Welcome to Sequential Monte Carlo methods!!

Lecture 1 – Introduction and probabilistic modelling

Johan Alenlöv, Linköping University

2025-02-03

Aim of this course

Aim: To provide an introduction to the theory and application of sequential Monte Carlo (SMC) methods.

Aim of this course

Aim: To provide an introduction to the theory and application of sequential Monte Carlo (SMC) methods.

After the course you should be able to derive your own SMC-based algorithms allowing you to do inference in nonlinear models.

Aim of this course

Aim: To provide an introduction to the theory and application of sequential Monte Carlo (SMC) methods.

After the course you should be able to derive your own SMC-based algorithms allowing you to do inference in nonlinear models.

Day 1-3: Focus on state space models (SSMs). How to learning them from data and how to estimate their hidden states.

Day 4-5: Using SMC for inference in general probabilistic models.

Ex) Indoor positioning (engineering)

Aim: Compute the **position** of a person moving around indoors using variations in the ambient magnetic field and the motion of the person (acceleration and angular velocities). All of this observed using sensors in a standard smartphone.

Ex) Indoor positioning (engineering)

Aim: Compute the **position** of a person moving around indoors using variations in the ambient magnetic field and the motion of the person (acceleration and angular velocities). All of this observed using sensors in a standard smartphone.

Ex) Indoor positioning (engineering)

Key ingredients of the solution:

1. The particle filter for computing the position
2. The Gaussian process for building and representing the map of the ambient magnetic field
3. Inertial sensor signal processing

Ex) Indoor positioning (engineering)

Key ingredients of the solution:

1. The particle filter for computing the position
2. The Gaussian process for building and representing the map of the ambient magnetic field
3. Inertial sensor signal processing

Movie – map making: www.youtube.com/watch?v=enlMiUqPVJo

Movie – indoor positioning result

Ex) Indoor positioning (engineering)

Key ingredients of the solution:

1. The particle filter for computing the position
2. The Gaussian process for building and representing the map of the ambient magnetic field
3. Inertial sensor signal processing

Movie – map making: www.youtube.com/watch?v=enlMiUqPVJo

Movie – indoor positioning result


Arno Solin, Simo Särkkä, Juho Kannala and Esa Rahtu. **Terrain navigation in the magnetic landscape: Particle filtering for indoor positioning.** *Proc. of the European Navigation Conf. (ENC)*, Helsinki, Finland, June, 2016.


Arno Solin, Manon Kok, Niklas Wahlström, Thomas B. Schön and Simo Särkkä. **Modeling and interpolation of the ambient magnetic field by Gaussian processes.** *IEEE Trans. on Robotics*, 34(4):1112–1127, 2018.


Carl Jidling, Niklas Wahlström, Adrian Wills and Thomas B. Schön. **Linearly constrained Gaussian processes.** *Advances in Neural Information Processing Systems (NIPS)*, Long Beach, CA, USA, December, 2017.

Ex) Epidemiological modelling (statistics)

Aim: To learn a model explaining the seasonal influenza epidemics and then make use of this model to compute predictions.

Ex) Epidemiological modelling (statistics)

Aim: To learn a model explaining the seasonal influenza epidemics and then make use of this model to compute predictions.

Susceptible/infected/recovered (SIR) model:

$$
\begin{aligned}
S_{t+dt} &= S_t + \mu\mathcal{P}dt - \mu S_t dt - (1 + F v_t)\beta_t S_t \mathcal{P}^{-1} I_t dt, \\
I_{t+dt} &= I_t - (\gamma + \mu) I_t dt + (1 + F v_t)\beta_t S_t \mathcal{P}^{-1} I_t dt, \\
R_{t+dt} &= R_t + \gamma I_t dt - \mu R_t dt, \\
\beta_t &= R_0(\gamma + \mu)(1 + \alpha \sin(2\pi t/12)),
\end{aligned}
$$

Measurements:

$$
y_k = \rho \operatorname{logit}(\overline{I}_k/\mathcal{P}) + e_k, \qquad e_k \sim \mathcal{N}(0, \sigma^2).
$$

Ex) Epidemiological modelling (statistics)

Aim: To learn a model explaining the seasonal influenza epidemics and then make use of this model to compute predictions.

