
Probabilistic Recurrent State-Space Models

Andreas Doerr^{1,2} Christian Daniel¹ Martin Schiegg¹ Duy Nguyen-Tuong¹ Stefan Schaal^{2,3} Marc Toussaint⁴
Sebastian Trimpe²

Abstract

State-space models (SSMs) are a highly expressive model class for learning patterns in time series data and for system identification. Deterministic versions of SSMs (e.g., LSTMs) proved extremely successful in modeling complex time series data. Fully probabilistic SSMs, however, are often found hard to train, even for smaller problems. We propose a novel model formulation and a scalable training algorithm based on doubly stochastic variational inference and Gaussian processes. This combination allows efficient incorporation of latent state temporal correlations, which we found to be key to robust training. The effectiveness of the proposed PR-SSM is evaluated on a set of real-world benchmark datasets in comparison to state-of-the-art probabilistic model learning methods. Scalability and robustness are demonstrated on a high dimensional problem.

1. Introduction

System identification, i.e. learning dynamics models from data (Ljung, 2010), is key in model-based control design (Aström & Murray, 2010; Camacho & Alba, 2013) and model-based reinforcement learning (RL) (Deisenroth & Rasmussen, 2011; Doerr et al., 2017b). State-space models (SSMs) are one popular class of representations for model learning (Billings, 2013), which describe a system with input \mathbf{u}_t , output \mathbf{y}_t , and a latent Markovian state \mathbf{x}_t . The transition model f , observation model g , process, and measurement noise ϵ_t and γ_t form a discrete-time SSM

$$\begin{aligned}\mathbf{x}_{t+1} &= f(\mathbf{x}_t, \mathbf{u}_t) + \epsilon_t, \\ \mathbf{y}_t &= g(\mathbf{x}_t) + \gamma_t.\end{aligned}\tag{1}$$

¹Bosch Center for Artificial Intelligence, Renningen, Germany.

²Max Planck Institute for Intelligent Systems, Stuttgart/Tübingen, Germany. ³University of Southern California, Los Angeles, USA.

⁴Machine Learning and Robotics Lab, University of Stuttgart, Germany. Correspondence to: Andreas Doerr <andreasdoerr@gmx.net>.

In real systems, the latent state \mathbf{x}_t can typically not be measured directly, but has to be inferred from a series of noisy output observations. Efficient methods exist for linear models (Van Overschee & De Moor, 2012) and non-linear, deterministic models (Hochreiter & Schmidhuber, 1997).

Probabilistic models enable safe RL and alleviate model bias (Deisenroth & Rasmussen, 2011). Placing a Gaussian process (GP) prior on the unknown transition function f and potentially on the observation model g allows for Bayesian model learning, resulting in GP-SSMs. Robust training of these *probabilistic, non-linear* SSMs is a challenging and only partially solved problem, especially for higher dimensional systems (Frigola et al., 2013; Bayer & Osendorfer, 2014; Krishnan et al., 2015; Fraccaro et al., 2016; Eleftheriadis et al., 2017; Svensson & Schön, 2017). This paper proposes the probabilistic recurrent state-space model¹ (PR-SSM²). PR-SSM takes inspiration from RNN model learning. In particular, the transition model is unrolled over time, therefore accounting for temporal correlations whilst simultaneously allowing for learning by backpropagation through time. The proposed method enables probabilistic model predictions, inferring complex latent state distributions, and principled model complexity regularization. We propose an adapted form of a recognition model for the initial state, which facilitates scalability through batch learning and learning of slow or unstable system dynamics (cf. Sec. 5).

In summary, the key contributions of this paper are:

- Combining gradient-based and sample-based inference for efficient learning of nonlinear Gaussian process state-space models;
- Tractable variational approximation, maintaining the true latent state posterior and temporal correlations;
- Doubly stochastic inference scheme for scalability;
- Recognition model, which allows for initializing the latent state distribution and thus for robust training and prediction.

Together, these contributions allow for efficient and robust learning of the PR-SSM. The proposed framework is evaluated on a set of real-world system identification datasets and benchmarked against a range of state-of-the-art methods.

¹Code available at: <https://github.com/boschresearch/PR-SSM>.

²Pronounced *prism*.

2. Related Work

Modeling the behavior of systems with only partially observable states has been an active field of research for many years and several schools of thought have emerged. Representations range from SSMs (Van Overschee & De Moor, 2012) over Predictive State Representations (PSRs) (Littman & Sutton, 2002; Singh et al., 2003; Rudary & Singh, 2004) to autoregressive models (Murray-Smith & Girard, 2001; Girard et al., 2003; Likar & Kocijan, 2007; Billings, 2013), as well as hybrid versions combining these approaches (Mattos et al., 2015; 2016; Doerr et al., 2017a).

Autoregressive (history-based) methods avoid the complex inference of a latent state and instead directly learn a mapping from a history of h past inputs and observations to the next observation, i.e. $\mathbf{y}_{t+1} = f(\mathbf{y}_{t:t-h}, \mathbf{u}_{t:t-h})$. These models face the issue of learning from noise corrupted input data. Recent work addresses this problem by either actively accounting for input noise (McHutchon & Rasmussen, 2011) or reverting to a hybrid, autoregressive formulation in a latent, but noise free state (Mattos et al., 2016; Doerr et al., 2017a). Such models can be made deep and trained in a recurrent manner (Mattos et al., 2015).

In contrast, SSMs are based on a compact, Markovian state representation. Furthermore, they allow for the direct application of many existing control algorithms, which rely on the explicit representation of the latent state. Exact solutions for state inference and model learning for linear Gaussian SSMs are given by the well known Kalman filter/smoothers (Kalman, 1960) and subspace identification (Van Overschee & De Moor, 2012). In the case of non-linear latent state transition dynamics, both deterministic and probabilistic variants are active fields of research.

Deterministic variants such as Long Short-Term Memory (LSTM) models have been shown to be powerful representations for tasks such as natural language processing (Venugopalan et al., 2014) or text understanding (Sutskever et al., 2011). However, for the purpose of system identification and control, probabilistic predictions are often preferred to make model errors explicit (Deisenroth & Rasmussen, 2011). A variety of stochastic deep recurrent models has been presented based on Stochastic Gradient Variational Bayes (SGVB) (Bayer & Osendorfer, 2014; Krishnan et al., 2015; Watter et al., 2015; Chung et al., 2015; Archer et al., 2015; Karl et al., 2016; Fraccaro et al., 2016; Gemici et al., 2017). The PR-SSM inference is inspired by the learning procedure in these deep recurrent models while employing GPs as a principled way of model regularization. Both procedures share the explicit unrolling of transition and observation model. Errors between the predicted and the observed system output are propagated back over time. Therefore, the transition dynamics has to be inferred, but the latent state (distribution) is given implicitly. This way, the challenging

initialization and optimization of latent state variables is prevented. In contrast to deep recurrent models, the PR-SSM loss and model regularization is automatically obtained from the GP assumption. Furthermore, PR-SSMs obtain predictive distributions and the proposed initial state recognition model facilitates learning on shorter sub-trajectories and unstable systems, which is not possible in deep recurrent models.

