a multi-objective surrogate model for the discrepancy measure between observed and synthetic data, which connects the consecutive states through shared parameters, and we train an additional surrogate for state transitions with samples from LFI state posteriors. Additionally, our method does not restrict the family of admissible solutions for the state transitions to being linear or Gaussian, unlike existing LFI methods for SSMs (Jasra et al. 2012; Martin et al. 2014), making it more widely applicable.

Although our method uses a more flexible surrogate for the LFI of states, we demonstrated that it requires neither additional data nor significantly more training time than traditionally used GP surrogates. We reached the sample-efficiency goal by sharing synthetic observations across all discrepancy objectives, allowing the method to use the same simulations indefinitely. As for the decreased training time, we proposed a moving window approach that allowed the surrogate to focus only on a few recent SSM time-steps at a time. In conclusion, having a more flexible surrogate improved state inference and provided better samples from state posteriors for learning the unknown dynamics.

The main limitation of our approach is that the proposed transition dynamics model does not account for long-term state dependencies. Our state transition surrogate considers only the most recent state as an input, assuming the Markov property, and therefore cannot forecast far into the future. The resulting predictions have a very low variance and a tendency to converge to similar values, which is expected when training on a single trajectory. Despite this limitation, our method remains effective in cases where the observation model serves as the primary source of information, while the transition dynamics model still plays a complementary role in state inference.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s11222-023-10339-8>.

Author Contributions All authors contributed to the research and development of the paper. Specifically, A.A. and S.K. conceptualised the original idea for the study. A.A. and T.T. were responsible for the implementation of the proposed method, with A.A. conducting the subsequent experiments and preparing the paper's first draft. H.P., A.H., and S.K. provided feedback on the methodological contributions at various stages of the manuscript's development. The design and planning of the case studies were carried out in a joint effort by A.A., A.H., and S.K. Finally, all authors took part in the review and approval of the final version of the manuscript.

Funding Open Access funding provided by Aalto University. This work was supported by the Academy of Finland (Flagship programme: Finnish Center for Artificial Intelligence FCAI; grants 328400, 319264, 292334) and UKRI Turing AI World-Leading Researcher Fellowship EP/W002973/1. HP was also supported by European Research Council grant 742158 (SCARABEE, Scalable inference algorithms for Bayesian

evolutionary epidemiology). Computational resources were provided by the Aalto Science-IT Project.

Data availability Not applicable.

Code Availability Additional experiments and implementation details of simulators can be found in the Supplement. All code is available through the link: <https://github.com/AaltoPML/LFI-in-SSMs-with-UD>

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

Ethics approval Not applicable.

Consent for publication All authors have agreed to submit the manuscript to the journal.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

Alpaydin, E., Kaynak, C.: Cascading classifiers. *Kybernetika* **34**(4), 369–374 (1998)

Alsing, J., Wandelt, B., Feeney, S.: Massive optimal data compression and density estimation for scalable, likelihood-free inference in cosmology. *Mon. Not. R. Astron. Soc.* **477**(3), 2874–2885 (2018)

Alvarez, M.A., Lawrence, N.D.: Computationally efficient convolved multiple output Gaussian processes. *J. Mach. Learn. Res.* **12**, 1459–1500 (2011)

Anderson, B.D., Moore, J.B.: Optimal filtering. Courier Corporation (2012)

Andrei, N.: Scaled conjugate gradient algorithms for unconstrained optimization. *Comput. Optim. Appl.* **38**(3), 401–416 (2007)

Aushev, A., Pesonen, H., Heinonen, M., Corander, J. Kaski, S.: Likelihood-free inference with deep Gaussian processes. [arXiv:2006.10571](https://arxiv.org/abs/2006.10571) (2020)

Balandat, M., Karrer, B. Jiang, D. R., Daulton, S., Letham, B., Wilson, A. G., Bakshy, E.: BoTorch: a framework for efficient Monte-Carlo Bayesian optimization. *Adv. Neural Inf. Process. Syst.* **33** (2020)

Barndorff-Nielsen, O.E., Shephard, N.: Econometric analysis of realized volatility and its use in estimating stochastic volatility models. *J. R. Stat. Soc. Ser. B* **64**(2), 253–280 (2002)

Barthelmé, S., Chopin, N.: Expectation propagation for likelihood-free inference. *J. Am. Stat. Assoc.* **109**(505), 315–333 (2014)

Beaumont, M.A.: Approximate Bayesian computation in evolution and ecology. *Annu. Rev. Ecol. Evol. Syst.* **41**, 379–406 (2010)