Susceptible/infected/recovered (SIR) model:

$$
\begin{aligned}
S_{t+dt} &= S_t + \mu\mathcal{P}dt - \mu S_t dt - (1 + F v_t)\beta_t S_t \mathcal{P}^{-1} I_t dt, \\
I_{t+dt} &= I_t - (\gamma + \mu) I_t dt + (1 + F v_t)\beta_t S_t \mathcal{P}^{-1} I_t dt, \\
R_{t+dt} &= R_t + \gamma I_t dt - \mu R_t dt, \\
\beta_t &= R_0(\gamma + \mu)(1 + \alpha \sin(2\pi t/12)),
\end{aligned}
$$

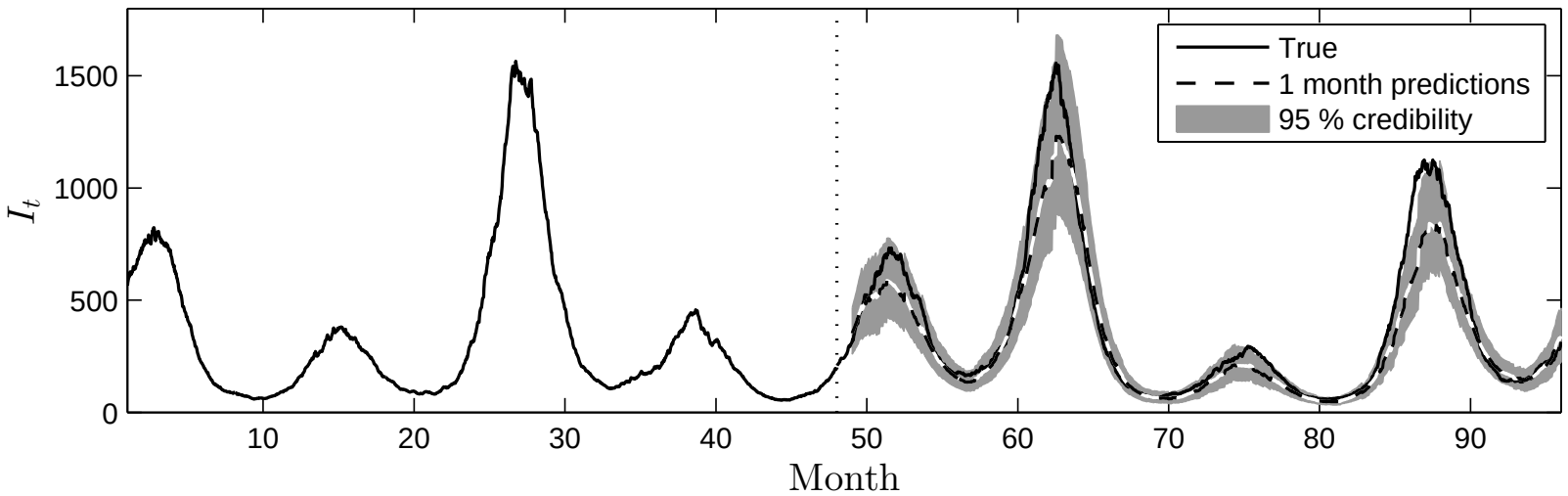
Measurements:

$$
y_k = \rho \operatorname{logit}(\bar{I}_k/\mathcal{P}) + e_k, \qquad e_k \sim \mathcal{N}(0, \sigma^2).
$$

Information about the unknown parameters $\theta = (\gamma, R_0, \alpha, F, \rho, \sigma)$ and states $x_t = (S_t, I_t, R_t)$ has to be learned from measurements.

Ex) Epidemiological modelling (statistics)

Compute $p(\theta, x_{1:T} \mid y_{1:T})$, where $y_{1:T} = (y_1, y_2, \ldots, y_T)$ and use it to compute the predictive distribution.



Disease activity (number of infected individuals I_t) over an eight year period.

Fredrik Lindsten, Michael I. Jordan and Thomas B. Schön. **Particle Gibbs with ancestor sampling.** *Journal of Machine Learning Research (JMLR)*, 15:2145-2184, June 2014.

Ex) Probabilistic programming (machine learning)

A probabilistic program encodes a probabilistic model according to the semantics of a particular probabilistic programming language, giving rise to a programmatic model.

Ex) Probabilistic programming (machine learning)

A probabilistic program encodes a probabilistic model according to the semantics of a particular probabilistic programming language, giving rise to a programmatic model.

The memory state of a running probabilistic program evolves dynamically and stochastically in time and so is a stochastic process.

Ex) Probabilistic programming (machine learning)

A probabilistic program encodes a probabilistic model according to the semantics of a particular probabilistic programming language, giving rise to a programmatic model.

The memory state of a running probabilistic program evolves dynamically and stochastically in time and so is a stochastic process.

SMC is a common inference method for programmatic model.

Creates a clear separation between the model and the inference methods. Opens up for the automation of inference!

```
x ~ Bernoulli(p);      assume(x)
if (x) {               value(x)
  y ~ N(0,1);          assume(y)
} else {
  y <- 0;
```

Ex) Probabilistic programming (machine learning)

A probabilistic program encodes a probabilistic model according to the semantics of a particular probabilistic programming language, giving rise to a programmatic model.

