2018 Intelligent Systems (**知的システム構成論**)

EM Algorithm for Learning Linear Dynamical Systems

May.10, 2018

Takehisa YAIRI（**矢入健久**）
E-mail: yairi@ailab.t.u-tokyo.ac.jp

Today's Goal

- Understand the maximum likelihood approach to learning dynamical systems
- Derive the EM (Expectation-Maximization) algorithm for linear dynamical systems
- Need to overcome many hurdles to reach there
  - Bayesian inference for dynamical systems
  - Properties of linear Gaussian systems
  - Kalman filtering and RTS smoothing
  - Jensen's inequality
  - Minorization-maximization (MM) algorithm
- It's a long journey…

Probabilistic Inference for Dynamical Systems

- Introduction of Bayesian Filtering -

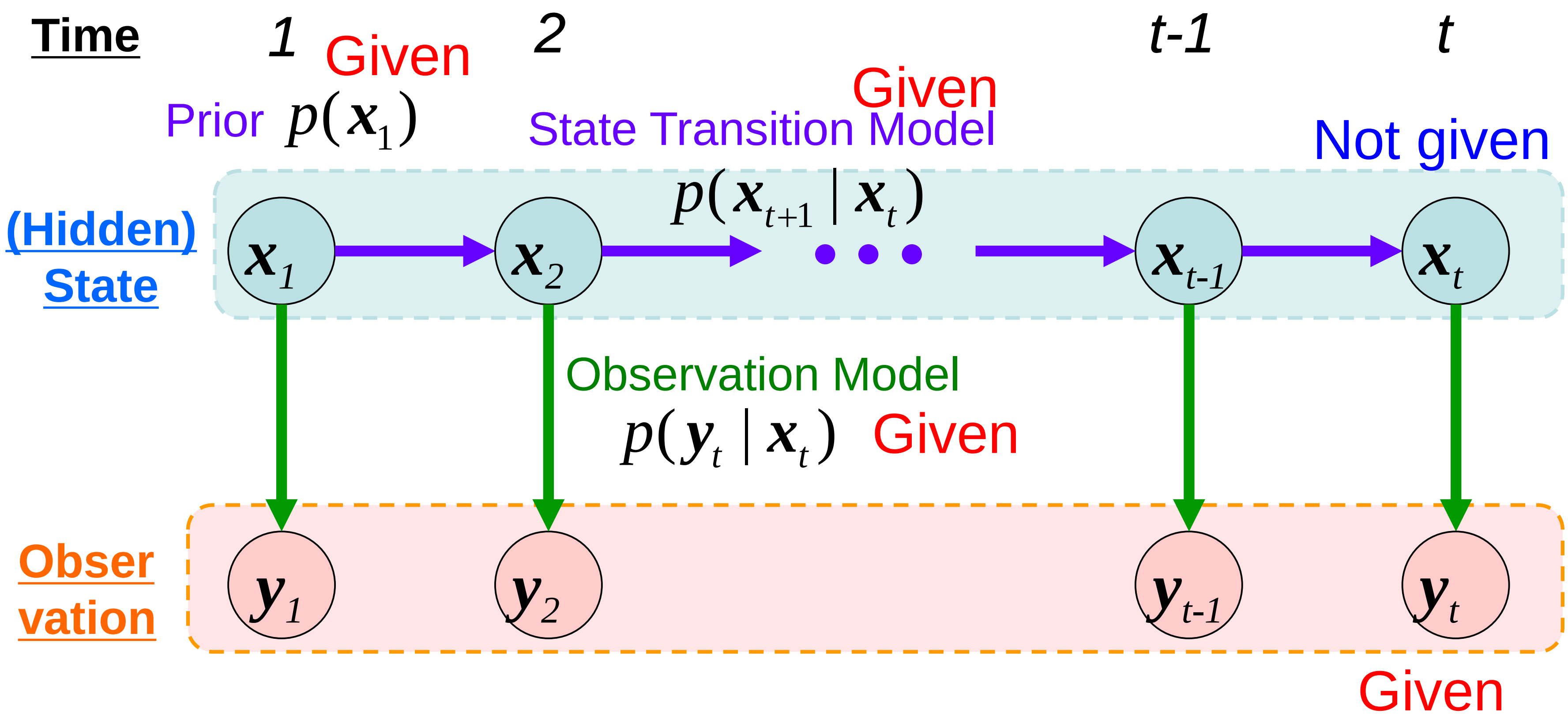
Bayesian Filtering
(Problem Definition)