GP-SSMs are a popular class of probabilistic SSMs (Wang et al., 2008; Ko & Fox, 2009; Turner et al., 2010; Frigola et al., 2013; 2014; Eleftheriadis et al., 2017). The use of GPs allows for a fully Bayesian treatment of the modeling problem resulting in an automatic complexity trade-off, which regularizes the learning problem. Filtering and smoothing in GP-SSMs has already been covered extensively: deterministic (e.g. linearization) as well as stochastic (e.g. particles) methods are presented in (Ko & Fox, 2009; Deisenroth et al., 2012). These methods, however, assume an established system model, which is generally not available without prior knowledge. In this work, the latent state smoothing distribution is given implicitly and optimized jointly during model learning.

Approaches to probabilistic GP-SSMs mainly differ in their approximations to the model’s joint distribution (e.g. when solving for the smoothing distribution or for the observation likelihood). One class of approaches aims to solve for the true distribution, which requires sample-based methods, e.g. Particle Markov Chain Monte Carlo (PMCMC), as in (Frigola et al., 2013; 2014). These methods are close to exact and thus are able to represent temporal correlations. However, they are computationally inefficient and intractable for higher latent state dimensions or larger datasets. A second class of approaches is based on variational inference and mean field approximations in the latent state (Mattos et al., 2015; Föll et al., 2017). These methods, however, operate on latent autoregressive models, which can be initialized by the observed output time series, such that the learned latent representation acts as a smoothed version of the observations. In Markovian latent spaces, no such prior information is available and therefore initialization is non-trivial. Model optimization based on mean field approximations empirically leads to highly suboptimal local solutions. Bridging the gap between both classes, recent methods strive to recover (temporal) latent state structure. In (Eleftheriadis et al., 2017), a linear, time-varying latent state structure is enforced as a tractable compromise between the true non-linear dependencies and no dependencies as in mean field variational inference. However, to facilitate learning, a more complex recognition model over the linear time-varying dynamics is required. In contrast, the proposed PR-SSM can efficiently incorporate the true dynamics by combining sampling- and gradient-based learning.

3. Gaussian Process State-Space Model

This section presents the general model background for GP-SSMs. Following a short recap of GPs in Sec. 3.1 and a specific sparse GP prior in Sec. 3.2, PR-SSM as one particular GP-SSM is introduced in Sec. 3.3. Inference on this model is detailed in Sec. 4.

3.1. Gaussian Process

A GP (Williams & Rasmussen, 2005) is a distribution over functions $f : \mathbb{R}^D \rightarrow \mathbb{R}$ that is fully defined by a mean function $m(\cdot)$ and covariance function $k(\cdot, \cdot)$. For each finite set of points $\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N]$ from the function's domain, the corresponding function evaluations $\mathbf{f} = [f(\mathbf{x}_1), \dots, f(\mathbf{x}_N)]$ are jointly Gaussian as given by

$$p(\mathbf{f} | \mathbf{X}) = \mathcal{N}(\mathbf{f} | \mathbf{m}_X, \mathbf{K}_{X,X}), \quad (2)$$

with mean vector \mathbf{m}_X having elements $m_i = m(\mathbf{x}_i)$ and covariance matrix $\mathbf{K}_{X,X}$ with entries $K_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. Given observed function values \mathbf{f} at input locations \mathbf{X} , the GP predictive distribution at a new input location \mathbf{x}^* is obtained as the conditional distribution

$$p(f^* | \mathbf{x}^*, \mathbf{f}, \mathbf{X}) = \mathcal{N}(f^* | \mu, \sigma^2), \quad (3)$$

with posterior mean and variance

$$\mu = m_{x^*} + \mathbf{k}_{x^*,X} \mathbf{K}_{X,X}^{-1} (\mathbf{f} - \mathbf{m}_X), \quad (4)$$

$$\sigma^2 = k_{x^*,x^*} - \mathbf{k}_{x^*,X} \mathbf{K}_{X,X}^{-1} \mathbf{k}_{X,x^*}, \quad (5)$$

where $k_{A,B}$ denotes the scalar or vector of covariances for each pair of elements in \mathbf{A} and \mathbf{B} . In this work, the squared exponential kernel with Automatic Relevance Determination (ARD) (Williams & Rasmussen, 2005) with hyperparameters θ_{GP} is employed. Due to the proposed sampling-based inference scheme (cf. Sec. 4), any other differentiable kernel might be incorporated instead.

3.2. GP Sparsification

Commonly, the GP prediction in (3) is obtained by conditioning on all training data \mathbf{X} , \mathbf{y} . To alleviate the computational cost, several sparse approximations have been presented (Snelson & Ghahramani, 2006). By introducing P inducing GP targets $\mathbf{z} = [z_1, \dots, z_P]$ at pseudo input points $\boldsymbol{\zeta} = [\boldsymbol{\zeta}_1, \dots, \boldsymbol{\zeta}_P]$, which are jointly Gaussian with the latent function f , the true GP predictive distribution is approximated by conditioning only on this set of inducing points,

$$p(f^* | \mathbf{x}^*, \mathbf{f}, \mathbf{X}) \approx p(f^* | \mathbf{x}^*, \mathbf{z}, \boldsymbol{\zeta}), \quad (6)$$

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z} | \mathbf{m}_\zeta, \mathbf{K}_{\zeta,\zeta}). \quad (7)$$

The predicted function values consequently become mutually independent given the inducing points.

3.3. PR-SSM Model Definition

The PR-SSM is built upon a GP prior on the transition function $f(\cdot)$ and a parametric observation model $g(\cdot)$. This is a common model structure, which can be assumed without loss of generality over (1), since any observation model can be absorbed into a sufficiently large latent state (Frigola-Alcade, 2015). Eliminating the non-parametric observation model, however, mitigates the problem of ‘severe non-identifiability’ between transition model $f(\cdot)$ and observation model $g(\cdot)$ (Frigola et al., 2014). Independent GP priors are employed for each latent state dimension d given individual inducing points $\boldsymbol{\zeta}_d$ and \mathbf{z}_d .