The memory state of a running probabilistic program evolves dynamically and stochastically in time and so is a stochastic process.

SMC is a common inference method for programmatic model.

Creates a clear separation between the model and the inference methods. Opens up for the automation of inference!

```
x ~ Bernoulli(p);      assume(x)
if (x) {               value(x)
  y ~ N(0,1);          assume(y)
} else {
  y <- 0;
```

More during lecture 17.

Course structure – overview

- 17 lectures (45 min. each)
- Credits offered: 6ECTS (Swedish system)
- Practicals (solve exercises and hand-in assignments, discuss and ask questions)
- Discussions (discuss concepts)
- Hand-in assignments. You can collaborate, but the reports with the solutions are individual.
  - One set to be done between day 2 and 3
  - One set to be done after day 5
- Complete course information (including lecture slides) is available from the course website: `https://www.ida.liu.se/divisions/stima/fokurser/smc2025/`
- Feel free to ask questions at any time!

The only way to really learn something is by implementing it on your own.

Outline – 5 days

Day 1 – Probabilistic modelling and particle filtering basics
  a) Probabilistic modelling of dynamical systems and filtering
  b) Introduce Monte Carlo and derive the bootstrap particle filter

Day 2 – Particle filtering
  a) Auxiliary particle filter, full adaptation and practicalities

Day 3 – Parameter learning
  a) Maximum likelihood parameter learning, convergence

Day 4 – Bayesian parameter learning
  a) Particle Metropolis Hastings
  b) Particle Gibbs

Day 5 – Beyond state space models (outlooks)
  a) General target sequences and SMC samplers
  b) Diffusion models

Outline – Lecture 1

Aim: Introduce the course and provide background on probabilistic modelling.

Outline:

1. Course introduction and practicalities
2. Probabilistic modelling
3. Key probabilistic objects
4. Ex. probabilistic autoregressive modelling (if there is time)

Probabilistic modelling

Modelling

Mathematical model: A compact representation—set of assumptions—of the data that in precise mathematical form captures the key properties of the underlying situation.

Modelling

Mathematical model: A compact representation—set of assumptions—of the data that in precise mathematical form captures the key properties of the underlying situation.

Most of the course (day 1-3) is concerned with dynamical phenomena. The methods are more general than that and during the last day we will broaden the scope significantly.

Dynamical phenomena produce temporal measurements (data) arriving as a **sequence**

$$y_{1:t} = (y_1, y_2, \ldots, y_t).$$

Modelling

Mathematical model: A compact representation—set of assumptions—of the data that in precise mathematical form captures the key properties of the underlying situation.

Most of the course (day 1-3) is concerned with dynamical phenomena. The methods are more general than that and during the last day we will broaden the scope significantly.

Dynamical phenomena produce temporal measurements (data) arriving as a **sequence**

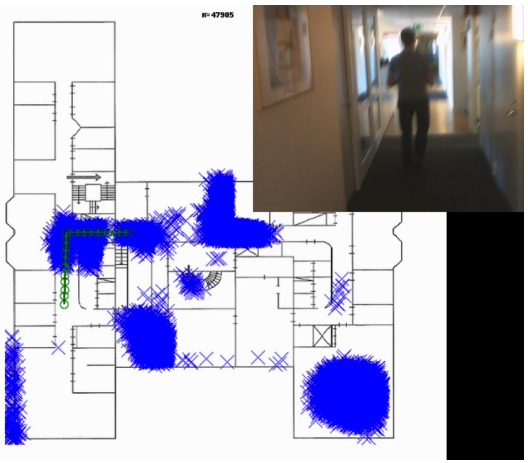
$$y_{1:t} = (y_1, y_2, \ldots, y_t).$$

Nice introduction to probabilistic modelling in Machine Learning

Ghahramani, Z. **Probabilistic machine learning and artificial intelligence**. *Nature* 521:452-459, 2015.

Representing and modifying uncertainty

It is important to maintain a solid representation of uncertainty in all mathematical objects and throughout all calculations.



The two basic rules from probability theory

Let X and Y be continuous random variables[1]. Let $p(\cdot)$ denote a general probability density function.

1. Marginalization (integrate out a variable): $p(x) = \int p(x, y) \mathrm{d}y$.
2. Conditional probability: $p(x, y) = p(x \mid y)p(y)$.

[1] Notation: Upper-case letters for random variables (r.v.) X when we talk about models. Lower-case letters for realizations of the r.v., $X = x$ and in algorithms. We will not use bold face for vectors.