- Purpose :
  - Estimate the current state $\boldsymbol{x}_t$ optimally using all observations up to now $\boldsymbol{y}_{1:t}$
- Given :
  - State transition model : $p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})$
  - Observation model : $p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)$
  - Prior distribution : $p(\boldsymbol{x}_1)$
  - Observations up to t : $\boldsymbol{y}_{1:t} \equiv \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_{t-1}, \boldsymbol{y}_t\}$

For simplicity, control input $\boldsymbol{u}_t$ is ignored

- Find :
  - Posterior distribution at time t : $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})$

Illustration



State Estimation:

When (observation) and (State model) and (Observation model) are given, estimate (Hidden state)

Bayesian Filtering: Derivation (1)

All observation up to time t

Obs. at time t

Obs. up to t-1

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) = p(\boldsymbol{x}_t \mid \boldsymbol{y}_t, \boldsymbol{y}_{t-1}, \cdots, \boldsymbol{y}_2, \boldsymbol{y}_1) = p(\boldsymbol{x}_t \mid \boldsymbol{y}_t, \boldsymbol{y}_{1:t-1})$$

$$= \frac{p(\boldsymbol{y}_t \mid \boldsymbol{x}_t, \boldsymbol{y}_{1:t-1}) \cdot p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1})}{p(\boldsymbol{y}_t \mid \boldsymbol{y}_{1:t-1})}$$

Bayes' theorem

$$P(X \mid Y) = \frac{P(Y \mid X) \cdot P(X)}{P(Y)}$$

Independent of $\boldsymbol{x}_t$

$$= \alpha \cdot \boxed{p(\boldsymbol{y}_t \mid \boldsymbol{x}_t, \boldsymbol{y}_{1:t-1})} \cdot \boxed{p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1})}$$

constant

Predictive distribution

Bayesian Filtering: Derivation (2)

$$\boxed{p(\mathbf{y}_t \mid \mathbf{x}_t, \cancel{\mathbf{y}_{1:t-1}})} = \underbrace{p(\mathbf{y}_t \mid \mathbf{x}_t)}_{\text{Observation model}}$$

Markov property (Conditional independence)

$$\boxed{p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1})} = \int p(\boldsymbol{x}_t, \boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1})\, d\boldsymbol{x}_{t-1}$$

Marginalization

$$= \int \left[p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1}, \cancel{\boldsymbol{y}_{1:t-1}}) \cdot p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1}) \right] d\boldsymbol{x}_{t-1}$$

Bayes' Rule

$$\approx \int \left[\underbrace{p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})}_{\text{State Transition model}} \cdot \underbrace{p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1})}_{\text{Posterior at previous time step}} \right] d\boldsymbol{x}_{t-1}$$

Markov property

Bayesian Filtering: Derivation (3)

After all,

$$\underbrace{p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})}_{\text{Posterior (Belief) at time t}} \approx \alpha \cdot \underbrace{p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)}_{\text{Observation Model}} \int \Big[\underbrace{p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})}_{\text{Transition Model}} \cdot \underbrace{p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1})}_{\text{Posterior (Belief) at time t-1}} \Big] d\boldsymbol{x}_{t-1}$$

- Posterior at every time step can be computed recursively
  - On-line update
  - No need to store past measurements
- What should we do with α (normalizing factor) ?
  - Automatically determined so that posterior integrated over X_t become 1

Normalizing Factor

The normalizing factor α can be computed explicitly, though we ignored it as it is independent of $\boldsymbol{x}_t$

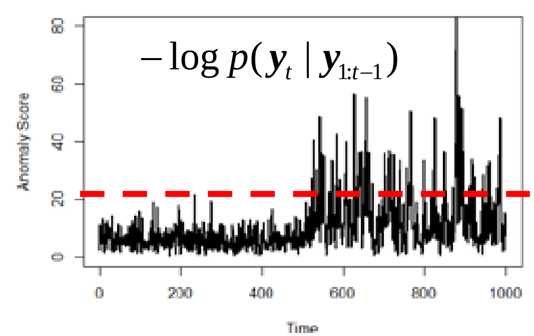
$$\alpha^{-1} = \boxed{p(\boldsymbol{y}_t \mid \boldsymbol{y}_{1:t-1})} = \int p(\boldsymbol{y}_t, \boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1})\, d\boldsymbol{x}_t$$

$$= \int p(\boldsymbol{y}_t \mid \boldsymbol{x}_t, \cancel{\boldsymbol{y}_{1:t-1}}) \cdot p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t-1})\, d\boldsymbol{x}_t$$

$$= \int p(\boldsymbol{y}_t \mid \boldsymbol{x}_t) \cdot \left[\int p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})\, p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{1:t-1}) d\boldsymbol{x}_{t-1}\right] d\boldsymbol{x}_t$$