Beaumont, M.A., Zhang, W., Balding, D.J.: Approximate Bayesian computation in population genetics. *Genetics* **162**(4), 2025–2035 (2022)

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

Bertorelle, G., Benazzo, A., Mona, S.: ABC as a flexible framework to estimate demography over space and time: some cons, many pros. *Mol. Ecol.* **19**(13), 2609–2625 (2010)

Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: Weight uncertainty in neural network. In: International Conference on Machine Learning, pp. 1613–1622. PMLR (2015)

Bonatti, C., Mohr, D.: One for all: Universal material model based on minimal state-space neural networks. *Sci. Adv.* **7**(26), eabf3658 (2021)

Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, pp. 144–152 (1992)

Brochu, E., Cora, V.M., De Freitas, N.: A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. [arXiv:1012.2599](https://arxiv.org/abs/1012.2599) (2010)

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., Zaremba, W.: OpenAI gym (2016)

Brockwell, P.J., Davis, R.A.: Time Series: Theory and Methods. Springer science & business media, New York (2009)

Byrd, R.H., Lu, P., Nocedal, J., Zhu, C.: A limited memory algorithm for bound constrained optimization. *SIAM J. Sci. Comput.* **16**(5), 1190–1208 (1995)

Cafisch, R.E., et al.: Monte Carlo and quasi-Monte Carlo methods. *Acta Numer* **1–49**, 1998 (1998)

Calvet, L.E., Czellar, V.: Accurate methods for approximate Bayesian computation filtering. *J. Financ. Economet.* **13**(4), 798–838 (2015)

Campello, R.J., Moulavi, D., Sander, J.: Density-based clustering based on hierarchical density estimates. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 160–172. Springer (2013)

Chen, X., Acharya, A., Oulasvirta, A.: An adaptive model of gaze-based selection. In: CHI Conference on Human Factors in Computing Systems (CHI’21). Association for Computing Machinery (2021)

Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)

Cranmer, K., Brehmer, J., Louppe, G.: The frontier of simulation-based inference. *Proc. Natl. Acad. Sci.* **117**(48), 30055–30062 (2020)

Csilléry, K., Blum, M.G., Gaggiotti, O.E., François, O.: Approximate bayesian computation (abc) in practice. *Trends Ecol. Evolut.* **25**(7), 410–418 (2010)

Daulton, S., Balandat, M., Bakshy, E.: Differentiable expected hypervolume improvement for parallel multi-objective Bayesian optimization. [arXiv:2006.05078](https://arxiv.org/abs/2006.05078) (2020)

Dean, T.A., Singh, S.S., Jasra, A., Peters, G.W.: Parameter estimation for hidden Markov models with intractable likelihoods. *Scand. J. Stat.* **41**(4), 970–987 (2014)

Doerr, A., Daniel, C., Schiegg, M., Nguyen-Tuong, D., Schaal, S., Toussaint, M., Trimpe, S.: Probabilistic recurrent state-space models. [arXiv:1801.10395](https://arxiv.org/abs/1801.10395) (2018)

Doucet, A., De Freitas, N., Gordon, N.J., et al.: Sequential Monte Carlo Methods in Practice, vol. 1. Springer, New York (2001)

Durkan, C., Murray, I., Papamakarios, G.: On contrastive learning for likelihood-free inference. In: International Conference on Machine Learning, pp. 2771–2781. PMLR (2020)

Errico, R.M., Yang, R., Privé, N.C., Tai, K.-S., Todling, R., Sienkiewicz, M.E., Guo, J.: Development and validation of observing-system simulation experiments at NASA’s global modeling and assimilation office. *Q. J. R. Meteorol. Soc.* **139**(674), 1162–1178 (2013)