The two basic rules from probability theory

Let X and Y be continuous random variables[1]. Let $p(\cdot)$ denote a general probability density function.

1. Marginalization (integrate out a variable): $p(x) = \int p(x, y)\mathrm{d}y$.
2. Conditional probability: $p(x, y) = p(x \mid y)p(y)$.

Combine them into Bayes' rule:

$$p(x \mid y) = \frac{p(y \mid x)p(x)}{p(y)} = \frac{p(y \mid x)p(x)}{\int p(y \mid x)p(x)\mathrm{d}x}.$$

[1] Notation: Upper-case letters for random variables (r.v.) X when we talk about models. Lower-case letters for realizations of the r.v., $X = x$ and in algorithms. We will not use bold face for vectors.

Basic variables classes

Measurements $y_{1:T} = (y_1, y_2, \ldots, y_T)$: The measured data somehow obtained from the phenomenon we are interested in.

Basic variables classes

Measurements $y_{1:T} = (y_1, y_2, \ldots, y_T)$: The measured data somehow obtained from the phenomenon we are interested in.

Unknown (static) model parameters θ: Describes the model, but unknown (or not known well enough) to the user.

Basic variables classes

Measurements $y_{1:T} = (y_1, y_2, \ldots, y_T)$: The measured data somehow obtained from the phenomenon we are interested in.

Unknown (static) model parameters θ: Describes the model, but unknown (or not known well enough) to the user.

Unknown model variables x_t (changing over time): Describes the state of the phenomenon at time t (in the indoor positioning example above x_t includes the unknown position).

Basic variables classes

Measurements $y_{1:T} = (y_1, y_2, \ldots, y_T)$: The measured data somehow obtained from the phenomenon we are interested in.

Unknown (static) model parameters θ: Describes the model, but unknown (or not known well enough) to the user.

Unknown model variables x_t (changing over time): Describes the state of the phenomenon at time t (in the indoor positioning example above x_t includes the unknown position).

Explanatory variables u: Known variables that we do not bother to model as stochastic.

Basic variables classes

Measurements $y_{1:T} = (y_1, y_2, \ldots, y_T)$: The measured data somehow obtained from the phenomenon we are interested in.

Unknown (static) model parameters θ: Describes the model, but unknown (or not known well enough) to the user.

Unknown model variables x_t (changing over time): Describes the state of the phenomenon at time t (in the indoor positioning example above x_t includes the unknown position).

Explanatory variables u: Known variables that we do not bother to model as stochastic.

A key task is often to learn θ and/or x_t based on the available measurements $y_{1:T}$.

Key probabilistic objects

For models involving measurements Y and unknown static parameter θ

Key probabilistic objects

For models involving measurements Y and unknown static parameter θ

Full probabilistic model: The joint model of the data and parameters.

$$p(\theta, y) = p(y \mid \theta)p(\theta)$$

Key probabilistic objects

For models involving measurements Y and unknown static parameter θ

Full probabilistic model: The joint model of the data and parameters.

$$p(\theta, y) = p(y \mid \theta)p(\theta)$$

Likelihood function: $\mathcal{L}(\theta; y) = p(Y = y \mid \theta)$

Key probabilistic objects

For models involving measurements Y and unknown static parameter θ

Full probabilistic model: The joint model of the data and parameters.

$$p(\theta, y) = p(y \mid \theta)p(\theta)$$

Likelihood function: $\mathcal{L}(\theta; y) = p(Y = y \mid \theta)$

Posterior distribution: Condition on y instead of θ.

$$p(\theta \mid y) = \frac{p(\theta, y)}{p(y)} = \frac{p(y \mid \theta)}{\int p(y \mid \theta)p(\theta)\mathrm{d}\theta}$$

Key probabilistic objects

For models involving measurements Y and unknown static parameter θ

Full probabilistic model: The joint model of the data and parameters.

$$p(\theta, y) = p(y \mid \theta)p(\theta)$$

Likelihood function: $\mathcal{L}(\theta; y) = p(Y = y \mid \theta)$

Posterior distribution: Condition on y instead of θ.

$$p(\theta \mid y) = \frac{p(\theta, y)}{p(y)} = \frac{p(y \mid \theta)}{\int p(y \mid \theta)p(\theta)\mathrm{d}\theta}$$

Prediction: A statement about a future event $\bar{y}$ that has not been observed.

$$p(\bar{y} \mid y) = \int p(\bar{y}, \theta \mid y)\mathrm{d}\theta = \int p(\bar{y} \mid \theta)p(\theta \mid y)\mathrm{d}\theta$$

Computational problems

The problem of learning a model based on data leads to computational challenges, both