In the following derivations, the system's latent state, input and output at time t are denoted by $\mathbf{x}_t \in \mathbb{R}^{D_x}$, $\mathbf{u}_t \in \mathbb{R}^{D_u}$, and $\mathbf{y}_t \in \mathbb{R}^{D_y}$, respectively. The shorthand $\hat{\mathbf{x}}_t = (\mathbf{x}_t, \mathbf{u}_t)$ denotes the transition model's input at time t . The output of the transition model is denoted by $\mathbf{f}_{t+1} = f(\hat{\mathbf{x}}_t)$. A time series of observations from time a to time b (including) is abbreviated by $\mathbf{y}_{a:b}$ (analogously for the other model variables).

The joint distribution of all PR-SSM random variables is given by

$$p(\mathbf{y}_{1:T}, \mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z}) = \left[\prod_{t=1}^T p(\mathbf{y}_t | \mathbf{x}_t) \right] p(\mathbf{x}_1) p(\mathbf{z}) \quad (8)$$

$$\left[\prod_{t=2}^T p(\mathbf{x}_t | \mathbf{f}_t) p(\mathbf{f}_t | \hat{\mathbf{x}}_{t-1}, \mathbf{z}) \right],$$

where $p(\mathbf{f}_t | \hat{\mathbf{x}}_{t-1}, \mathbf{z}) = \prod_{d=1}^{D_x} p(f_{t,d} | \hat{\mathbf{x}}_{t-1}, \mathbf{z}_d)$ and $\mathbf{z} \equiv [z_1, \dots, z_{D_x}]$. A graphical model of the resulting PR-SSM is shown in Fig. 1.

The individual contributions to (8) are given by the observation model and the transition model, which are now described in detail. The observation model is governed by

$$p(\mathbf{y}_t | \mathbf{x}_t) = \mathcal{N}(\mathbf{y}_t | g(\mathbf{x}_t), \text{diag}(\sigma_{y,1}^2, \dots, \sigma_{y,D_y}^2)), \quad (9)$$

In particular, in our experiments, we employed a parametric observation model

$$g(\mathbf{x}_t) = \mathbf{C} \mathbf{x}_t. \quad (10)$$

The matrix \mathbf{C} is chosen to select the D_y first entries of \mathbf{x}_t by defining $\mathbf{C} := [\mathbf{I}, \mathbf{0}] \in \mathbb{R}^{D_y \times D_x}$ with \mathbf{I} being the identity matrix. This model is suitable for observation spaces that are low-dimensional compared to the latent state dimensionality, i.e. $D_y < D_x$, which is often the case for physical systems with a restricted number of sensors. The first D_y latent state dimensions can therefore be interpreted as noise free sensor measurements. For high-dimensional observation spaces (e.g. images), a more involved, given observation model (e.g. a pretrained neural network) may be seamlessly

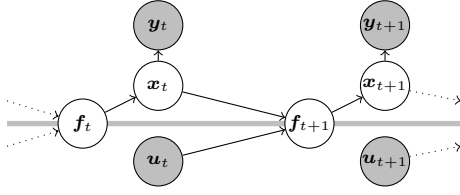


Figure 1. Graphical model of the PR-SSM. Gray nodes are observed variables in contrast to latent variables in white nodes. Thick lines indicate variables, which are jointly Gaussian under a GP prior.

incorporated into the presented framework as long as $g(\cdot)$ is differentiable.

Process noise is modeled as

$$p(\mathbf{x}_t | \mathbf{f}_t) = \mathcal{N}(\mathbf{x}_t | \mathbf{f}_t, \text{diag}(\sigma_{x,1}^2, \dots, \sigma_{x,D_x}^2)). \quad (11)$$

The transition dynamics is described independently for each latent state dimension d by $p(f_{t,d} | \hat{\mathbf{x}}_{t-1}, z_d)p(z_d)$. This probability is given by the sparse GP prior (7) and predictive distribution (6), where $x^* = \hat{\mathbf{x}}_t$ and $f^* = f_{t,d}$. The initial system state distribution $p(\mathbf{x}_1)$ is unknown and has to be estimated.

4. PR-SSM Inference

Computing the log likelihood or a posterior based on (8) is generally intractable due to the nonlinear GP dynamics model in the latent state. However, the log marginal likelihood $\log p(\mathbf{y}_{1:T})$ (evidence) can be bounded from below by the Evidence Lower BOund (ELBO) (Blei et al., 2017). This ELBO is derived via Jensen’s inequality by introducing a computationally simpler, variational distribution $q(\mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z})$ to approximate the model’s true posterior distribution $p(\mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z} | \mathbf{y}_{1:T})$ (cf. eq. (8)). In contrast to previous work (Frigola et al., 2014; Mattos et al., 2015; Eleftheriadis et al., 2017), the proposed approximation explicitly incorporates the true temporal correlations in the latent state, whilst being scalable to large datasets. Previous work based on sequential Monte Carlo methods (Frigola et al., 2013; Svensson & Schön, 2017) already allowed for temporal correlations but required computationally challenging resampling in each timestep. The inference scheme is inspired by doubly stochastic variational inference for deep GPs as presented in (Salimbeni & Deisenroth, 2017).

4.1. Variational Sparse GP

PR-SSM employs a variational sparse GP (Titsias, 2009) based on a variational distribution $q(\mathbf{z})$ on the GP’s inducing outputs as previously used in (Frigola et al., 2014; Eleftheriadis et al., 2017). Eliminating the inducing outputs, however, results in dependencies between inducing outputs and data which, in turn, leads to a complexity of $\mathcal{O}(NP^2)$, where