Predictive distribution for the next observation

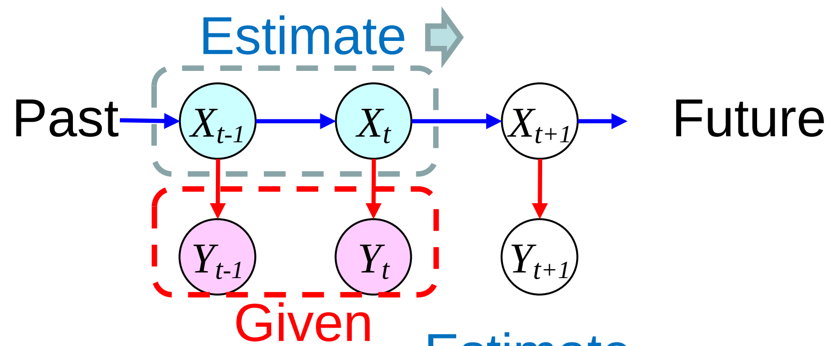
- Useful for anomaly detection !



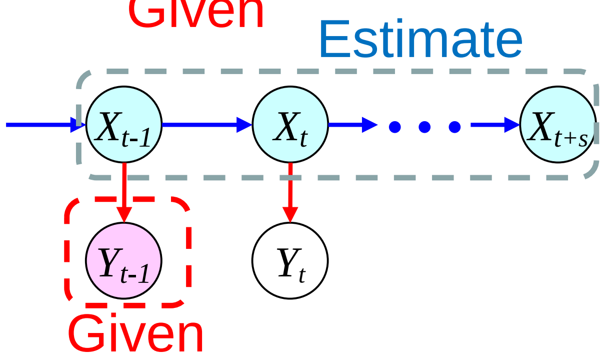
Prediction and Smoothing

- "Prediction" and "smoothing" are similar to but different from "filtering"

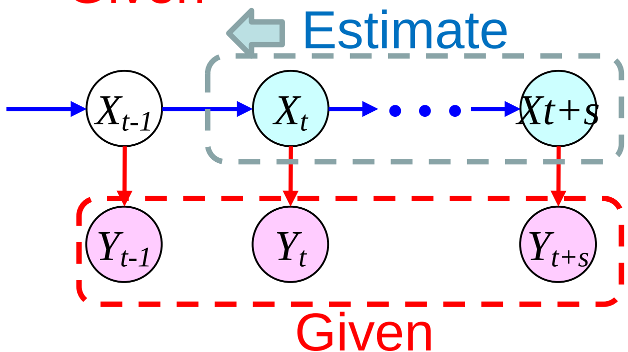
Filtering : $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})$



Prediction : $p(\boldsymbol{x}_{t+s} \mid \boldsymbol{y}_{1:t})$

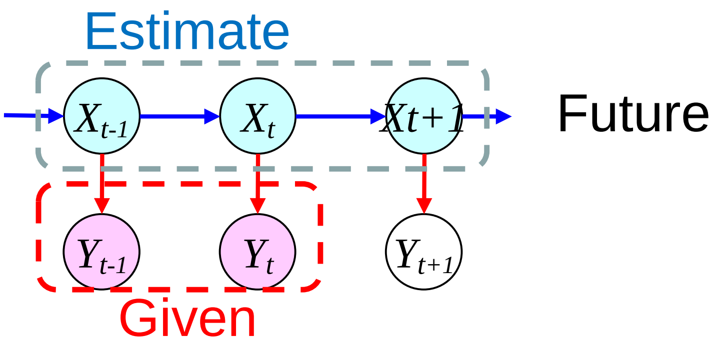


Smoothing : $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t+s})$



Bayesian Prediction (1)

One-step Prediction



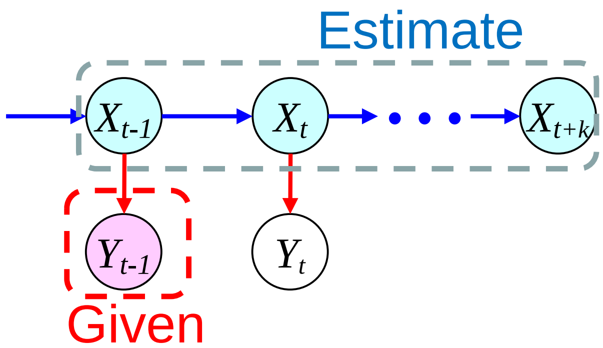
$$p(\mathbf{x}_{t+1} \mid \mathbf{y}_{1:t}) = \int p(\mathbf{x}_{t+1}, \mathbf{x}_t \mid \mathbf{y}_{1:t}) \, d\mathbf{x}_t$$ Marginalization