- Integration: e.g. the high-dimensional integrals arising during marg. (averaging over all possible parameter values θ):

$$p(y_{1:T}) = \int p(y_{1:T} \mid \theta) p(\theta) \mathrm{d}\theta.$$

- Optimization: e.g. when extracting point estimates, for example by maximizing the posterior or the likelihood

$$\widehat{\theta} = \arg\max_{\theta} \; p(y_{1:T} \mid \theta)$$

Computational problems

The problem of learning a model based on data leads to computational challenges, both

- Integration: e.g. the high-dimensional integrals arising during marg. (averaging over all possible parameter values θ):

$$p(y_{1:T}) = \int p(y_{1:T} \mid \theta) p(\theta) \mathrm{d}\theta.$$

- Optimization: e.g. when extracting point estimates, for example by maximizing the posterior or the likelihood

$$\widehat{\theta} = \arg\max_{\theta} \; p(y_{1:T} \mid \theta)$$

Typically impossible to compute exactly, use approximate methods

- Monte Carlo (MC), Markov chain MC (MCMC), and sequential MC (SMC).
- Variational inference (VI).

Probabilistic autoregressive model

Probabilistic autoregressive model

An autoregressive model of order n is given by

$$Y_t = A_1 Y_{t-1} + A_2 Y_{t-2} + \cdots + A_n Y_{t-n} + E_t, \quad E_t \sim \mathcal{N}(\mu, \tau^{-1})$$

where μ and τ are known explanatory variables ($\mu = 0, \tau \neq 0$).

Probabilistic autoregressive model

An autoregressive model of order n is given by

$$Y_t = A_1 Y_{t-1} + A_2 Y_{t-2} + \cdots + A_n Y_{t-n} + E_t, \quad E_t \sim \mathcal{N}(\mu, \tau^{-1})$$

where μ and τ are known explanatory variables ($\mu = 0, \tau \neq 0$).

The unknown model variables are collected as

$$\theta = (A_1, A_2, \ldots, A_n)^\mathsf{T}$$

with the prior

$$\theta \sim \mathcal{N}(0, \rho^{-1} I_n), \qquad \text{where } \rho \text{ assumed to be known.}$$

Probabilistic autoregressive model

An autoregressive model of order n is given by

$$Y_t = A_1 Y_{t-1} + A_2 Y_{t-2} + \cdots + A_n Y_{t-n} + E_t, \quad E_t \sim \mathcal{N}(\mu, \tau^{-1})$$

where μ and τ are known explanatory variables ($\mu = 0, \tau \neq 0$).

The unknown model variables are collected as

$$\theta = (A_1, A_2, \ldots, A_n)^\mathsf{T}$$

with the prior

$$\theta \sim \mathcal{N}(0, \rho^{-1} I_n), \qquad \text{where } \rho \text{ assumed to be known.}$$

Task: Compute the posterior $p(\theta \mid y_{1:T})$.

Probabilistic autoregressive model

Full probabilistic model $p(\theta, y_{1:T}) = p(y_{1:T} \mid \theta)p(\theta)$, where the data distribution is given by

$$p(y_{1:T} \mid \theta) = p(y_T \mid y_{1:T-1}, \theta)p(y_{1:T-1} \mid \theta) = \cdots = \prod_{t=1}^{T} p(y_t \mid y_{1:t-1}, \theta).$$

Probabilistic autoregressive model

Full probabilistic model $p(\theta, y_{1:T}) = p(y_{1:T} \mid \theta)p(\theta)$, where the data distribution is given by

$$p(y_{1:T} \mid \theta) = p(y_T \mid y_{1:T-1}, \theta)p(y_{1:T-1} \mid \theta) = \cdots = \prod_{t=1}^{T} p(y_t \mid y_{1:t-1}, \theta).$$

From the model we have that

$$p(y_t \mid y_{1:t-1}, \theta) = \mathcal{N}(y_t \mid \theta^{\mathsf{T}} z_t, \tau^{-1}),$$

where $Z_t = (Y_{t-1}, Y_{t-2}, \ldots, Y_{t-n})^{\mathsf{T}}$.

