Likelihood-Free Inference in State-Space Models with Unknown Dynamics

Anonymous Author(s)
Affiliation
Address
email

Abstract

We introduce a method for inferring and predicting latent states in the important and difficult case of state-space models (SSM) where observations can only be simulated, and transition dynamics are unknown. In this setting, the likelihood of observations is not available and only synthetic observations can be generated from a black-box simulator. We propose a way of doing likelihood-free inference (LFI) of states and state prediction with a limited number of simulations. Our approach uses a multi-output Gaussian process for state inference, and a Bayesian Neural Network as a model of the transition dynamics for state prediction. We improve upon existing LFI methods for the inference task, while also accurately learning transition dynamics. The proposed method is necessary for modelling inverse problems in dynamical systems with computationally expensive simulations, as demonstrated in experiments with non-stationary user models.

1 Introduction

Conventional likelihood-free inference (LFI) methods [1, 2] estimate parameters $\boldsymbol{\theta}$ of a static statistical model, given the dataset $\mathbf{x}^*$ and a black-box simulator $g_{\boldsymbol{\theta}}$. When the likelihood is intractable, these methods use synthetic datasets $\mathbf{x}^{(i)} \sim g_{\boldsymbol{\theta}}(\mathbf{x}|\boldsymbol{\theta})$ produced by the simulator to assist the inference. LFI has been successfully applied to identify parameters of complex real-world systems, such as financial markets, [3, 4, 5], species populations [6, 7, 8] and cosmology models [9, 10, 11]. Many real-world systems, however, are time-dependent, and can be described as a state-space model (SSM) [12, 13], where observed measurements $\mathbf{x}_t \in \mathbb{R}^n$ are emitted given a series of latent variables, the states, $\boldsymbol{\theta}_t \in \mathbb{R}^m$. When working with dynamical systems, one should account not only for an observation model, a simulator $g_{\boldsymbol{\theta}}$, but for Markovian state transition dynamics $h_{\boldsymbol{\theta}}$, as well, defined as $\boldsymbol{\theta}_{t+1} \sim h_{\boldsymbol{\theta}}(\boldsymbol{\theta}_{t+1}|\boldsymbol{\theta}_t)$. Therefore, traditional LFI methods, designed for static models, are ill-suited for dynamical systems, such as system simulation experiments in meteorology [14, 15], simulation-based forecasting in cosmology [16, 17] or adaptive modelling of human behaviour [18, 19].

Current LFI methods for dynamical systems [20, 21] proceed by assuming some simplified form of transition dynamics in an SSM. Unless the SSM is both linear and Gaussian, state values are difficult to estimate. This issue has been addressed in a large and growing set of methods, including extended Kalman filters (EKF) [22, 23, 24], GP-SSMs [25, 26], Bayes filtering (BF) [27, 28] and sequential Monte Carlo (SMC) methods [29, 30, 31]. Unfortunately, when the latent states have the additional requirement of being valid simulator parameters, these methods need a tractable likelihood that is not available when the observation model is black-box. At the same time, LFI has been applied to SSMs, but instead of focusing on the problem we address here, the work has focused on more efficient sampling-based methods [32], generation of a better matching statistics [33], on establishing theoretical convergence guarantees [34, 21, 35]. Despite these advances, inference with current LFI

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

methods in SSMs is limited in cases when h_t is non-linear and cannot be sampled or simplified, and when the simulators are computationally expensive.

In this paper, we introduce a method capable of obtaining likelihood-free state approximations and state predictions in discrete-time SSMs. We assume that the state transition model is unknown, can be non-linear and the number of samples from the observation model is limited, when simulators are computationally expensive. To solve the LFI problem in SSMs, we first need to estimate simulator parameters for observed measurements with as few samples as possible, and then use these estimates to learn the transition dynamics. We propose a method, which uses non-parametric multi-objective models for LFI of states, and non-parametric model of transition dynamics for learning state transition dynamics. The contributions of the paper are:

- Focusing on problems where LFI has to be sample-efficient, we improve upon the current LFI methods for the state inference task, while using samples from the state posteriors to accurately learn the state transition dynamics, as shown by comparisons with state-of-the-art SSMs inference techniques. We demonstrate that the proposed method is especially suitable for applications with non-stationary user models.
- We propose sampling from transition dynamics model as a sample-efficient strategy to determine where to run simulations next. We show empirical comparisons with traditional techniques in settings where simulations are computationally expensive.