$$= \int p(\mathbf{x}_{t+1} \mid \mathbf{x}_t, \cancel{\mathbf{y}_{1:t}}) \, p(\mathbf{x}_t \mid \mathbf{y}_{1:t}) \, d\mathbf{x}_t$$ Bayes rule

$$= \int \underbrace{p(\mathbf{x}_{t+1} \mid \mathbf{x}_t)}_{\text{State Transition model}} \, \underbrace{p(\mathbf{x}_t \mid \mathbf{y}_{1:t})}_{\text{Filtering Distribution at time t}} \, d\mathbf{x}_t$$ Markov property

Bayesian Prediction (2)

k-step Prediction



$$p(\boldsymbol{x}_{t+k} \mid \boldsymbol{y}_{1:t}) = \int p(\boldsymbol{x}_{t+k}, \boldsymbol{x}_{t+k-1} \mid \boldsymbol{y}_{1:t})\, d\boldsymbol{x}_{t+k-1}$$

$$= \iint p(\boldsymbol{x}_{t+k}, \boldsymbol{x}_{t+k-1}, \boldsymbol{x}_{t+k-2} \mid \boldsymbol{y}_{1:t})\, d\boldsymbol{x}_{t+k-1} d\boldsymbol{x}_{t+k-2}$$

$$= \iint \cdots \int p(\boldsymbol{x}_{t+k}, \boldsymbol{x}_{t+k-1}, \cdots, \boldsymbol{x}_{t} \mid \boldsymbol{y}_{1:t})\, d\boldsymbol{x}_{t+k-1} \cdots d\boldsymbol{x}_{t}$$

Multiple integral

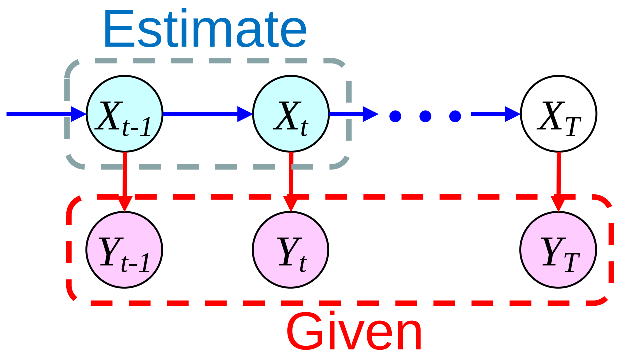
$$= \int \underbrace{p(\boldsymbol{x}_{t+k} \mid \boldsymbol{x}_{t+k-1})}_{\text{State Transition model}}\ \underbrace{p(\boldsymbol{x}_{t+k-1} \mid \boldsymbol{y}_{1:t})}_{\text{(k-1)-step prediction}}\, d\boldsymbol{x}_{t+k-1}$$

Recursive form

Bayesian Smoothing (1)

- Derive the equation for computing the posterior of state variable in smoothing where t < T

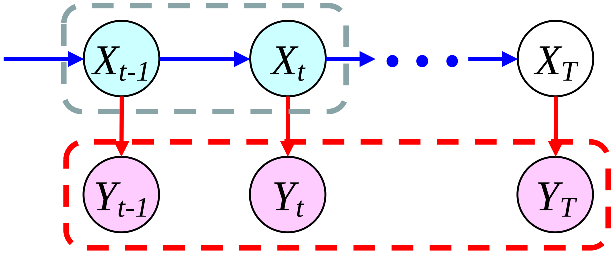
T : terminal time step



- Consider a backward recursive form !
  - i.e., Represent $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:T})$ in terms of $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})$

Bayesian Smoothing (2)

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:T}) = \int p(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})\, d\boldsymbol{x}_{t+1}$$



marginal of joint distribution with $\mathbf{x}_{t+1}$

$$p(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}) = p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:T}) \cdot p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})$$ Bayes rule

$$= p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:t}) \cdot p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})$$ Markov property

$$= \frac{p(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t})}{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t})} \cdot p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})$$ ⇦ P(A|B) = P(A,B)/P(B)

$$= \frac{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \boldsymbol{y}_{1:t}) \cdot p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) \cdot p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})}{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t})}$$

Bayesian Smoothing (3)

$$p(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}) = \frac{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \cancel{\boldsymbol{y}_{1:t}}) \cdot p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) \cdot p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})}{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t})}$$