Probabilistic autoregressive model

Full probabilistic model $p(\theta, y_{1:T}) = p(y_{1:T} \mid \theta)p(\theta)$, where the data distribution is given by

$$p(y_{1:T} \mid \theta) = p(y_T \mid y_{1:T-1}, \theta)p(y_{1:T-1} \mid \theta) = \cdots = \prod_{t=1}^{T} p(y_t \mid y_{1:t-1}, \theta).$$

From the model we have that

$$p(y_t \mid y_{1:t-1}, \theta) = \mathcal{N}(y_t \mid \theta^{\mathsf{T}} z_t, \tau^{-1}),$$

where $Z_t = (Y_{t-1}, Y_{t-2}, \ldots, Y_{t-n})^{\mathsf{T}}$. Hence,

$$p(y_{1:T} \mid \theta) = \prod_{t=1}^{T} \mathcal{N}\left(y_t \mid \theta^{\mathsf{T}} z_t, \tau^{-1}\right) = \mathcal{N}\left(\boldsymbol{y} \mid \boldsymbol{z}\theta, \tau^{-1} I_T\right),$$

where we have made use of $\boldsymbol{Y} = (Y_1, Y_2, \ldots, Y_T)^{\mathsf{T}}$ and $\boldsymbol{Z} = (Z_1, Z_2, \ldots, Z_T)^{\mathsf{T}}$.

Probabilistic autoregressive model

$$
\begin{aligned}
p(\theta, y) &= \underbrace{\mathcal{N}\left(y \,\middle|\, z\theta, \tau^{-1} I_T\right)}_{p(y \mid \theta)} \underbrace{\mathcal{N}\left(\theta \,\middle|\, 0, \rho^{-1} I_n\right)}_{p(\theta)} \\
&= \mathcal{N}\left(\begin{pmatrix} \theta \\ y \end{pmatrix} \middle| \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \rho^{-1} I_2 & \rho^{-1} z^{\mathsf{T}} \\ \rho^{-1} z & \tau^{-1} I_T + \rho^{-1} z z^{\mathsf{T}} \end{pmatrix} \right).
\end{aligned}
$$

Probabilistic autoregressive model

$$
\begin{aligned}
p(\theta, y) &= \underbrace{\mathcal{N}\left(y \,\middle|\, z\theta, \tau^{-1} I_T\right)}_{p(y \mid \theta)} \underbrace{\mathcal{N}\left(\theta \,\middle|\, 0, \rho^{-1} I_n\right)}_{p(\theta)} \\
&= \mathcal{N}\left(\begin{pmatrix} \theta \\ y \end{pmatrix} \middle| \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \rho^{-1} I_2 & \rho^{-1} z^{\mathsf{T}} \\ \rho^{-1} z & \tau^{-1} I_T + \rho^{-1} z z^{\mathsf{T}} \end{pmatrix} \right).
\end{aligned}
$$

The posterior is given by

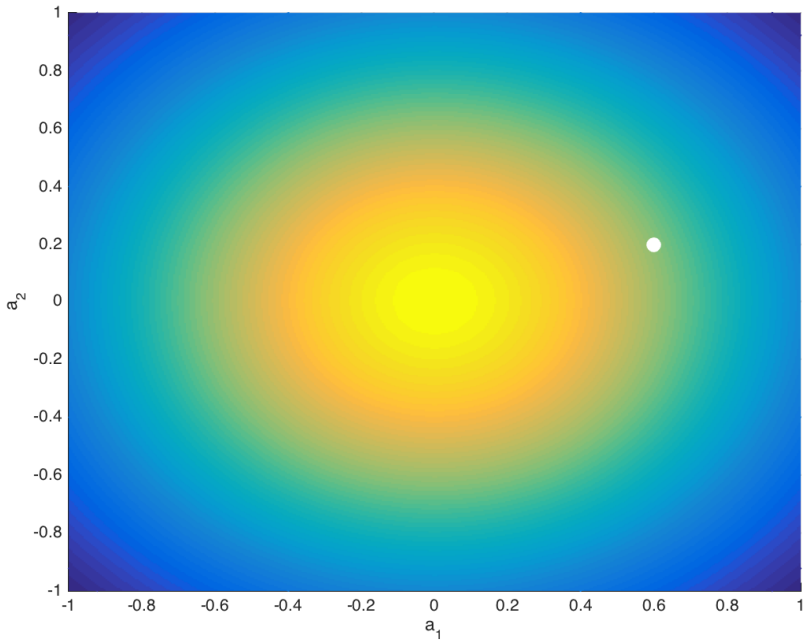
$$
p(\theta \,|\, y) = \mathcal{N}(\theta \,|\, m_T, S_T),
$$