2 Likelihood-Free State Inference

For LFI, we follow a Bayesian optimization for LFI (BOLFI) approach [36], in which a Gaussian process (GP) surrogate is used for a discrepancy measure δ (e.g. Euclidean distance) between observed $\mathbf{x}^*$ and synthetic datasets $\mathbf{x}^{(i)}$. This approach assumes that synthetic observations with the small discrepancy (smaller than a user-defined threshold ϵ) were produced by the simulator parameters that could plausibly replicate the observed datasets. In this way, the likelihood can be approximated as $p(\mathbf{x}^*|\boldsymbol{\theta}) \propto F\left(\frac{\epsilon-\mu(\boldsymbol{\theta})}{\sqrt{\nu(\boldsymbol{\theta})+\sigma^2}}\right)$, where $F(\cdot)$ is a Gaussian CDF with the mean 0 and the variance 1, $\mu(\boldsymbol{\theta})$ and $\nu(\boldsymbol{\theta})$ are the mean and the standard deviation of a GP, and σ is a likelihood noise of the GP. Because BOLFI actively chooses where to run simulations, its posterior approximation requires much fewer synthetic observations than other LFI methods. However, BOLFI was not designed for SSMs, and hence, does not make use of any temporal information that is typical for SSMs to improve quality of inference.

As an extension to BOLFI, we employ a multi-objective surrogate model for discrepancies $\boldsymbol{\delta}_t$, thus considering multiple discrepancy objectives at the same time and leveraging information between consecutive states. This approach allows using a discrepancy model of the previous state to infer the current state value, instead of simply discarding it. There is an additional advantage of having a multi-objective surrogate in an SSM setting, namely we can have a much more flexible surrogate for LFI than a traditional GP. This does not need any additional data to fit the surrogates, because all synthetic observations for discrepancy objectives can be shared across all state values. When we have a new observation, we simply need to recalculate the discrepancy values for all synthetic datasets. Once we have a trained surrogate for discrepancy objectives, we infer state posteriors $p(\boldsymbol{\theta}_t|\mathbf{x}_t)$, similarly as in BOLFI.

There is an additional challenge for adapting multi-objective surrogates in SSMs, namely a high computational cost associated with considering too many objectives. SSMs can potentially have hundreds of observations, and expanding the number of considered objectives may be detrimental for 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 objectives 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 L of the window 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. The moving window reduces the number of objectives L considered at a time, making multi-objective modelling for SSMs feasible. In the experiments, we used a linear model of coregionalization (LMC) [37] as a surrogate, with a moving window size of 2.

3 Learning State Transition Dynamics

Once we have approximated the posteriors $p(\boldsymbol{\theta}_t|\mathbf{x}_t^*)$ with LFI, we can use their empirical samples to learn a state transition model. This model should be able to train from noisy samples of LFI posterior approximations, and be flexible enough to fit arbitrary function families the dynamics may follow. Thus, the transition model should be Bayesian and non-parametric. Such a model would take the uncertainty associated with posterior approximations into account 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 from paired posterior samples. For each SSM time interval, we group consecutive state posterior samples in a training set, and expand it when new state posteriors become available. This way, the transition model does not need to be retrained when new observations present themselves and can be actively used throughout state inference for determining where to run simulations next. This can be done, by sampling the predictive posterior $p(\boldsymbol{\theta}_{T+1}|\mathbf{x}_T^*)$ from the trained model $\hat{h}_{\boldsymbol{\theta}}$ through $p(\boldsymbol{\theta}_{T+1}|\mathbf{x}_T^*) = \int \hat{h}_{\boldsymbol{\theta}}(\boldsymbol{\theta}_{T+1}|\boldsymbol{\theta}_T)p(\boldsymbol{\theta}_T|\mathbf{x}_T^*)d\boldsymbol{\theta}_T$. Thus, the state transition model $\hat{h}_{\boldsymbol{\theta}}$ does not inform state posteriors directly, but provides simulation candidates for the LFI surrogate. In the experiments, we used a Bayesian Neural Network (BNN) [38, 39] as a transition model for the state dynamics.