Joint

$$= \frac{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t) \cdot p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) \cdot p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})}{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t})}$$

Markov property

Finally,

Smoothing at time t

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:T}) = \int \frac{\overbrace{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t)}^{\text{Transition model}} \cdot \overbrace{p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})}^{\text{Filtering at time t}} \cdot \overbrace{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})}^{\text{Smoothing at t+1}}}{\underbrace{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t})}_{\text{One-step prediction at t}}} d\boldsymbol{x}_{t+1}$$

Properties of Linear Gaussian Systems

Multivariate Gaussian Distribution (1)

Consider that vectors $\boldsymbol{x}$, $\boldsymbol{y}$ are multivariate Gaussian

$$\boldsymbol{x} \sim \mathrm{N}\left(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x\right) \qquad \boldsymbol{y} \sim \mathrm{N}\left(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y\right)$$

Covariance matrices

$$\Sigma_x = V(\boldsymbol{x}) \equiv E\left[(\boldsymbol{x} - \boldsymbol{\mu}_x)(\boldsymbol{x} - \boldsymbol{\mu}_x)^T\right]$$

Cross-Covariance matrix between $\boldsymbol{x}$ and $\boldsymbol{y}$

$$\Sigma_{xy} = Cov(\boldsymbol{x}, \boldsymbol{y}) \equiv E\left[(\boldsymbol{x} - \boldsymbol{\mu}_x)(\boldsymbol{y} - \boldsymbol{\mu}_y)^T\right]$$

Joint distribution is also a multivariate Gaussian

$$\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} \sim \mathrm{N}\left(\begin{bmatrix} \boldsymbol{\mu}_x \\ \boldsymbol{\mu}_y \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_x & \boldsymbol{\Sigma}_{xy} \\ \boldsymbol{\Sigma}_{xy}^T & \boldsymbol{\Sigma}_y \end{bmatrix}\right)$$

(Joint Gaussian)

Multivariate Gaussian Distribution (2)

- Linear combination

$\boldsymbol{x}$ and $\boldsymbol{y}$ are independent

Assume $\quad \boldsymbol{x} \sim N(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) \quad \boldsymbol{y} \sim N(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)$ $\quad \boxed{Cov(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{0}}$

Then $\quad \boldsymbol{Ax} + \boldsymbol{By} \sim N(\boldsymbol{A\mu}_x + \boldsymbol{B\mu}_y, \boldsymbol{A\Sigma}_x\boldsymbol{A}^T + \boldsymbol{B\Sigma}_y\boldsymbol{B}^T)$

- If $\boldsymbol{x}$ and $\boldsymbol{y}$ are joint Gaussian (and mutually dependent), then conditional probability distribution of $\boldsymbol{y}$ given $\boldsymbol{x}$ (or posterior of $\boldsymbol{y}$) is obtained as:

$$\boxed{p(\boldsymbol{y} \mid \boldsymbol{x}) = \mathrm{N}\left(\boldsymbol{y} \mid \boldsymbol{\mu}_y + \boldsymbol{\Sigma}_{xy}{}^T \boldsymbol{\Sigma}_x{}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_x),\ \boldsymbol{\Sigma}_y - \boldsymbol{\Sigma}_{xy}{}^T \boldsymbol{\Sigma}_x{}^{-1} \boldsymbol{\Sigma}_{xy}\right)}$$

cf. Prior : $P(\boldsymbol{y}) = \mathrm{N}(\boldsymbol{y} \mid \boldsymbol{\mu}_y,\ \boldsymbol{\Sigma}_y)$

Essence of Kalman filter

Important Results for Linear Gaussian Systems (1)

Assume:

$\boldsymbol{x}$ is a Gaussian : $\boldsymbol{x} \sim \mathrm{N}(\boldsymbol{m}, \boldsymbol{P}) \quad \Leftrightarrow \quad p(\boldsymbol{x}) = \mathrm{N}(\boldsymbol{x} \mid \boldsymbol{m}, \boldsymbol{P})$

$\boldsymbol{y}$ is a linear transform of $\boldsymbol{x}$ plus Gaussian noise $\boldsymbol{v}$:

$$\boldsymbol{v} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{R}) \quad \Leftrightarrow \quad p(\boldsymbol{v}) = \mathrm{N}(\boldsymbol{v} \mid \boldsymbol{0}, \boldsymbol{R})$$

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b} + \boldsymbol{v} \quad \Leftrightarrow \quad p(\boldsymbol{y} \mid \boldsymbol{x}) = \mathrm{N}(\boldsymbol{y} \mid \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}, \boldsymbol{R})$$

