Automating probabilistic modeling
of dynamical systems and their surroundings

Thomas Schön
Division of Systems and Control
Department of Information Technology
Uppsala University.

Email: thomas.schon@it.uu.se,
www: user.it.uu.se/~thosc112

Machine learning develops methods for machines to improve their performance at certain tasks based on data.
Background – what we do in the team

We automate the extraction of knowledge and understanding from data.

Both basic research and applied research (with companies).

Create probabilistic models for dynamical systems and their surroundings.

Develop methods to learn models from data.

The models can then be used by machines (or humans) to understand and/or take decisions about what will happen next.
Probabilistic modeling allow for representing and manipulating uncertainty in data, models, decisions and predictions.

Probability theory explains how to mathematically represent and manipulate uncertainty.

Nice introduction to probabilistic modeling in Machine Learning

Probabilistic modeling of dynamical systems

A state space model (SSM) consists of an unknown state \( \{x_t\}_{t \geq 1} \) that is indirectly observed via a measurement process \( \{y_t\}_{t \geq 1} \).

A **parametric** state space model is given by:

\[
\begin{align*}
    x_{t+1} | x_t & \sim p_\theta(x_{t+1} | x_t, u_t), \\
    y_t | x_t & \sim p_\theta(y_t | x_t, u_t), \\
    x_1 & \sim p_\theta(x_1), \\
    \theta & \sim p(\theta).
\end{align*}
\]

where

\[
\begin{align*}
    x_{t+1} &= f_\theta(x_t, u_t) + v_{\theta,t}, \\
    y_t &= g_\theta(x_t, u_t) + e_{\theta,t}, \\
    x_1 &\sim p_\theta(x_1), \\
    \theta &\sim p(\theta).
\end{align*}
\]

The **full probabilistic model** is given by

\[
p(x_{1:T}, \theta, y_{1:T}) = \underbrace{p(y_{1:T} \mid x_{1:T}, \theta)}_{\text{data distribution}} \underbrace{p(x_{1:T}, \theta)}_{\text{prior}}
\]
Example – “what are $x_t$, $\theta$ and $y_t$”?

**Aim (motion capture):** Compute $x_t$ (position and orientation of the different body segments) of a person ($\theta$ describes the body shape) moving around indoors using measurements $y_t$ (accelerometers, gyroscopes and ultrawideband).

---


Probabilistic modeling of dynamical systems

Distribution describing a parametric nonlinear state space model:

\[
p(x_{1:T}, \theta, y_{1:T}) = \prod_{t=1}^{T} p(y_t | x_t, \theta) \prod_{t=1}^{T-1} p(x_{t+1} | x_t, \theta) \, p(x_1 | \theta) \, p(\theta)
\]

Model = probability distribution!
Use flexible models

Key lesson from modern Machine Learning:

Flexible models often gives the best performance.

How can we build flexible models?

1. Models that use a large (but fixed) number of parameters compared with the data set. (parametric, ex. deep learning)

2. Models that use more parameters as we get access to more data. (non-parametric, ex. Gaussian process)
**Gaussian Process (GP) regression**

GP regression to find a static one dim. nonlinear function $h(z)$.

This GP model was used as a component in a larger model, see Fredrik Lindsten, Thomas B. Schön and Michael I. Jordan. Bayesian semiparametric Wiener system identification. *Automatica*, 49(7): 2053-2063, July 2013.
Gaussian Process nonlinear state space model

Consider the Gaussian Process SSM (GP-SSM):

\[
\begin{align*}
    x_{t+1} &= f(x_t) + v_t, \quad \text{s.t.} \quad f(x) \sim \mathcal{GP}(0, \kappa_\theta, f(x,x')),
    
    y_t &= g(x_t) + e_t, \quad \text{s.t.} \quad g(x) \sim \mathcal{GP}(0, \kappa_\theta, g(x,x')).
\end{align*}
\]

The model functions \( f \) and \( g \) are assumed to be realizations from Gaussian process priors and \( v_t \sim \mathcal{N}(0, Q), e_t \sim \mathcal{N}(0, R) \).

We can now find the posterior distribution

\[
p(f, g, Q, R, \theta \mid y_{1:T}),
\]

by making use of particle MCMC type algorithms.


Results in a **flexible** non-parametric model where the GP prior on $f$ takes on the role of a regularizer.

Provides a data-driven way of **tuning** the model flexibility.

**Toy example:**

\[
x_{t+1} = -10 \frac{x_t}{1 + 3x_t^2} + v_t,
\]

\[
y_t = x_t + e_t.
\]

Andreas Svensson and Thomas B. Schön. **A flexible state space model for learning nonlinear dynamical systems**, *Automatica*, 2017. (Accepted for publication)
Outline and message

1. What we do in the team.
2. Probabilistic modeling of dynamical systems
3. Key lesson: the use of flexible models

   The Gaussian process SSM

4. **Modeling the surroundings**

5. Inference and learning

6. The ASSEMBLE project (automate modeling and learning)

Probabilistic modeling allows us to systematically represent and manipulate uncertainty.
Modeling the surroundings – an example

The Earth’s magnetic field sets a background for the ambient magnetic field. Deviations make the field vary from point to point.

**Aim:** Build a map (i.e., a model) of the magnetic environment based on measurements from magnetometers.

**Solution:** Customized Gaussian process that obeys Maxwell’s equations.

[Video link](www.youtube.com/watch?v=enlMiUqPVJo)

Modeling the surroundings

Probabilistic graphical models naturally opens up for more general models also including the surroundings.

Spatio-temporal models.

“The nonlinear SSM is just a special case.”

The learning problem (dynamical systems)

Compute the posterior distribution

\[
p(x_{1:T}, \theta \mid y_{1:T}) = p(x_{1:T} \mid \theta, y_{1:T}) p(\theta \mid y_{1:T})\].

HD integration/optimization problems without analytical solution.

Sequential Monte Carlo provide approximations to integration problems where there is a \textit{sequential structure} present.

Variational approximations also highly relevant.

Currently this calls for the derivation and implementation of \textit{tailored} learning algorithms for each problem (too expensive).
ASSEMBLE project

**Aim:** Automate probabilistic modeling of dynamical systems (and their surroundings) via a formally defined **probabilistic modeling language**.

We will create a **market place** for inference and learning algorithms and probabilistic model libraries for dynamical systems and their environments.
A flexible model often gives the best performance.

Spatio-temporal models include the surroundings.

The resulting learning problems require approximations.

Huge need for automation (ASSEMBLE project).

Hosting the SMC workshop in Uppsala, Aug. 30-Sep. 1, 2017.

Preliminary (for now) web site:

http://www.it.uu.se/conferences/smc2017
References to some of our work

Tutorial on probabilistic modeling of nonlinear dynamical systems:


SMC for graphical models


Fredrik Lindsten, Adam M. Johansen, Christian A. Naesseth, Bonnie Kirkpatrick, Thomas B. Schön, John Aston and Alexandre Bouchard-Côté. Divide-and-Conquer with Sequential Monte Carlo. Journal of Computational and Graphical Statistics (JCGS), 2016. (Accepted for publication)

Particle Gibbs with ancestor sampling construction


Financial support from the Swedish Research Council (VR) and the Swedish Foundation for Strategic research (SSF) is gratefully acknowledged.

Thomas Schön