4 Experiments

We assess the quality of our method, LMC-BNN, for the inference and prediction tasks in a series of SSM experiments, where a simulator serves as the observation model $g_{\boldsymbol{\theta}}$. For the inference task, we compare the quality of our estimates against other LFI methods: BOLFI [36], SNPE [40], SNLE [41], SNRE [42]. We use a fixed simulation budget for all these methods, with 20 simulations to initialize the models, and then with 2 additional simulations for each new time-step. However, for the neural density estimation (NDE) approaches (SNPE , SNLE, SNRE), we provided all simulations at once, since they do not need additional simulations when observations change. For the prediction task, we sampled trajectories from the transition model and evaluated them against trajectories from the true dynamics. We performed these experiments in SSMs with simulators that have tractable likelihoods, providing them to the state-of-the-art SSM inference methods, GP-SSM [] and PR-SSM [], while our method was still doing LFI. For all methods, we provided 50 observations, and then sampled trajectories that had the same length of 50 time-steps.

We also supplied two variations of our method that differ only in the way the next simulations are sampled: LMC-BLR, where samples were taken from a Bayesian Linear Regression (BLR) model that was used as a local linearized model of the transition dynamics; and LMC-qEHVI, where a popular acquisition function for multitask BO, q-Expected Hypervolume Improvement (qEHVI) [43], was used to provide samples.

4.1 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 that completed two different tasks: uniform manifold approximation and projection (UMAP) [44] parametrization and eye movement control for gaze-based selection. Our goal in these experiments was to infer parameters of user models that generated the behavioural data, with three state dimensions in each user model. In the experimental results, these case studies are referred to as UMAP and Gaze, respectively.

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 noise. The NN model is a modified version (leaving only one unique solution for each observation) of a popular non-linear SSM [45]. Lastly, SV models are widely used for predicting volatility of asset prices [46, 47]. We use the specific model by [48]. The dimensionality of states in these models ranges from one to three, starting from LG and moving to SV. More detailed descriptions of all SSMs can be found in the Appendix.

Table 1: Comparison of LFI methods (rows) in different SMMs (columns) for the state inference task. The performance was measured with 95% confidence interval (CI) of the RMSE between estimated vs ground truth state values for 50 time-steps. The best results in each column are highlighted in bold.

Methods	**LG**	**NN**	**SV**	**UMAP**	**Gaze**
LMC-BNN	1.7 ± 0.1	6.75 ± 1.15	16.85 ± 2.75	$\mathbf{59.15 \pm 3.65}$	$\mathbf{59.4 \pm 5.9}$
LMC-BLR	$\mathbf{1.3 \pm 0.1}$	$\mathbf{5.55 \pm 0.75}$	$\mathbf{13.15 \pm 3.25}$	61.95 ± 3.05	60.6 ± 5.8
LMC-qEHVI	1.5 ± 0.1	7.1 ± 0.6	24.4 ± 2.5	65.35 ± 3.85	56.9 ± 4.5
BOLFI	1.55 ± 0.15	6.05 ± 0.45	27.35 ± 3.45	67.85 ± 3.35	73.45 ± 5.75
SNPE	7.15 ± 0.65	50.85 ± 1.35	77.4 ± 3.1	71.4 ± 3.5	68.1 ± 7.8
SNLE	6 ± 0.5	24.2 ± 1.6	54.25 ± 2.45	69.95 ± 3.35	77.25 ± 4.05
SNRE	10.4 ± 1.7	57.1 ± 1.7	57.15 ± 2.35	73.65 ± 1.25	80.75 ± 1.35

Table 2: Comparison of transition dynamics models (rows) in different SMMs (columns). The performance was measured with 95% confidence interval (CI) of the RMSE between sampled vs ground truth trajectories of length 50. The best results in each column are highlighted in bold.

Methods	**LG**	**NN**	**SV**	**UMAP**	**Gaze**
LMC-BNN	205 ± 9	$\mathbf{165 \pm 15}$	135 ± 22	$\mathbf{1383 \pm 30}$	$\mathbf{1353 \pm 7}$
LMC-BLR	$\mathbf{64 \pm 7}$	258 ± 37	$\mathbf{100 \pm 37}$	1409 ± 49	1372 ± 3
GP-SSM	284 ± 71	2204 ± 1111	3206 ± 1175	-	-
PR-SSM	253 ± 68	610 ± 510	1378 ± 740	-	-

4.2 Results and Discussion

The results for the inference and prediction tasks are presented in Tables 1 and 2, respectively. These tables report Root Mean Squared Error (RMSE), as a performance measure. In the inference task, LMC-based methods clearly outperformed BOLFI and NDE approaches. This indicates that considering multiple objectives at the same time was beneficial for the state inference, and that the model actually leverages information from consecutive states without hindering the performance. Additionally, it can be seen that LMC-based methods performed differently, which can be only attributed to how the next simulations were chosen, as the surrogate was exactly the same in all three methods. As 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 the simpler models. This suggests that BLR is expressive enough to replicate simple transitions, but struggles with more complex ones, for which BNN was more suitable.