N is the number of data points and P the number of inducing points (Titsias, 2009). Unfortunately, this complexity is still prohibitive for large datasets. Therefore, we resort to an explicit representation of the variational distribution over inducing outputs as previously proposed in (Hensman et al., 2013). This explicit representation enables scalability by utilizing stochastic gradient-based optimization since individual GP predictions become independent given the explicit inducing points. Following a mean-field variational approximation the inducing output distribution is given as $q(\mathbf{z}) = \prod_{d=1}^{D_x} \mathcal{N}(\mathbf{z}_d | \mu_d, \Sigma_d)$ for each latent state dimension d with diagonal variance Σ_d . Marginalizing out the inducing outputs, the GP predictive distribution is obtained as Gaussian with mean and variance given by

$$\begin{aligned} \mu &= m_{\hat{\mathbf{x}}_t} + \alpha(\hat{\mathbf{x}}_t)(\mu_d - m_{\zeta_d}), \\ \sigma^2 &= k_{\hat{\mathbf{x}}_t, \hat{\mathbf{x}}_t} - \alpha(\hat{\mathbf{x}}_t)(K_{\zeta_d, \zeta_d} - \Sigma_d)\alpha(\hat{\mathbf{x}}_t)^T, \\ \alpha(\hat{\mathbf{x}}_t) &:= k_{\hat{\mathbf{x}}_t, \zeta_d} K_{\zeta_d, \zeta_d}^{-1}. \end{aligned} \quad (12)$$

4.2. Variational Approximation

In previous work (Mattos et al., 2015), a factorized variational distribution is considered based on a mean-field approximation for the latent states $\mathbf{x}_{1:T}$. Their variational distribution is given by

$$q(\mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z}) = \left[\prod_{d=1}^{D_x} q(\mathbf{z}_d) \left[\prod_{t=2}^T p(f_{t,d} | \hat{\mathbf{x}}_{t-1}, \mathbf{z}_d) \right] \right] \left[\prod_{t=1}^T q(\mathbf{x}_t) \right].$$

This choice, however, leads to several caveats: (i) The number of model parameters grows linearly with the length of the time series since each latent state is parametrized by its individual distribution $q(\mathbf{x}_t)$ for every time step. (ii) Initializing the latent state is non-trivial since the true, underlying observation mapping is generally unknown and non-bijective. (iii) The model design does not represent correlations between time steps. Instead, these correlations are only introduced by enforcing pairwise couplings during the optimization process. The first two problems have been addressed in (Mattos et al., 2015; Eleftheriadis et al., 2017) by introducing a *recognition* model, e.g. a Bi-RNN³, which acts as a smoother which can be learned through backpropagation and which allows to obtain the latent states given the input/output sequence.

The issue of representing correlations between time steps, however, is currently an open problem which we aim to address with our proposed model structure. Properly representing these correlations is a crucial step in making the

³A bi-directional RNN operates on a sequence from left to right and vice versa to obtain predictions based on past and future inputs.

optimization problem tractable in order to learn GP-SSMs for complex systems.

For PR-SSM, the variational distribution is given by

$$q(\mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z}) = \prod_{t=2}^T \left[p(\mathbf{x}_t | \mathbf{f}_t) \prod_{d=1}^{D_x} [p(f_{t,d} | \hat{\mathbf{x}}_{t-1}, \mathbf{z}_d) q(\mathbf{z}_d)] \right] q(\mathbf{x}_1), \quad (13)$$

with

$$q(\mathbf{x}_1) = \mathcal{N}(\mathbf{x}_1 | \boldsymbol{\mu}_{x_1}, \boldsymbol{\Sigma}_{x_1}), q(\mathbf{z}_d) = \mathcal{N}(\mathbf{z}_d | \boldsymbol{\mu}_d, \boldsymbol{\Sigma}_d).$$

In contrast to previous work, the proposed variational distribution does not factorize over the latent state but takes into account the true transition model, based on the sparse GP approximation from (8). In previous work, stronger approximations have been required to achieve an analytically tractable ELBO. This work, however, deals with the more complex distribution by combining sampling and gradient-based methods.

In (Frigola et al., 2014), the variational distribution over inducing outputs has been optimally eliminated. This leads to a smoothing problem in a second system requiring computationally expensive, e.g. sample-based, smoothing methods. Instead, we approximate the distribution by a Gaussian, which is the optimal solution in case of sparse GP regression (cf. (Titsias, 2009)).

The PR-SSM model parameters include the variational parameters for the initial state and inducing outputs and hyperparameters, such as inducing inputs, noise parameters and GP kernel parameters: $\theta_{\text{PR-SSM}} = (\boldsymbol{\mu}_{x_1}, \boldsymbol{\Sigma}_{x_1}, \boldsymbol{\mu}_{1:D_x}, \boldsymbol{\Sigma}_{1:D_x}, \boldsymbol{\zeta}_{1:D_x}, \sigma_{x,1:D_x}^2, \sigma_{y,1:D_y}^2, \theta_{\text{GP},1:D_x})$. Note that in the PR-SSM, the number of parameters grows only with the number of latent dimensions, but not with the length of the time series.

4.3. Variational Evidence Lower Bound

Following standard variational inference techniques (Blei et al., 2017), the ELBO is given by

$$\begin{aligned} \log p(\mathbf{y}_{1:T}) &\geq \mathbb{E}_{q(\mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z})} \left[\log \frac{p(\mathbf{y}_{1:T}, \mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z})}{q(\mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z})} \right] \\ &=: \mathcal{L}_{\text{PR-SSM}}. \end{aligned} \quad (14)$$

Maximizing the ELBO is equivalent to minimizing $\text{KL}(q(\mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z}) \| p(\mathbf{x}_{1:T}, \mathbf{f}_{2:T}, \mathbf{z} | \mathbf{y}_{1:T}))$ (Blei et al., 2017), therefore this is a way to optimize the approximated model parameter distribution with respect to the intractable, true model parameter posterior.

Based on (8) and (13) and using standard variational calcu-

lus, the ELBO (14) can be transformed into

$$\begin{aligned} \mathcal{L}_{\text{PR-SSM}} &= \sum_{t=1}^T \mathbb{E}_{q(\mathbf{x}_t)} [\log p(\mathbf{y}_t | \mathbf{x}_t)] \\ &\quad - \sum_{d=1}^{D_x} \text{KL}(q(\mathbf{z}_d) \| p(\mathbf{z}_d; \boldsymbol{\zeta}_d)), \end{aligned} \quad (15)$$

with $q(\mathbf{x}_t)$ defined in Sec. 4.4. The first part is the expected log-likelihood of the observed system outputs \mathbf{y} based on the observation model and the variational latent state distribution $q(\mathbf{x}_t)$. This term captures the capability of the learned latent state model to explain the observed system behavior. The second term is a regularizer on the inducing output distribution that penalizes deviations from the GP prior. Due to this term, PR-SSM automatically trades off data fit against model complexity. A detailed derivation of the ELBO can be found in the supplementary material.