Joint distribution:

$$p(\boldsymbol{x}, \boldsymbol{y}) = p(\boldsymbol{y} \mid \boldsymbol{x}) p(\boldsymbol{x}) = \mathrm{N}(\boldsymbol{y} \mid \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}, \boldsymbol{R}) \cdot \mathrm{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \boldsymbol{P})$$

$$= \mathrm{N}\left(\begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{bmatrix} \middle| \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{b} \end{bmatrix}, \begin{bmatrix} \boldsymbol{P} & \boldsymbol{P}\boldsymbol{A}^T \\ \boldsymbol{A}\boldsymbol{P} & \boldsymbol{A}\boldsymbol{P}\boldsymbol{A}^T + \boldsymbol{R} \end{bmatrix} \right)$$

Marginal distribution:

$$p(\boldsymbol{y}) = \int p(\boldsymbol{x}, \boldsymbol{y}) d\boldsymbol{x} = \mathrm{N}\left(\boldsymbol{y} \middle| \boldsymbol{A}\boldsymbol{\mu} + \boldsymbol{b}, \boldsymbol{A}\boldsymbol{P}\boldsymbol{A}^T + \boldsymbol{R}\right)$$

Important Results for Linear Gaussian Systems (2)

Assume joint distribution of $\boldsymbol{x}$ and $\boldsymbol{y}$ is Gaussian:

$$p(\boldsymbol{x},\boldsymbol{y}) = \mathrm{N}\left(\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix}\middle|\begin{bmatrix}\boldsymbol{a}\\ \boldsymbol{b}\end{bmatrix},\begin{bmatrix}\boldsymbol{P} & \boldsymbol{R}\\ \boldsymbol{R}^T & \boldsymbol{Q}\end{bmatrix}\right)$$

Marginal : $p(\boldsymbol{x}) = \mathrm{N}(\boldsymbol{x}\,|\boldsymbol{a},\boldsymbol{P})$ $\quad p(\boldsymbol{y}) = \mathrm{N}(\boldsymbol{y}\,|\boldsymbol{b},\boldsymbol{Q})$

Conditional distribution:

$$p(\boldsymbol{x}\mid\boldsymbol{y}) = \mathrm{N}\left(\boldsymbol{x}\,\middle|\boldsymbol{a}+\boldsymbol{R}\boldsymbol{Q}^{-1}(\boldsymbol{y}-\boldsymbol{b}),\boldsymbol{P}-\boldsymbol{R}\boldsymbol{Q}^{-1}\boldsymbol{R}^T\right)$$

$$p(\boldsymbol{y}\mid\boldsymbol{x}) = \mathrm{N}\left(\boldsymbol{y}\,\middle|\boldsymbol{b}+\boldsymbol{R}^T\boldsymbol{P}^{-1}(\boldsymbol{x}-\boldsymbol{a}),\boldsymbol{Q}-\boldsymbol{R}^T\boldsymbol{P}^{-1}\boldsymbol{R}\right)$$

Important Results for Linear Gaussian Systems (3)

With these two results, we can compute

Prior dist.		Joint dist.		Posterior	Likelihood
$p(\boldsymbol{x})$	$\Rightarrow$	$p(\boldsymbol{x}, \boldsymbol{y})$	$\Rightarrow$	$p(\boldsymbol{x} \mid \boldsymbol{y})$	$p(\boldsymbol{y})$

for linear Gaussian systems.

Imagine that $\boldsymbol{x}$ is (hidden) state vector,
and $\boldsymbol{y}$ is measurement vector

Inference for Linear Dynamical Systems
- Kalman filtering and RTS smoothing -

Assumptions of Linear Gaussian

- Linear dynamical system
  (state space representation)

$$\begin{cases} \boldsymbol{x}_t = \boldsymbol{A}\boldsymbol{x}_{t-1} + \boldsymbol{w}_t & \text{(State equation)} \\ \boldsymbol{y}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{v}_t & \text{(Observation equation)} \end{cases}$$

- Noises are Gaussian (and independent)

$$\boldsymbol{w}_t \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{Q}) \qquad \boldsymbol{v}_t \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{R})$$

- Prior distribution of initial state is Gaussian

$$p(\boldsymbol{x}_1) = \mathrm{N}(\boldsymbol{x}_1 \mid \boldsymbol{m}_0, \boldsymbol{V}_0)$$

- Assume filter distribution $\mathrm{p}(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})$ is also Gaussian

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_t \mid \boldsymbol{m}_t, \boldsymbol{V}_t)$$

Derivation of Kalman Filter (1)

- Goal : Derive a recursive equation of filter dist.