The comparisons with GP-SSMs and PR-SSMs for learning transition dynamics show that our method learns the dynamics more accurately. The SSMs methods that were supposed to provide ground truth performance showed worse results than BLR and BNN approaches. This can be only 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 linearization, which is a far more limiting approach than having BNN. Therefore, BNN is more preferable as a model for transition dynamics.

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 and 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 allowed per time-step.

References

[1] Scott A Sisson, Yanan Fan, and Mark Beaumont. *Handbook of approximate Bayesian computation*. CRC Press, 2018.

[2] Mikael Sunnåker, Alberto Giovanni Busetto, Elina Numminen, Jukka Corander, Matthieu Foll, and Christophe Dessimoz. Approximate bayesian computation. *PLoS Comput Biol*, 9(1):e1002803, 2013.

[3] Gareth W Peters, Scott A Sisson, and Yanan Fan. Likelihood-free bayesian inference for α-stable models. *Computational Statistics & Data Analysis*, 56(11):3743–3756, 2012.

[4] Simon Barthelmé and Nicolas Chopin. Expectation propagation for likelihood-free inference. *Journal of the American Statistical Association*, 109(505):315–333, 2014.

[5] Victor M-H Ong, David J Nott, Minh-Ngoc Tran, Scott A Sisson, and Christopher C Drovandi. Likelihood-free inference in high dimensions with synthetic likelihood. *Computational Statistics & Data Analysis*, 128:271–291, 2018.

[6] Mark A Beaumont. Approximate bayesian computation in evolution and ecology. *Annual review of ecology, evolution, and systematics*, 41:379–406, 2010.

[7] Giorgio Bertorelle, Andrea Benazzo, and S Mona. Abc as a flexible framework to estimate demography over space and time: some cons, many pros. *Molecular ecology*, 19(13):2609–2625, 2010.

[8] Mark A Beaumont, Wenyang Zhang, and David J Balding. Approximate bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

[9] Chad M Schafer and Peter E Freeman. Likelihood-free inference in cosmology: Potential for the estimation of luminosity functions. In *Statistical Challenges in Modern Astronomy V*, pages 3–19. Springer, 2012.

[10] Justin Alsing, Benjamin Wandelt, and Stephen Feeney. Massive optimal data compression and density estimation for scalable, likelihood-free inference in cosmology. *Monthly Notices of the Royal Astronomical Society*, 477(3):2874–2885, 2018.

[11] Niall Jeffrey, Justin Alsing, and François Lanusse. Likelihood-free inference with neural compression of des sv weak lensing map statistics. *Monthly Notices of the Royal Astronomical Society*, 501(1):954–969, 2021.

[12] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[13] Rudolf Emil Kalman et al. Contributions to the theory of optimal control. *Boletín de la Sociedad Matemática*, 5(2):102–119, 1960.

[14] Xubin Zeng, Robert Atlas, Ronald J Birk, Frederick H Carr, Matthew J Carrier, Lidia Cucurull, William H Hooke, Eugenia Kalnay, Raghu Murtugudde, Derek J Posselt, et al. Use of observing system simulation experiments in the united states. *Bulletin of the American Meteorological Society*, 101(8):E1427–E1438, 2020.

[15] Ronald M Errico, Runhua Yang, Nikki C Privé, King-Sheng Tai, Ricardo Todling, Meta E Sienkiewicz, and Jing Guo. Development and validation of observing-system simulation experiments at nasa's global modeling and assimilation office. *Quarterly Journal of the Royal Meteorological Society*, 139(674):1162–1178, 2013.

[16] Johannes U Lange, Frank C van den Bosch, Andrew R Zentner, Kuan Wang, Andrew P Hearin, and Hong Guo. Cosmological evidence modelling: a new simulation-based approach to constrain cosmology on non-linear scales. *Monthly Notices of the Royal Astronomical Society*, 490(2):1870–1878, 2019.

[17] Siyu He, Yin Li, Yu Feng, Shirley Ho, Siamak Ravanbakhsh, Wei Chen, and Barnabás Póczos. Learning to predict the cosmological structure formation. *Proceedings of the National Academy of Sciences*, 116(28):13825–13832, 2019.

[18] Theodosis Georgiou and Yiannis Demiris. Adaptive user modelling in car racing games using behavioural and physiological data. *User Modeling and User-Adapted Interaction*, 27(2):267–311, 2017.