4.4. Stochastic Gradient ELBO Optimization

Training the proposed PR-SSM requires maximizing the ELBO in (15) with respect to the model parameters $\theta_{\text{PR-SSM}}$. While the second term, as KL between two Gaussian distributions, can be easily computed, the first term requires evaluation of an expectation with respect to the latent state distribution $q(\mathbf{x}_t)$, given by

$$\begin{aligned} q(\mathbf{x}_t) &= \int \prod_{\tau=2}^t [p(\mathbf{x}_\tau | \mathbf{f}_\tau) p(\mathbf{f}_\tau | \hat{\mathbf{x}}_{\tau-1}, \mathbf{z})] \\ &\quad q(\mathbf{x}_1) q(\mathbf{z}) d\mathbf{x}_{1:t-1} d\mathbf{f}_{2:t} d\mathbf{z}. \end{aligned} \quad (16)$$

Since the true non-linear, latent dynamics is maintained in the variational approximation (13), analytic evaluation of (16) is still intractable. Because of the latent state's Markovian structure, the marginal latent state distribution $q(\mathbf{x}_t)$ at time t is *conditionally* independent of past time steps, given the previous state distribution $q(\mathbf{x}_{t-1})$ and the explicit representation of GP inducing points. This enables a differentiable, sampling-based estimation of the expectation term. Samples $\tilde{\mathbf{x}}_t$ from (16) can be obtained by recursively drawing from the sparse GP posterior in (12) for $t = 1, \dots, T$. Drawing samples from a Gaussian distribution can be made differentiable with respect to its parameters μ_d, σ_d^2 using the *re-parametrisation trick* (Kingma & Welling, 2013). The gradient can be propagated back through time due to this re-parametrization and unrolling of the latent state. An unbiased estimator of the first term in the ELBO in (15) is given by

$$\mathbb{E}_{q(\mathbf{x}_t)} [\log p(\mathbf{y}_t | \mathbf{x}_t)] \approx \frac{1}{N} \sum_{i=1}^N \log p(\mathbf{y}_t | \tilde{\mathbf{x}}_t^{(i)}). \quad (17)$$

Based on the stochastic ELBO evaluation, analytic gradients of (15) can be derived to facilitate stochastic gradient-descent-based model optimization.

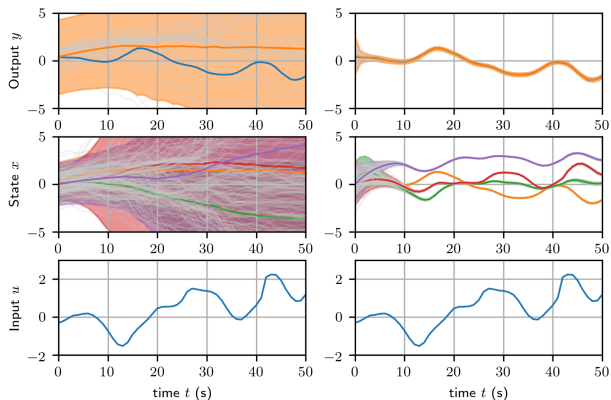


Figure 2. Predictions of the initial, untrained (left) and the final, trained PR-SSM (right) based on the full gradient ELBO optimization. The system input/output data (blue) is visualized together with the model prediction (orange) for a part of the *Furnace* dataset. Samples of the latent state distribution and output distribution are shown in gray. The colored, shaded areas visualize mean \pm two std of the latent state x and observation y . The initial model exhibits a random walk behavior in the latent space. In the trained model, the decay of the initial state uncertainty can be observed in the first time steps. In this experiment, no recognition model has been used (cf. Sec. 5).

4.5. Model Predictions

After model optimization based on the ELBO (15), model predictions for a new input sequence $\mathbf{u}_{1:T}$ and initial latent state \mathbf{x}_1 can be obtained based on the approximate, variational posterior distribution in (13). In contrast to (Mattos et al., 2015), no approximations such as moment matching are required for model predictions. Instead, the complex latent state distribution is approximated based on samples from (16). The predicted observation distribution can then be computed from the latent distribution according to the observation model in (9). Instead of a fixed, uninformative initial latent state, a learned recognition model (cf. Sec. 5 for details) can be utilized to find a more informative model initialization.

5. Extensions for Large Datasets

Optimizing the ELBO (15) based on the full gradient is prohibitive for large datasets and long trajectories. Instead, a stochastic optimization scheme based on mini-batches of sub-trajectories is introduced.

Directly optimizing the initial latent state distribution $q(\mathbf{x}_1)$ for each sub-trajectory would lead to a full parametrization of the latent state which is undesirable as described in Sec. 4.2. Instead, we propose a parametric recognition model, which initializes the latent state $q(\mathbf{x}_1)$. In recent work on SSMs (Mattos et al., 2015; Eleftheriadis et al., 2017), a recognition model is introduced to parametrize the smoothing distribution $p(\mathbf{x}_{1:T} | \mathbf{y}_{1:T}, \mathbf{u}_{1:T})$. We propose a

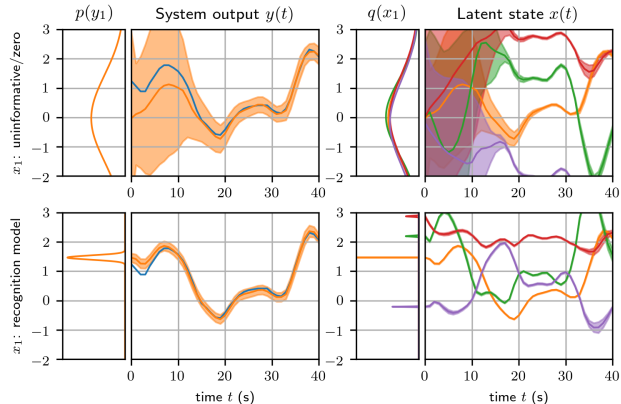


Figure 3. Comparison of the fully trained PR-SSM predictions with (lower row) and without (upper row) initial state recognition model on a test dataset. The initial transient based on the uncertainty from an uninformative initial state distribution $q(\mathbf{x}_1) = \mathcal{N}(\mathbf{x}_1 | \mathbf{0}, \mathbf{I})$ decays, as shown in upper plots. Below the outcome is shown when $q(\mathbf{x}_1)$ is initialized by the smoothing distribution $q(\mathbf{x}_1 | \mathbf{y}_{1:L}, \mathbf{u}_{1:L})$, given the first L steps of system input/output. Using the recognition model yields a significantly improved latent state initialization and therefore decreases the initial state uncertainty and the initial transient behavior.

similar approach, but only to model the initial latent state

$$q(\mathbf{x}_1) = \mathcal{N}(\mathbf{x}_1 | \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \approx q(\mathbf{x}_1 | \mathbf{y}_{1:L}, \mathbf{u}_{1:L}), \quad (18)$$

$$\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1 = h(\mathbf{y}_{1:L}, \mathbf{u}_{1:L}; \theta_{\text{recog}}). \quad (19)$$

The initial latent state distribution is approximated by a Gaussian, where mean and variance are modeled by a recognition model h . The recognition model acts as a smoother, operating on the first L elements of the system input/output data to infer the first latent state. Instead of directly optimizing $q(\mathbf{x}_1)$ during training, errors are propagated back into the recognition model h , which is parametrized by θ_{recog} .