$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_t \mid \boldsymbol{m}_t, \boldsymbol{V}_t)$ (Filter distribution at t)

$p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t+1}) = \mathrm{N}(\boldsymbol{x}_{t+1} \mid \boldsymbol{m}_{t+1}, \boldsymbol{V}_{t+1})$ (Filter distribution at $t+1$)

⇩ As filter distribution is Gaussian,

Derive $\boldsymbol{m}_{t+1}, \boldsymbol{V}_{t+1}$ from $\boldsymbol{m}_t, \boldsymbol{V}_t$

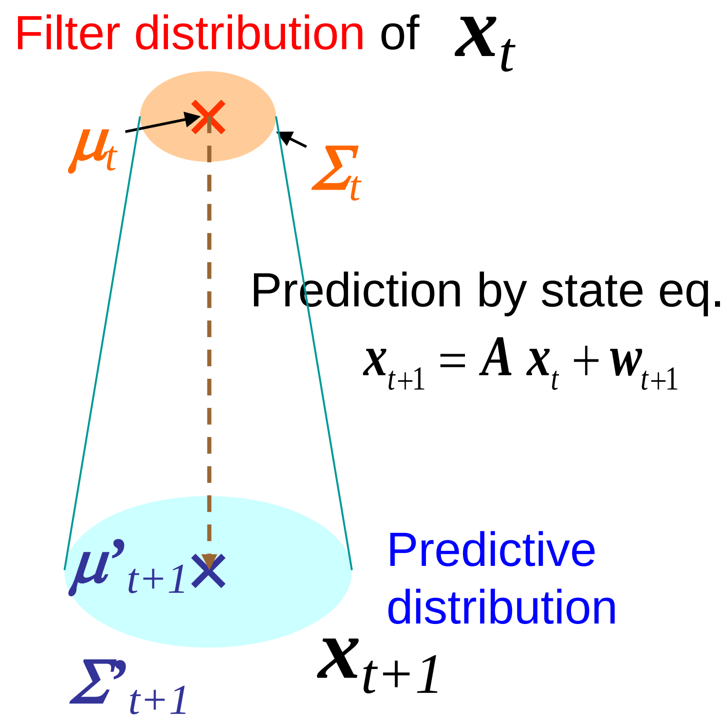
- Two steps :

1. Compute predictive distribution : $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t})$
2. Update filter distribution : $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t+1}) = p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t}, \boldsymbol{y}_{t+1})$

Derivation of Kalman Filter (2): Prediction

Start with filter distribution at time t

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_t \mid \boldsymbol{m}_t, \boldsymbol{V}_t)$$



State Equation

$$\begin{cases} \boldsymbol{x}_{t+1} = \boldsymbol{A}\,\boldsymbol{x}_t + \boldsymbol{w}_{t+1} \\ \boldsymbol{w}_{t+1} \sim N(\boldsymbol{0}, \boldsymbol{Q}) \end{cases}$$

Equivalent

$$p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t) = \mathrm{N}(\boldsymbol{x}_{t+1} \mid \boldsymbol{A}\boldsymbol{x}_t, \boldsymbol{Q})$$

One-step Prediction

$$p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t}) = \int p(\boldsymbol{x}_{t+1}, \boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})\, d\boldsymbol{x}_t$$

$$= \int p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t)\; p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t})\, d\boldsymbol{x}_t$$

How can we compute this ?

Derivation of Kalman Filter (2): Prediction

Remember the "important property of Gaussian Distribution" !

- $\boldsymbol{x}_{t+1}$ is a linear transform of $\boldsymbol{x}_t$ plus Gaussian noise

Joint distribution of $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t+1}$:

$$p(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t}) = p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t) p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_{t+1} \mid \boldsymbol{A}\boldsymbol{x}_t, \boldsymbol{Q}) \cdot \mathrm{N}(\boldsymbol{x}_t \mid \boldsymbol{\mu}_t, \boldsymbol{V}_t)$$

$$= \mathrm{N}\left(\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} \end{bmatrix} \middle| \begin{bmatrix} \boldsymbol{m}_t \\ \boldsymbol{A}\boldsymbol{m}_t \end{bmatrix}, \begin{bmatrix} \boldsymbol{V}_t & \boldsymbol{V}_t \boldsymbol{A}^T \\ \boldsymbol{A}\boldsymbol{V}_t & \boldsymbol{A}\boldsymbol{V}_t\boldsymbol{A}^T + \boldsymbol{Q} \end{bmatrix} \right)$$