[19] Olivier Gimenez, Vivien Rossi, Rémi Choquet, Camille Dehais, Blaise Doris, Hubert Varella, Jean-Pierre Vila, and Roger Pradel. State-space modelling of data on marked individuals. *Ecological modelling*, 206(3-4):431–438, 2007.

[20] Tina Toni, David Welch, Natalja Strelkowa, Andreas Ipsen, and Michael PH Stumpf. Approximate bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, 6(31):187–202, 2009.

[21] Thomas A Dean, Sumeetpal S Singh, Ajay Jasra, and Gareth W Peters. Parameter estimation for hidden markov models with intractable likelihoods. *Scandinavian Journal of Statistics*, 41(4):970–987, 2014.

[22] Emrah Zerdali and Murat Barut. The comparisons of optimized extended kalman filters for speed-sensorless control of induction motors. *IEEE Transactions on industrial electronics*, 64(6):4340–4351, 2017.

[23] Brian DO Anderson and John B Moore. *Optimal filtering*. Courier Corporation, 2012.

[24] Masaru Hoshiya and Etsuro Saito. Structural identification by extended kalman filter. *Journal of engineering mechanics*, 110(12):1757–1770, 1984.

[25] Silvan Melchior, Sebastian Curi, Felix Berkenkamp, and Andreas Krause. Structured variational inference in unstable gaussian process state space models. *arXiv preprint arXiv:1907.07035*, 2019.

[26] Roger Frigola, Yutian Chen, and Carl Edward Rasmussen. Variational gaussian process state-space models. *Advances in neural information processing systems*, 27:3680–3688, 2014.

[27] Maximilian Karl, Maximilian Soelch, Justin Bayer, and Patrick Van der Smagt. Deep variational bayes filters: Unsupervised learning of state space models from raw data. *arXiv preprint arXiv:1605.06432*, 2016.

[28] Vaclav Smidl and Anthony Quinn. Variational bayesian filtering. *IEEE Transactions on Signal Processing*, 56(10):5020–5030, 2008.

[29] Sinan Yıldırım, Sumeetpal S Singh, Thomas Dean, and Ajay Jasra. Parameter estimation in hidden markov models with intractable likelihoods using sequential monte carlo. *Journal of Computational and Graphical Statistics*, 24(3):846–865, 2015.

[30] Adrian Smith. *Sequential Monte Carlo methods in practice*. Springer Science & Business Media, 2013.

[31] François Septier, Gareth W Peters, and Ido Nevat. Bayesian filtering with intractable likelihood using sequential mcmc. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6313–6317. IEEE, 2013.

[32] Ajay Jasra, Sumeetpal S Singh, James S Martin, and Emma McCoy. Filtering via approximate Bayesian computation. *Statistics and Computing*, 22(6):1223–1237, 2012.

[33] Gael M Martin, Brendan PM McCabe, David T Frazier, Worapree Maneesoonthorn, and Christian P Robert. Auxiliary likelihood-based approximate Bayesian computation in state space models. *Journal of Computational and Graphical Statistics*, 28(3):508–522, 2019.

[34] Laurent E Calvet and Veronika Czellar. Accurate methods for approximate Bayesian computation filtering. *Journal of Financial Econometrics*, 13(4):798–838, 2015.

[35] James S Martin, Ajay Jasra, Sumeetpal S Singh, Nick Whiteley, Pierre Del Moral, and Emma McCoy. Approximate Bayesian computation for smoothing. *Stochastic Analysis and Applications*, 32(3):397–420, 2014.

[36] Michael U Gutmann and Jukka Corander. Bayesian optimization for likelihood-free inference of simulator-based statistical models. *The Journal of Machine Learning Research*, 17(1):4256–4302, 2016.

[37] Thomas R Fanshawe and Peter J Diggle. Bivariate geostatistical modelling: a review and an application to spatial variation in radon concentrations. *Environmental and ecological statistics*, 19(2):139–160, 2012.

[38] Igor Kononenko. Bayesian neural networks. *Biological Cybernetics*, 61(5):361–370, 1989.

[39] Piero Esposito. Blitz - bayesian layers in torch zoo (a bayesian deep learing library for torch). `https://github.com/piEsposito/blitz-bayesian-deep-learning/`, 2020.

[40] George Papamakarios and Iain Murray. Fast ε-free inference of simulation models with bayesian conditional density estimation. In *Advances in neural information processing systems*, pages 1028–1036, 2016.