Additionally, the proposed recognition model can also be used to predict behavior on test data, where the initial latent state is not known.

6. Experimental Evaluation

In the following, we present insights into the PR-SSM optimization schemes, comparisons to state-of-the-art model learning methods and a large scale experiment.

6.1. PR-SSM Learning

For small datasets (i.e. short training trajectory lengths), the model can be trained based on the full gradient of the ELBO in (15). A comparison of the model predictions before and after training with the full ELBO gradient is shown in Fig. 2.

Empirically, three major shortcomings of the full gradient-based optimization schemes are observed: (i) Computing the full gradient for long trajectories is expensive and prone

Table 1. Comparison of model learning methods on five real-world benchmark examples. The RMSE result (mean (std) over 5 independently learned models) is given for the free simulation on the test dataset. The best result (solid underline) and second best result (dashed underline) is indicated. The proposed PR-SSM consistently outperforms the reference (SS-GP-SSM) in the class of Markovian state space models and robustly achieves performance comparable to the one of state-of-the-art latent, autoregressive models. Best viewed in color.

TASK	ONE-STEP-AHEAD, AUTOREGRESSIVE		MULTI-STEP-AHEAD, LATENT SPACE AUTOREGRESSIVE			MARKOVIAN STATE-SPACE MODELS	
	GP-NARX	NIGP	REVARB 1	REVARB 2	MSGP	SS-GP-SSM	PR-SSM
ACTUATOR	0.627 (0.005)	0.599 (0)	<u>0.438 (0.049)</u>	0.613 (0.190)	<u>0.771 (0.098)</u>	0.696 (0.034)	<u>0.502 (0.031)</u>
BALLBEAM	0.284 (0.222)	<u>0.087 (0)</u>	0.139 (0.007)	0.209 (0.012)	0.124 (0.034)	411.6 (273.0)	<u>0.073 (0.007)</u>
DRIVES	0.701 (0.015)	<u>0.373 (0)</u>	<u>0.828 (0.025)</u>	<u>0.868 (0.113)</u>	<u>0.451 (0.021)</u>	0.718 (0.009)	0.492 (0.038)
FURNACE	<u>1.201 (0.000)</u>	1.205 (0)	<u>1.195 (0.002)</u>	<u>1.188 (0.001)</u>	<u>1.277 (0.127)</u>	<u>1.318 (0.027)</u>	1.249 (0.029)
DRYER	0.310 (0.044)	0.268 (0)	<u>0.851 (0.011)</u>	0.355 (0.027)	<u>0.146 (0.004)</u>	0.152 (0.006)	<u>0.140 (0.018)</u>
SARCOS	<u>0.169 (-)</u>	N.A.	N.A.	N.A.	N.A.	N.A.	<u>0.049 (-)</u>

to the well-known problems of exploding and vanishing gradients (Pascanu et al., 2013). (ii) An uninformative initial state is prohibitive for unstable systems or systems with slowly decaying initial state transients. (iii) Momentum-based optimizers (e.g. Adam) exhibit fragile optimization performance and are prone to overfitting.

The proposed method addresses these problems by employing the stochastic ELBO gradient based on minibatches of sub-trajectories and the initial state recognition model (cf. Sec. 5). Fig. 3 visualizes the initial state distribution $q(x_1)$ and the corresponding predictive output distribution $p(y_1)$ for the fully trained model based on the full gradient (top row), as well as for the model based on the stochastic gradient and recognition model (bottom row). The transient dynamics and the associated model uncertainty is clearly visible for the first 15 time steps until the initial transient decays and approaches the true system behavior. In contrast, the learned recognition model almost perfectly initializes the latent state, leading to much smaller deviations in the predicted observations and far less predictive uncertainty. Notice how the recognition model is most certain about the distribution of the first latent state dimension (orange), which is directly coupled to the observation through the parametric observation model (cf. (9)). The uncertainty for the remaining, latent states, in contrast, is slightly higher.

Comparing the full ELBO gradient-based model learning and the stochastic version with the recognition model, the stochastic model learning is far more robust and counteracts the overfitting tendencies in the full gradient-based model learning. A comparison of the model learning progress for both methods is depicted in the supplementary material. Due to the improved optimization robustness and the applicability to larger datasets, the stochastic, recognition-model-based optimization scheme is employed for the model learning benchmark presented in the next section. Empirically, the cost of the proposed sampling scheme is much lower

compared to methods employing SMC for sampling the full model posterior. In the experiments, 50 latent state samples were employed (details in the supplementary material).

6.2. Model Learning Benchmark

The performance of PR-SSM is assessed in comparison to state-of-the-art model learning methods on several real-world datasets as previously utilized by (Mattos et al., 2015). The suite of reference methods is composed of: One-step ahead autoregressive GP models: GP-FITC (Snelson & Ghahramani, 2006) and NIGP (McHutchon & Rasmussen, 2011). Multi-step-ahead autoregressive and recurrent GP models in latent space: REVARB based on 1, respectively 2, hidden layers (Mattos et al., 2015) and MSGP (Doerr et al., 2017a). GP-SSMs, based on a full Markovian state: SS-GP-SSM (Svensson & Schön, 2017) and the proposed PR-SSM. Currently, no published and runnable code exists for the model learning frameworks presented in (Turner et al., 2010; Frigola et al., 2013; 2014; Eleftheriadis et al., 2017). To enable a fair comparison of the methods’ performance and robustness, whitened data, a default configuration across tasks and a predefined amount of input/output data for initialization is employed. The presented results are therefore not directly comparable to previous work, where different data pre/postprocessing or method configurations are employed. A more thorough evaluation, which matches the published results from previous work, as well as experimental details are given in the supplementary material.