- Esposito, P.: BLiTZ - Bayesian layers in Torch zoo (a Bayesian deep learning library for torch). <https://github.com/piEsposito/blitz-bayesian-deep-learning/> (2020)
- Fanshawe, T.R., Diggle, P.J.: Bivariate geostatistical modelling: a review and an application to spatial variation in radon concentrations. *Environ. Ecol. Stat.* **19**(2), 139–160 (2012)
- Fengler, A., Govindarajan, L.N., Chen, T., Frank, M.J.: Likelihood approximation networks (lans) for fast inference of simulation models in cognitive neuroscience. *Elife* **10**, e65074 (2021)
- Fiske, S.T., Taylor, S.E.: *Social Cognition: From Brains to Culture*. Sage (2013)
- Frigola, R., Chen, Y., Rasmussen, C.E.: Variational Gaussian process state-space models. *Adv. Neural. Inf. Process. Syst.* **27**, 3680–3688 (2014)
- Futrell, R., Gibson, E., Levy, R.P.: Lossy-context surprisal: An information-theoretic model of memory effects in sentence processing. *Cogn. Sci.* **44**(3), e12814 (2020)
- Georgiou, T., Demiris, Y.: Adaptive user modelling in car racing games using behavioural and physiological data. *User Model. User-Adap. Inter.* **27**(2), 267–311 (2017)
- Ghassemi, M., Wu, M., Hughes, M.C., Szolovits, P., Doshi-Velez, F.: Predicting intervention onset in the ICU with switching state space models. *AMIA Summits Transl. Sci. Proc.* **2017**, 82 (2017)
- Gimenez, O., Rossi, V., Choquet, R., Dehais, C., Doris, B., Varella, H., Vila, J.-P., Pradel, R.: State-space modelling of data on marked individuals. *Ecol. Model.* **206**(3–4), 431–438 (2007)
- Goncalves, P., Lueckmann, J.-M., Bassetto, G., Oecal, K., Nonnenmacher, M., Macke, J. H.: Flexible statistical inference for mechanistic models of neural dynamics. In: *Bonn Brain 3 Conference 2018*, Bonn, Germany (2018)
- GPY. GPY: A gaussian process framework in python. <http://github.com/SheffieldML/GPy> (2012)
- Greenberg, D., Nonnenmacher, M., Macke, J.: Automatic posterior transformation for likelihood-free inference. In: *International Conference on Machine Learning*, pp. 2404–2414. PMLR (2019)
- Gutmann, M.U., Corander, J.: Bayesian optimization for likelihood-free inference of simulator-based statistical models. *J. Mach. Learn. Res.* **17**(1), 4256–4302 (2016)
- He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
- He, S., Li, Y., Feng, Y., Ho, S., Ravanbakhsh, S., Chen, W., Póczos, B.: Learning to predict the cosmological structure formation. *Proc. Natl. Acad. Sci.* **116**(28), 13825–13832 (2019)
- Hermans, J., Begy, V., Louppe, G.: Likelihood-free MCMC with amortized approximate ratio estimators. In: *International Conference on Machine Learning*, pp. 4239–4248. PMLR (2020)
- Hoffman, M.D., Gelman, A.: The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *J. Mach. Learn. Res.* **15**(1), 1593–1623 (2014)
- Hornik, K., Stinchcombe, M., White, H.: Multilayer feedforward networks are universal approximators. *Neural Netw.* **2**(5), 359–366 (1989)
- Ialongo, A.D., Van Der Wilk, M., Hensman, J., Rasmussen, C.E.: Overcoming mean-field approximations in recurrent Gaussian process models. [arXiv:1906.05828](https://arxiv.org/abs/1906.05828) (2019)
- Izenman, A.J.: *An introduction to Kalman filtering with applications* (1988)
- Jasra, A., Singh, S.S., Martin, J.S., McCoy, E.: Filtering via approximate Bayesian computation. *Stat. Comput.* **22**(6), 1223–1237 (2012)
- Jeffrey, N., Alsing, J., Lanusse, F.: Likelihood-free inference with neural compression of DES SV weak lensing map statistics. *Mon. Not. R. Astron. Soc.* **501**(1), 954–969 (2021)
- Kahneman, D., Tversky, A.: Prospect theory: An analysis of decision under risk. In: *Handbook of the fundamentals of financial decision making: Part I*, pp. 99–127. World Scientific (2013)