[41] George Papamakarios, David Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 837–848. PMLR, 2019.

[42] Conor Durkan, Iain Murray, and George Papamakarios. On contrastive learning for likelihood-free inference. In *International Conference on Machine Learning*, pages 2771–2781. PMLR, 2020.

[43] Samuel Daulton, Maximilian Balandat, and Eytan Bakshy. Differentiable expected hypervolume improvement for parallel multi-objective Bayesian optimization. *arXiv preprint arXiv:2006.05078*, 2020.

[44] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[45] Genshiro Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *Journal of computational and graphical statistics*, 5(1):1–25, 1996.

[46] Stephen J Taylor. Modeling stochastic volatility: A review and comparative study. *Mathematical finance*, 4(2):183–204, 1994.

[47] Neil Shephard. Statistical aspects of arch and stochastic volatility. *Monographs on Statistics and Applied Probability*, 65:1–68, 1996.

[48] Ole E Barndorff-Nielsen and Neil Shephard. Econometric analysis of realized volatility and its use in estimating stochastic volatility models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(2):253–280, 2002.

[49] Ethem Alpaydin and Cenk Kaynak. Cascading classifiers. *Kybernetika*, 34(4):369–374, 1998.

[50] Davoud Moulavi, Pablo A Jaskowiak, Ricardo JGB Campello, Arthur Zimek, and Jörg Sander. Density-based clustering validation. In *Proceedings of the 2014 SIAM international conference on data mining*, pages 839–847. SIAM, 2014.

[51] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, 1992.

[52] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[53] Immo Schuetz, T Scott Murdison, Kevin J MacKenzie, and Marina Zannoli. 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*, pages 1–13, 2019.

[54] Xinyong Zhang, Xiangshi Ren, and Hongbin Zha. Modeling dwell-based eye pointing target acquisition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2083–2092, 2010.

[55] Xiuli Chen, Aditya Acharya, and Antti Oulasvirta. An adaptive model of gaze-based selection. In *CHI Conference on Human Factors in Computing Systems (CHI'21)*. Association for Computing Machinery, 2021.

A State-Space Model Descriptions

A.1 Linear Gaussian

In the linear Gaussian (LG) model, the state transition dynamics (Figure 1a, right) and observation model are both linear

$$\theta_t = 0.95 \times \theta_{t-1} + v_t \tag{1}$$
$$x_t = \theta_t + w_t, \tag{2}$$

with added Gaussian noise $v_t \sim \mathcal{N}(0, 1^2)$ and $w_t \sim \mathcal{N}(0, 10^2)$. The initial state value for the transition dynamics is $\theta_0 = 100$. The prior for the state values is

$$\theta \sim \text{Unif}(0, 15) \in \mathbb{R}. \tag{3}$$

A.2 Stochastic Volatility

Stochastic volatility (SV) models are widely used for predicting volatility of asset prices [46, 47]. Here, we use the model by [48] that specifies transition dynamics (Figure 1c, right) of volatility θ_v as

$$\theta_{v,t+1} = (z_t - z_{t+1} + \sum_{j=1}^{k} e_j) \tag{4}$$
$$z_{t+1} = \exp(-\lambda) z_t + \sum_{j=1}^{k} \exp\{-\lambda(t + 1 - c_j)\} e_j, \tag{5}$$

with $c_{1:k} \sim \text{Unif}(t, t+1)$, $e_{1:k} \sim \text{Expon}(0.5/0.25^2)$ and $\lambda = 0.01$. The random increases of volatility are regulated by the Poisson distributed variable $k \sim \text{Poisson}(0.5\lambda^2/0.25^2)$. The observation model for the log-return of an asset $x_t \in \mathbb{R}$ follows

$$x_t = \theta_\mu + \theta_\beta \theta_{v,t} + (\theta_{v,t}^{0.5} + 10^{-5}) \varepsilon_t, \tag{6}$$