The benchmark results are summarized in Tab. 1. A detailed visualization of the resulting model predictions on the *Drives* dataset is shown in Fig. 4. For the one-step-ahead models (GP-NARX, NIGP), two variants are used to obtain long-term predictive distributions: Propagating the mean (no uncertainty propagation) and approximating the true posterior by a Gaussian using exact moment matching (Girard et al., 2003). The results show that PR-SSM consistently

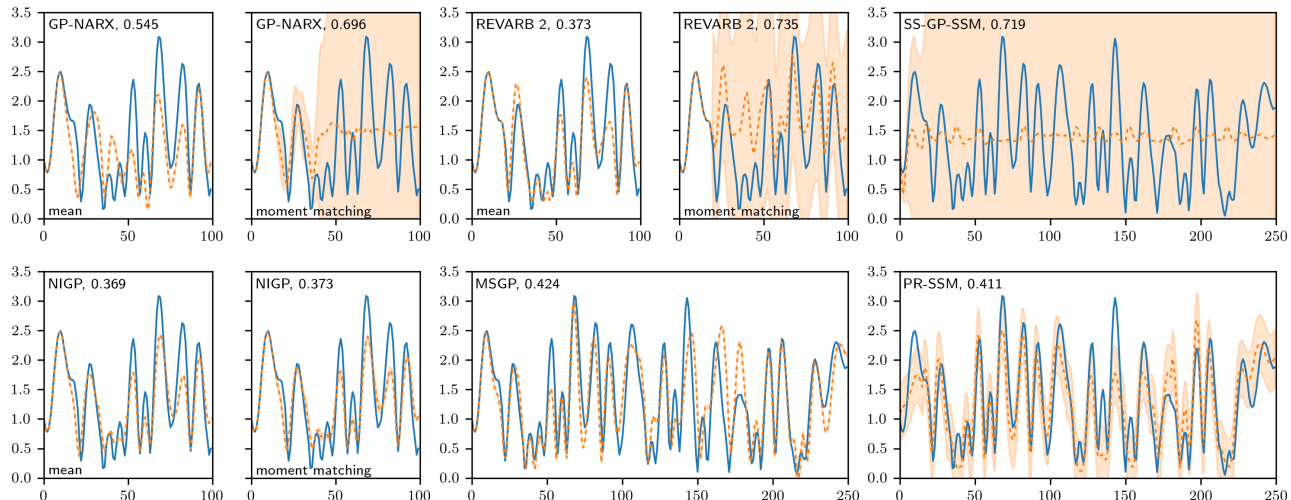


Figure 4. Free simulation results for the benchmark methods on the *Drives* test dataset. The true, observed system output (blue) is compared to the individual model’s predictive output distribution (orange, mean \pm two std). Results are presented for the one-step-ahead models GP-NARX and NIGP in the left column. REVARB and MSGP (shown in the middle column) are both based on multi-step optimized autoregressive GP models in latent space. In the right column, the SS-GP-SSMs, as a model based on a Markovian latent state, is compared to the proposed PR-SSM.

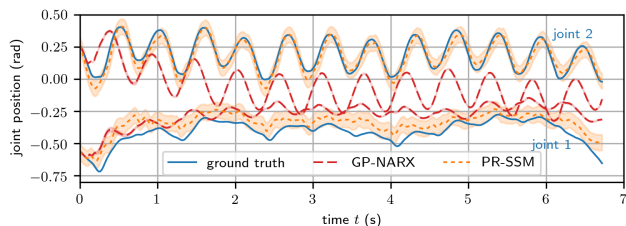


Figure 5. Results on the Sarcos large scale task: Predictions from the GP-NARX baseline (red) and the PR-SSM (orange) for two out of seven joint positions. The ground truth, measured joint positions are shown in blue. PR-SSM clearly improves over the GP-NARX predictions. Similar results are obtained for PR-SSM on the remaining 5 joints, where the GP-NARX model fails completely (cf. supplementary materials for details).

outperforms the SS-GP-SSM learning method. Similarly, performance is improved in comparison to baseline methods (GP-NARX and NIGP). In the ensemble of models based on long-term optimized autoregressive structure (REVARB, MSGP), no method is clearly superior. However, the performance of PR-SSM is consistently strong. The probabilistic methods results are competitive or improve over the performance of deterministic RNN/LSTM models, as shown in (Mattos et al., 2015). Note that PR-SSM demonstrates robust model learning performance across all datasets.

6.3. Large Scale Experiment

To evaluate the scalability, results are provided for the forward dynamics model of the SARCOS 7 degree of freedom robotic arm. The task is characterized by 60 experiments of length 337 ($\approx 20,000$ datapoints), 7 input, and 7 output

dimensions. PR-SSM is set up with a latent state dimensionality $D_x = 14$. From the set of reference methods, only GP-NARX can be adapted to run efficiently on this dataset without major effort (details are given in the supplementary material). A visualization of the model predictions is shown in Fig 5 and prediction RMSEs are listed in Tab. 1. The results show that PR-SSM is able to learn robustly and accurately for all system outputs from all experimental data. In contrast, the GP-NARX baseline achieves worse predictions and fails to predict the remaining five joints (not shown).

7. Conclusion

In this work, we presented Probabilistic Recurrent State-Space Models (PR-SSM) as a novel model structure and efficient inference scheme for learning probabilistic, Markovian state-space models. Based on GP priors and doubly stochastic variational inference, a novel model optimization criterion is derived, which is closely related to the one of powerful, but deterministic, RNNs or LSTMs. By maintaining the true latent state distribution and thereby enabling long-term gradients, efficient inference in latent space becomes feasible. Furthermore, a novel recognition model enables learning of unstable or slow dynamics as well as scalability to large datasets. Robustness, scalability and high performance in model learning is demonstrated on real-world datasets in comparison to state-of-the-art methods.

A limitation of PR-SSM is its dependency on an a-priori fixed latent state dimensionality. This shortcoming could potentially be resolved by a sparsity enforcing latent state prior, which would suppress unnecessary latent state dimensions.

Acknowledgements

This research was supported in part by National Science Foundation grants IIS-1205249, IIS-1017134, EECS-0926052, the Office of Naval Research, the Okawa Foundation, the Max-Planck-Society, and the Cyber Valley initiative.