where

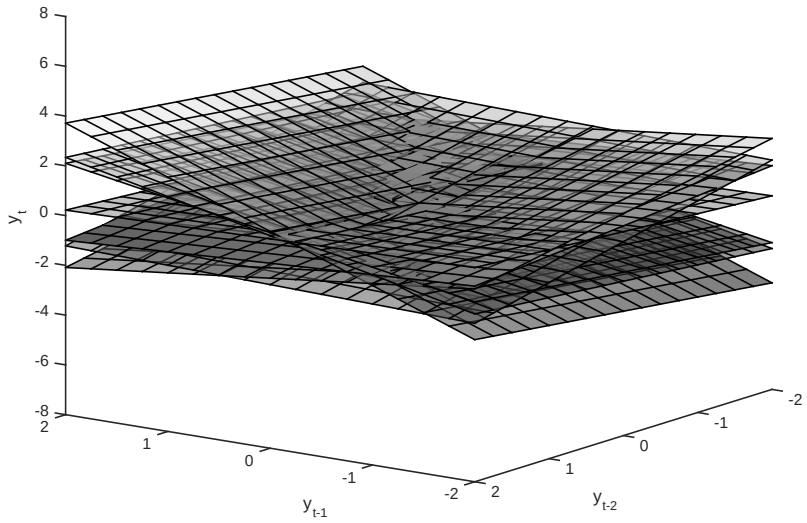
$$
\begin{aligned}
m_T &= \tau S_T z^{\mathsf{T}} y, \\
S_T &= \left(\rho^{-1} I_2 + \sigma z^{\mathsf{T}} z\right)^{\mathsf{T}}.
\end{aligned}
$$

Ex) Situation before any data is used

$$Y_t = A_1 Y_{t-1} + A_2 Y_{t-2} + E_t, \qquad E_t \sim \mathcal{N}(0, 0.2).$$



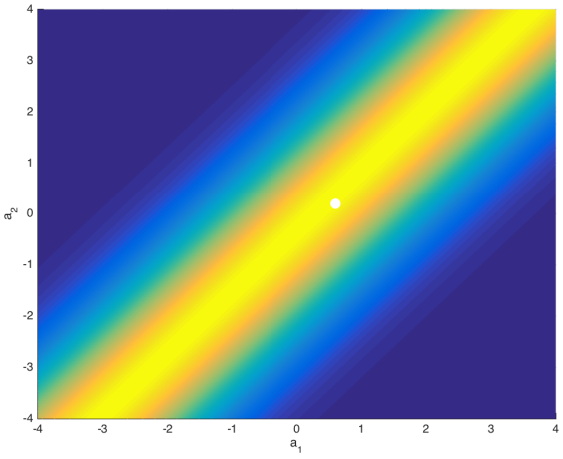
Prior



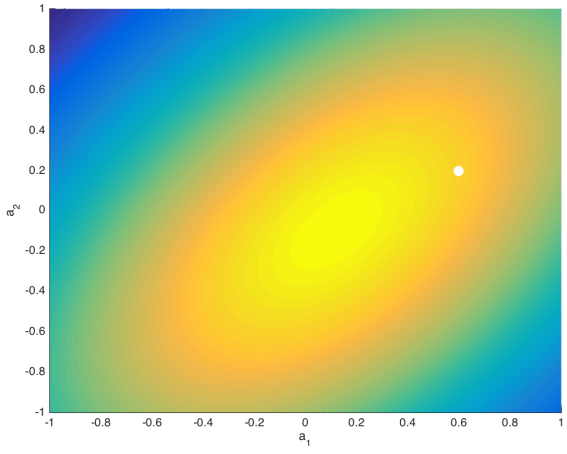
7 samples from the prior

White dot – true value for $\theta = (0.6, 0.2)$.