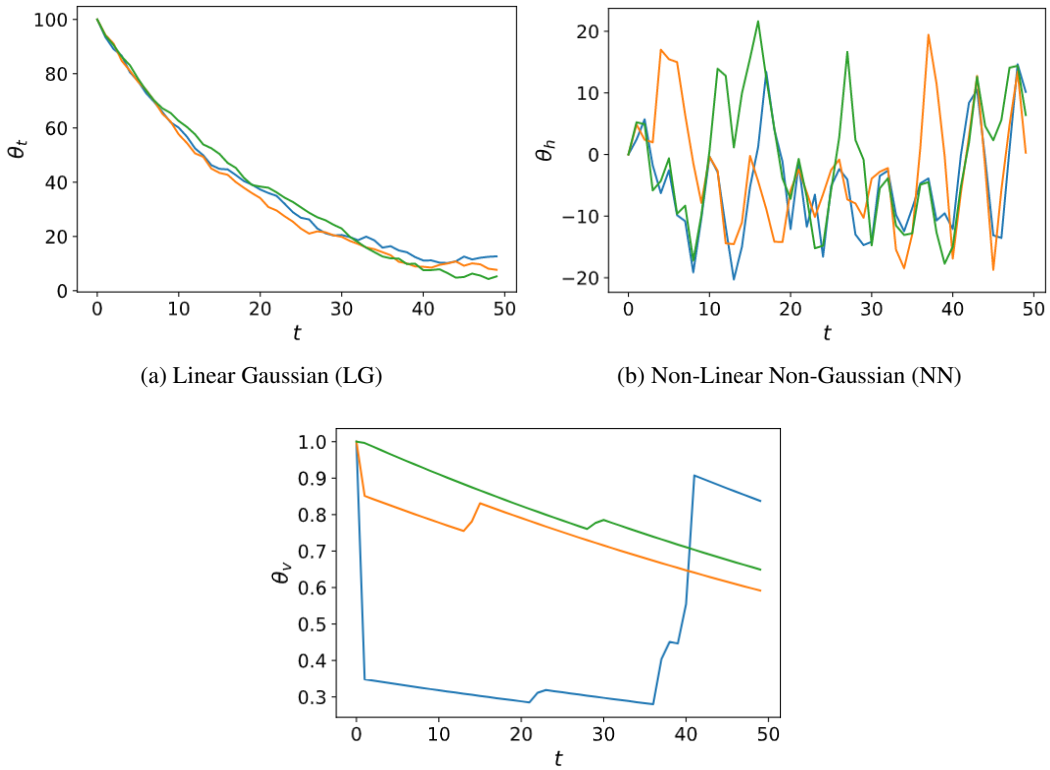
where $\varepsilon_t \sim \mathcal{N}(0, 1)$. As in this SSM, $\theta_\mu = 0$ and $\theta_\beta = 0$ remain the same, while the volatility θ_v follows the transition dynamics, starting with the initial value for the volatility $\theta_{v,0}$ of 1. We used the following priors for LFI of state values:

$$\theta_\mu \sim \text{Unif}(-2, 2) \in \mathbb{R} \tag{7}$$
$$\theta_\beta \sim \text{Unif}(-5, 5) \in \mathbb{R} \tag{8}$$
$$\theta_v \sim \text{Unif}(0, 3) \in \mathbb{R}. \tag{9}$$

A.3 Non-Linear Non-Gaussian

The non-linear non-Gaussian (NN) model is a modified version of a popular non-linear SSM [45], where the transition dynamics (Figure 1b, right) and observation model are

$$\theta_{h,t} = \frac{\theta_{h,t-1}}{2} + 25 \frac{\theta_{h,t-1}}{\theta_{h,t-1}^2 + 1} + 8\cos(1.2t) + v_t \tag{10}$$
$$x_t = \frac{\theta_{h,t}}{5} + \theta_n w_t, \tag{11}$$



(a) Linear Gaussian (LG)

(b) Non-Linear Non-Gaussian (NN)

(c) Stochastic Volatility (SV)

Figure 1: Transition trajectories (different colours) of states (y-axis) sampled from three SSM transition dynamics across 50 time-steps (x-axis) with different random seeds. The complexity of the dynamics gradually increases in SSMs, starting with the smooth LG (a) dynamics, where the difference between consecutive states is very small, followed by the NN model (b) whose dynamics is very erratic with a lot of unexpected changes, and finishing with SV (b) dynamics with a few occasional drastic changes of state values.

with added Gaussian noise $v_t \sim \mathcal{N}(0, 10)$ and $w_t \sim \mathcal{N}(0, 1)$. The initial state value for the noise standard deviation $\theta_{h,0}$ is 0, while the unobserved signal $\theta_n = 1$ always remains the same. The priors for the parameters are

$$\theta_n \sim \text{Unif}(0, 10) \in \mathbb{R} \tag{12}$$

$$\theta_h \sim \text{Unif}(-30, 30) \in \mathbb{R}. \tag{13}$$

A.4 UMAP Parameterization

In this section, we infer Uniform Manifold Approximation and Projection (UMAP) [44] parameters that generate low-dimensional data representations for a classification task that best satisfy the subject's needs. The behavioural data in this experiment is the dataset embedding that maximizes a function of the subject's human needs. In the beginning, the subject does not have any prior knowledge of the dataset, so data exploration takes the priority. As the user becomes more familiar with data, the priority gradually shifts to maximizing classification accuracy on a validation set. We assume that by modelling the change in human-user preferences, we can anticipate them and propose good UMAP parameter candidates to the human-user faster.