References

- Archer, E., Park, I. M., Buesing, L., Cunningham, J., and Paninski, L. Black box variational inference for state space models. *arXiv preprint arXiv:1511.07367*, 2015.
- Aström, K. J. and Murray, R. M. *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.
- Bayer, J. and Osendorfer, C. Learning stochastic recurrent networks. *arXiv preprint arXiv:1411.7610*, 2014.
- Billings, S. A. *Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains*. John Wiley & Sons, 2013.
- Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.
- Camacho, E. F. and Alba, C. B. *Model predictive control*. Springer Science & Business Media, 2013.
- Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A. C., and Bengio, Y. A recurrent latent variable model for sequential data. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 2980–2988, 2015.
- Deisenroth, M. P. and Rasmussen, C. E. PILCO: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pp. 465–472, 2011.
- Deisenroth, M. P., Turner, R. D., Huber, M. F., Hanebeck, U. D., and Rasmussen, C. E. Robust filtering and smoothing with gaussian processes. *IEEE Transactions on Automatic Control*, 57(7):1865–1871, 2012.
- Doerr, A., Daniel, C., Nguyen-Tuong, D., Marco, A., Schaal, S., Toussaint, M., and Trimpe, S. Optimizing long-term predictions for model-based policy search. In *Conference on Robot Learning (CORL)*, pp. 227–238, 2017a.
- Doerr, A., Nguyen-Tuong, D., Marco, A., Schaal, S., and Trimpe, S. Model-based policy search for automatic tuning of multivariate PID controllers. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2017b.
- Eleftheriadis, S., Nicholson, T., Deisenroth, M., and Hensman, J. Identification of Gaussian process state space models. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 5315–5325, 2017.
- Föll, R., Haasdonk, B., Hanselmann, M., and Ulmer, H. Deep recurrent Gaussian process with variational sparse spectrum approximation. *arXiv preprint arXiv:1711.00799*, 2017.
- Fracaro, M., Sønderby, S. K., Paquet, U., and Winther, O. Sequential neural models with stochastic layers. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 2199–2207, 2016.
- Frigola, R., Lindsten, F., Schön, T. B., and Rasmussen, C. E. Bayesian inference and learning in Gaussian process state-space models with particle MCMC. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 3156–3164, 2013.
- Frigola, R., Chen, Y., and Rasmussen, C. E. Variational Gaussian process state-space models. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 3680–3688, 2014.
- Frigola-Alcade, R. Bayesian time series learning with Gaussian processes. *University of Cambridge*, 2015.
- Gemici, M., Hung, C.-C., Santoro, A., Wayne, G., Mohamed, S., Rezende, D. J., Amos, D., and Lillicrap, T. Generative temporal models with memory. *arXiv preprint arXiv:1702.04649*, 2017.
- Girard, A., Rasmussen, C. E., Quinonero-Candela, J., Murray-Smith, R., Winther, O., and Larsen, J. Multiple-step ahead prediction for non linear dynamic systems: a Gaussian process treatment with propagation of the uncertainty. In *Advances in Neural Information Processing Systems (NIPS)*, volume 15, pp. 529–536, 2003.
- Hensman, J., Fusi, N., and Lawrence, N. D. Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*, 2013.
- Hochreiter, S. and Schmidhuber, J. LSTM can solve hard long time lag problems. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 473–479, 1997.
- Kalman, R. A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.
- Karl, M., Soelch, M., Bayer, J., and van der Smagt, P. Deep variational bayes filters: Unsupervised learning of state space models from raw data. In *International Conference on Learning Representations*, 2016.

- Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- Ko, J. and Fox, D. GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models. *Autonomous Robots*, 27(1):75–90, 2009.
- Krishnan, R. G., Shalit, U., and Sontag, D. Deep Kalman filters. *arXiv preprint arXiv:1511.05121*, 2015.
- Likar, B. and Kocijan, J. Predictive control of a gas–liquid separation plant based on a Gaussian process model. *Computers & chemical engineering*, 31(3):142–152, 2007.
- Littman, M. L. and Sutton, R. S. Predictive representations of state. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1555–1561, 2002.
- Ljung, L. Perspectives on system identification. *Annual Reviews in Control*, 34(1):1–12, 2010.
- Mattos, C. L. C., Dai, Z., Damianou, A., Forth, J., Barreto, G. A., and Lawrence, N. D. Recurrent Gaussian processes. *arXiv preprint arXiv:1511.06644*, 2015.
- Mattos, C. L. C., Damianou, A., Barreto, G. A., and Lawrence, N. D. Latent autoregressive Gaussian processes models for robust system identification. *IFAC-PapersOnLine*, 49(7):1121–1126, 2016.
- McHutchon, A. and Rasmussen, C. E. Gaussian process training with input noise. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1341–1349, 2011.
- Murray-Smith, R. and Girard, A. Gaussian process priors with ARMA noise models. In *Irish Signals and Systems Conference, Maynooth*, pp. 147–152, 2001.
- Pascanu, R., Mikolov, T., and Bengio, Y. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pp. 1310–1318, 2013.
- Rudary, M. R. and Singh, S. P. A nonlinear predictive state representation. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 855–862, 2004.
- Salimbeni, H. and Deisenroth, M. Doubly stochastic variational inference for deep Gaussian processes. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 4591–4602, 2017.
- Singh, S. P., Littman, M. L., Jong, N. K., Pardoe, D., and Stone, P. Learning predictive state representations. In *Proceedings of the 20th International Conference on Machine Learning (ICML)*, pp. 712–719, 2003.
- Snelson, E. and Ghahramani, Z. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems (NIPS)*, volume 18, pp. 1257, 2006.
- Sutskever, I., Martens, J., and Hinton, G. E. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pp. 1017–1024, 2011.
- Svensson, A. and Schön, T. B. A flexible state–space model for learning nonlinear dynamical systems. *Automatica*, 80:189–199, 2017.
- Titsias, M. K. Variational learning of inducing variables in sparse Gaussian processes. In *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 5, pp. 567–574, 2009.
- Turner, R., Deisenroth, M., and Rasmussen, C. State-space inference and learning with Gaussian processes. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 868–875, 2010.
- Van Overschee, P. and De Moor, B. *Subspace identification for linear systems: Theory - Implementation - Applications*. Springer Science & Business Media, 2012.
- Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., and Saenko, K. Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729*, 2014.
- Wang, J. M., Fleet, D. J., and Hertzmann, A. Gaussian process dynamical models for human motion. *IEEE transactions on pattern analysis and machine intelligence*, 30(2): 283–298, 2008.
- Watter, M., Springenberg, J., Boedecker, J., and Riedmiller, M. Embed to control: A locally linear latent dynamics model for control from raw images. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 2746–2754, 2015.
- Williams, C. K. and Rasmussen, C. E. *Gaussian processes for machine learning*. MIT Press, 2005.