- Kalman, R.E., et al.: Contributions to the theory of optimal control. *Boletín de la Sociedad Matemática* **5**(2), 102–119 (1960)
- Kalnay, E.: *Atmospheric Modeling, Data Assimilation and Predictability*. Cambridge University Press, Cambridge (2003)
- Karl, M., Soelch, M., Bayer, J., Van der Smagt, P.: Deep variational Bayes filters: Unsupervised learning of state space models from raw data. [arXiv:1605.06432](https://arxiv.org/abs/1605.06432) (2016)
- Kitagawa, G.: Monte Carlo filter and smoother for non-Gaussian nonlinear state space models. *J. Comput. Graph. Stat.* **5**(1), 1–25 (1996)
- Koller, D., Friedman, N.: *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, Cambridge (2009)
- Kononenko, I.: Bayesian neural networks. *Biol. Cybern.* **61**(5), 361–370 (1989)
- Lange, J.U., van den Bosch, F.C., Zentner, A.R., Wang, K., Hearin, A.P., Guo, H.: Cosmological evidence modelling: a new simulation-based approach to constrain cosmology on non-linear scales. *Mon. Not. R. Astron. Soc.* **490**(2), 1870–1878 (2019)
- Lichtenstein, S., Slovic, P.: *The Construction of Preference*. Cambridge University Press, Cambridge (2006)
- Lintusaari, J., Vuollekoski, H., Kangasrääsiö, A., Skytén, K., Järvenpää, M., Marttinen, P., Gutmann, M.U., Vehtari, A., Corander, J., Kaski, S.: ELFI: Engine for likelihood-free inference. *J. Mach. Learn. Res.* **19**(16), 1–7 (2018)
- Martin, G.M., McCabe, B.P., Frazier, D.T., Maneesoonthorn, W., Robert, C.P.: Auxiliary likelihood-based approximate Bayesian computation in state space models. *J. Comput. Graph. Stat.* **28**(3), 508–522 (2019)
- Martin, J.S., Jasra, A., Singh, S.S., Whiteley, N., Del Moral, P., McCoy, E.: Approximate Bayesian computation for smoothing. *Stoch. Anal. Appl.* **32**(3), 397–420 (2014)
- McInnes, L., Healy, J., Astels, S.: hdbscan: Hierarchical density based clustering. *J. Open Source Softw.* **10**, 1–12 (2017). <https://doi.org/10.21105/joss.00205>
- McInnes, L., Healy, J., Melville, J.: UMAP: Uniform manifold approximation and projection for dimension reduction. [arXiv:1802.03426](https://arxiv.org/abs/1802.03426) (2018a)
- McInnes, L., Healy, J., Saul, N., Grossberger, L.: UMAP: Uniform manifold approximation and projection. *J. Open Source Softw.* **3**(2019), 861 (2018)
- Melchior, S., Curi, S., Berkenkamp, F., Krause, A.: Structured variational inference in unstable Gaussian process state space models. [arXiv:1907.07035](https://arxiv.org/abs/1907.07035) (2019)
- Moulavi, D., Jaskowiak, P.A., Campello, R.J., Zimek, A., Sander, J.: Density-based clustering validation. In: *Proceedings of the 2014 SIAM International Conference on Data Mining*, pp. 839–847. SIAM (2014)
- Myung, I.J.: Tutorial on maximum likelihood estimation. *J. Math. Psychol.* **47**(1), 90–100 (2003)
- Ong, V.M.-H., Nott, D.J., Tran, M.-N., Sisson, S.A., Drovandi, C.C.: Likelihood-free inference in high dimensions with synthetic likelihood. *Comput. Stat. Data Anal.* **128**, 271–291 (2018)
- Owen, A.B.: Scrambling Sobol’ and Niederreiter-Xing points. *J. Complex.* **14**(4), 466–489 (1998)
- Papamakarios, G., Murray, I.: Fast ϵ -free inference of simulation models with Bayesian conditional density estimation. In: *Advances in Neural Information Processing Systems*, pp. 1028–1036. (2016)
- Papamakarios, G., Sterratt, D., Murray, I.: Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In: *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 837–848. PMLR (2019)
- Parzen, E.: On spectral analysis with missing observations and amplitude modulation. *Sankhyā Indian J. Stat. Ser. A* pp. 383–392 (1963)
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: PyTorch: an imperative style, high-performance deep learning library. [arXiv:1912.01703](https://arxiv.org/abs/1912.01703) (2019)

- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
- Peters, G.W., Sisson, S.A., Fan, Y.: Likelihood-free Bayesian inference for α -stable models. *Comput. Stat. Data Anal.* **56**(11), 3743–3756 (2012)
- Pothos, E.M., Chater, N.: A simplicity principle in unsupervised human categorization. *Cogn. Sci.* **26**(3), 303–343 (2002)
- Pritchard, J.K., Seielstad, M.T., Perez-Lezaun, A., Feldman, M.W.: Population growth of human y chromosomes: a study of y chromosome microsatellites. *Mol. Biol. Evol.* **16**(12), 1791–1798 (1999)
- Raffin, A., Hill, A., Ernestus, M., Gleave, A., Kanervisto, A., Dormann, N.: Stable baselines3. <https://github.com/DLR-RM/stable-baselines3> (2019)
- Reynolds, A.M., Rhodes, C.J.: The lévy flight paradigm: random search patterns and mechanisms. *Ecology* **90**(4), 877–887 (2009)
- Rivals, I., Personnaz, L.: Black-box modeling with state-space neural networks. In: *Neural Adaptive Control Technology*, pp. 237–264. World Scientific (1996)
- Rubin, D.B.: Bayesianly justifiable and relevant frequency calculations for the applied statistician. *Ann. Stat.* pp. 1151–1172 (1984)
- Salvatier, J., Wiecki, T.V., Fonnesbeck, C.: Probabilistic programming in Python using PyMC3. *PeerJ Comput. Sci.* **2**, e55 (2016)
- Schafer, C. M., Freeman, P. E.: Likelihood-free inference in cosmology: Potential for the estimation of luminosity functions. In: *Statistical Challenges in Modern Astronomy V*, pp. 3–19. Springer (2012)
- Schall, J.D.: Accumulators, neurons, and response time. *Trends Neurosci.* **42**(12), 848–860 (2019)
- Schuetz, I., Murdison, T. S., MacKenzie, K. J., Zannoli, M.: An explanation of Fitts’ law-like performance in gaze-based selection tasks using a psychophysics approach. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–13 (2019)
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017)
- Scott, D.W.: *Multivariate Density Estimation: Theory, Practice, and Visualization*. Wiley, New York (2015)
- Septier, F., Peters, G.W., Nevat, I.: Bayesian filtering with intractable likelihood using sequential MCMC. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6313–6317. IEEE (2013)
- Shafi, K., Latif, N., Shad, S.A., Idrees, Z., Gulzar, S.: Estimating option greeks under the stochastic volatility using simulation. *Phys. A* **503**, 1288–1296 (2018)
- Shephard, N.: Statistical aspects of ARCH and stochastic volatility. *Monograph. Stat. Appl. Probab.* **65**, 1–68 (1996)
- Sisson, S.A., Fan, Y., Beaumont, M.: *Handbook of Approximate Bayesian Computation*. CRC Press (2018)
- Slovic, P., Finucane, M., Peters, E., MacGregor, D.G.: Rational actors or rational fools: Implications of the affect heuristic for behavioral economics. *J. Socio-Econ.* **31**(4), 329–342 (2002)
- Smidl, V., Quinn, A.: Variational Bayesian filtering. *IEEE Trans. Signal Process.* **56**(10), 5020–5030 (2008)
- Smith, A.: *Sequential Monte Carlo Methods in Practice*. Springer Science & Business Media, New York (2013)
- Srinivas, N., Krause, A., Kakade, S.M., Seeger, M.: Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv:0912.3995* (2009)
- Sunnåker, M., Busetto, A.G., Numminen, E., Corander, J., Foll, M., Dessimoz, C.: Approximate Bayesian computation. *PLoS Comput. Biol.* **9**(1), e1002803 (2013)
- Sutskever, I., Martens, J., Dahl, G., Hinton, G.: On the importance of initialization and momentum in deep learning. In: *International Conference on Machine Learning*, pp. 1139–1147. PMLR (2013)
- Tavaré, S., Balding, D.J., Griffiths, R.C., Donnelly, P.: Inferring coalescence times from DNA sequence data. *Genetics* **145**(2), 505–518 (1997)
- Taylor, S.J.: Modeling stochastic volatility: A review and comparative study. *Math. Financ.* **4**(2), 183–204 (1994)
- Tejero-Cantero, A., Boelts, J., Deistler, M., Lueckmann, J.-M., Durkan, C., Gonçalves, P.J., Greenberg, D.S., Macke, J.H.: SBI: a toolkit for simulation-based inference. *J. Open Source Softw.* **5**(52), 2505 (2020). <https://doi.org/10.21105/joss.02505>
- Toni, T., Welch, D., Strelkowa, N., Ipsen, A., Stumpf, M.P.: Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *J. R. Soc. Interface* **6**(31), 187–202 (2009)
- Wieschen, E.M., Voss, A., Radev, S.: Jumping to conclusion? a lévy flight model of decision making. *Quant. Methods Psychol.* **16**(2), 120–132 (2020)
- Wilson, J.T., Hutter, F., Deisenroth, M.P.: Maximizing acquisition functions for Bayesian optimization. *arXiv:1805.10196* (2018)
- Zeng, X., Atlas, R., Birk, R.J., Carr, F.H., Carrier, M.J., Cucurull, L., Hooke, W.H., Kalnay, E., Murtugudde, R., Posselt, D.J., et al.: Use of observing system simulation experiments in the United States. *Bull. Am. Meteor. Soc.* **101**(8), E1427–E1438 (2020)
- Zerdali, E., Barut, M.: The comparisons of optimized extended Kalman filters for speed-sensorless control of induction motors. *IEEE Trans. Ind. Electron.* **64**(6), 4340–4351 (2017)
- Zhang, C., Bütepage, J., Kjellström, H., Mandt, S.: Advances in variational inference. *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(8), 2008–2026 (2018)
- Zhang, X., Ren, X., Zha, H.: Modeling dwell-based eye pointing target acquisition. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2083–2092 (2010)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.