For this experiment, the subject's needs are simulated by an evaluation function that takes the embedding of the handwritten digit dataset [49] as an input and assigns a corresponding preference

score. The evaluation function is based on the weighted sum of the Density Based Cluster Validity (DBCV) score $\mathcal{U}^{(i)}$ [50] and the C-Support Vector Classification (SVC) [51, 52] accuracy $\mathcal{P}^{(i)}$. By regulating the weight w_t, one can prioritize data exploration $\mathcal{U}^{(i)}$ or maximization of a classification accuracy $\mathcal{P}^{(i)}$ with

$$\delta_t^{(i)} = (1 - w_t)\, \mathcal{U}^{(i)} + w_t\, \mathcal{P}^{(i)}, \quad w_t = \frac{t + \sin t}{50}. \tag{14}$$

Since we assume the subject cannot specify the ideal embedding, the SSM observations $\mathbf{x}^*$ are latent. But since the subject has the ability to evaluate embeddings with the preference score, we model the preference score $\delta_t^{(i)}$ directly as a discrepancy. The goal is to infer the state values $\boldsymbol{\theta} = \{\theta_d, \theta_{dist}, \theta_n\}$ that correspond to the UMAP parameters: the dimension of the target reduced space θ_d, an algorithmic parameter θ_{dist} that controls how densely the points packed, and the neighborhood size θ_n to use for local metric approximation. Therefore, the UMAP algorithm serves as an observation model and the transition dynamics is unknown, but implicitly regulated through (14). We use the following priors for states throughout the experiments:

$$\theta_d \sim Unif(1, 64) \in \mathbb{Z} \tag{15}$$
$$\theta_{dist} \sim Unif(0, 0.99) \in \mathbb{R} \tag{16}$$
$$\theta_n \sim Unif(2, 200) \in \mathbb{Z}. \tag{17}$$

A.5 Eye Movement Control For Gaze-Based Selection

In this section, we infer properties of human gaze in a series of simulated eye movement control tasks [53, 54]. In these tasks, a user model of the human searches for a target on a 2D screen by performing eye movements (saccades), based on its beliefs about the target location and information from peripheral vision. When the gaze location matches the location on the screen, the task is considered complete and a new target appears. As the human subject performs more tasks, they get weary, which results in an increasing latency between saccades θ_l. Our hypothesis is that modelling the dynamics of eye movement latency will allow robust inference of the individual characteristics that are independent of weariness.

The subject needs several moves to reach the target because the actions and observations are corrupted with two noise sources: the ocular motor noise θ_{om} and the spatial noise of peripheral vision θ_s. The quantitative values for these two variables vary for each individual, while the saccade latency θ_l varies between different task trial instances. We assess the performance of the subject based on total time $x \in \mathbb{R}$ it took for the gaze to reach the target,

$$x_t = \sum_e (2.7 \times A^{(e)} + \theta_{l,t}), \quad \theta_{l,t} = 12 \times log(t + 1) + 37. \tag{18}$$

Here, x_t are SSM observations, $\boldsymbol{\theta} = \{\theta_{om}, \theta_s, \theta_l\}$ are state values that need to be inferred, $A^(e)$ is the saccade amplitude, and e is the saccade index. The increase of the latency θ_l serves as state transition dynamics h_t that needs to be modelled for prediction, while eye movement control task environment is considered as an observation model $g_{\boldsymbol{\theta}}$. To produce behavioural data, ground truth values of 0.01 and 0.09 for θ_{om} and θ_s respectively were used in the observation model. Finally, we calculate a Euclidean distance directly applied to x as a discrepancy and use the following priors for the states:

$$\theta_{om} \sim Unif(0, 0.2) \in \mathbb{R} \tag{19}$$
$$\theta_s \sim Unif(0, 0.2) \in \mathbb{R} \tag{20}$$
$$\theta_l \sim Unif(30, 60) \in \mathbb{R}, \tag{21}$$

The user model was implemented with reinforcement learning by [55].