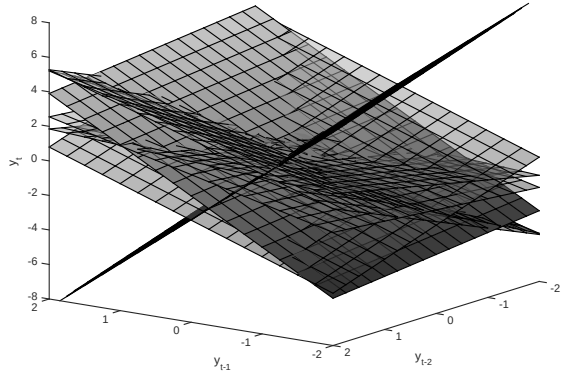
Ex) Situation after y_1 is obtained



Likelihood

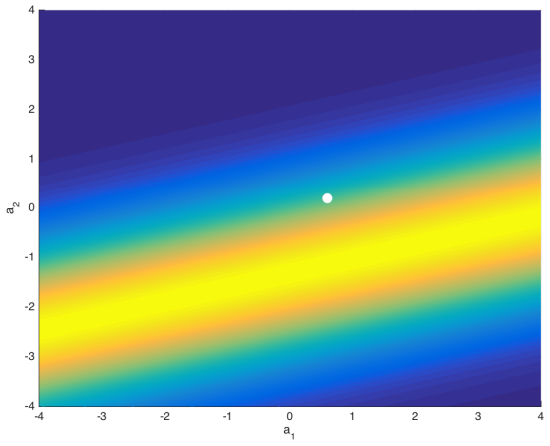


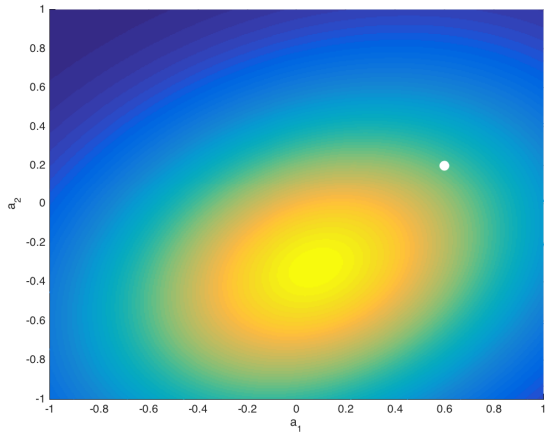
Posterior

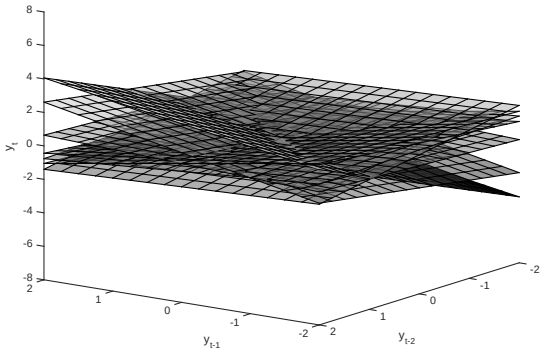


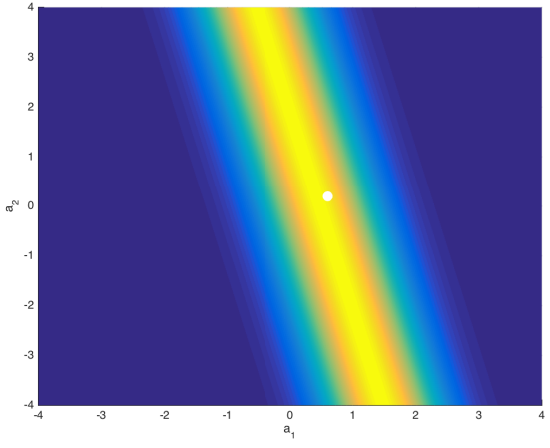
7 samples from the posterior

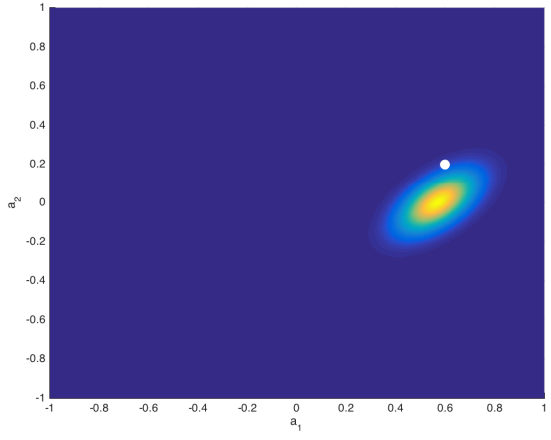
Ex) Situation after $y_{1:2}$ and $y_{1:20}$

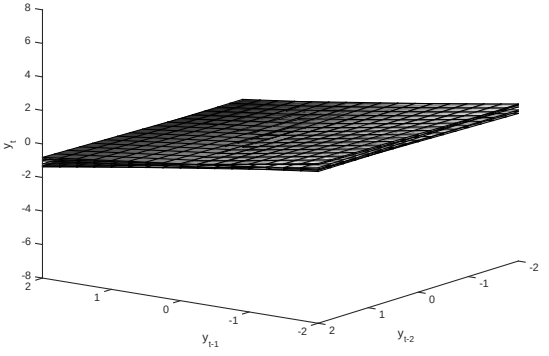












Likelihood

Posterior

7 samples from the posterior

A few concepts to summarize lecture 1

Mathematical model: A compact representation—set of assumptions—of some phenomenon of interest.

Probabilistic modelling: Provides the capability to represent and manipulate **uncertainty** in data, models, decisions and predictions.

Full probabilistic model: The joint distribution of all observed (here $y_{1:T}$) and unobserved (here θ) variables.

Data distribution/likelihood: Distribution describing the observed data conditioned on unobserved variables.

Prior distribution: Encodes initial assumptions on the unobserved variables.

Posterior distribution: Conditional distribution of the unobserved variables given the observed variables.