Marginal distribution of $\boldsymbol{x}_{t+1}$: Predictive distribution !

$$p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t}) = \int p(\boldsymbol{x}_{t+1}, \boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) d\boldsymbol{x}_t = \mathrm{N}\left(\boldsymbol{x}_{t+1} \middle| \boldsymbol{A}\boldsymbol{m}_t, \underbrace{\boldsymbol{A}\boldsymbol{V}_t\boldsymbol{A}^T + \boldsymbol{Q}}_{\overset{\text{def.}}{=} \boldsymbol{P}_t}\right)$$

Derivation of Kalman Filter: Update

- Predictive dist.: $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_{t+1} \mid \boldsymbol{A}\boldsymbol{m}_t, \boldsymbol{P}_t)$
- Now that a new measurement $\mathbf{y}_{t+1}$ comes in, we want the filter dist. at t+1: $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t+1})$
- Consider the joint dist. of $\mathbf{x}_{t+1}$ and $\mathbf{y}_{t+1}$, given $\mathbf{y}_{1:t}$
- Observation equation:

$$\begin{cases} \boldsymbol{y}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{v}_t \\ \boldsymbol{v}_t \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{R}) \end{cases} \quad \Rightarrow \quad p(\boldsymbol{y}_{t+1} \mid \boldsymbol{x}_{t+1}) = \mathrm{N}(\boldsymbol{y}_{t+1} \mid \boldsymbol{C}\boldsymbol{x}_{t+1}, \boldsymbol{R})$$

equiv.

$$p(\boldsymbol{x}_{t+1}, \boldsymbol{y}_{t+1} \mid \boldsymbol{y}_{1:t}) = \mathrm{N}\left(\begin{bmatrix} \boldsymbol{x}_{t+1} \\ \boldsymbol{y}_{t+1} \end{bmatrix} \middle| \begin{bmatrix} \boldsymbol{A}\boldsymbol{m}_t \\ \boldsymbol{C}\boldsymbol{A}\boldsymbol{m}_t \end{bmatrix}, \begin{bmatrix} \boldsymbol{P}_t & \boldsymbol{P}_t\boldsymbol{C}^T \\ \boldsymbol{C}\boldsymbol{P}_t & \boldsymbol{C}\boldsymbol{P}_t\boldsymbol{C}^T + \boldsymbol{R} \end{bmatrix} \right)$$

where $\boldsymbol{P}_t = \boldsymbol{A}\boldsymbol{V}_t\boldsymbol{A}^T + \boldsymbol{Q}$

Derivation of Kalman Filter: Update

- Recall another important property of Gaussian !

$$p(\boldsymbol{x},\boldsymbol{y}) = \mathrm{N}\left(\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix}\middle|\begin{bmatrix}\boldsymbol{a}\\ \boldsymbol{b}\end{bmatrix},\begin{bmatrix}\boldsymbol{P} & \boldsymbol{R}\\ \boldsymbol{R}^T & \boldsymbol{Q}\end{bmatrix}\right)$$ Joint

$$p(\boldsymbol{x}\mid\boldsymbol{y}) = \mathrm{N}\left(\boldsymbol{x}\,\middle|\,\boldsymbol{a}+\boldsymbol{R}\boldsymbol{Q}^{-1}(\boldsymbol{y}-\boldsymbol{b}),\boldsymbol{P}-\boldsymbol{R}\boldsymbol{Q}^{-1}\boldsymbol{R}^T\right)$$ Conditional

$$p(\boldsymbol{x}_{t+1},\boldsymbol{y}_{t+1}\mid\boldsymbol{y}_{1:t}) = \mathrm{N}\left(\begin{bmatrix}\boldsymbol{x}_{t+1}\\ \boldsymbol{y}_{t+1}\end{bmatrix}\middle|\begin{bmatrix}\boldsymbol{A}\boldsymbol{m}_t\\ \boldsymbol{C}\boldsymbol{A}\boldsymbol{m}_t\end{bmatrix},\begin{bmatrix}\boldsymbol{P}_t & \boldsymbol{P}_t\boldsymbol{C}^T\\ \boldsymbol{C}\boldsymbol{P}_t & \boldsymbol{C}\boldsymbol{P}_t\boldsymbol{C}^T+\boldsymbol{R}\end{bmatrix}\right)$$

Confirm yourself !

$$\begin{aligned}p(\boldsymbol{x}_{t+1}\mid\boldsymbol{y}_{t+1},\boldsymbol{y}_{1:t}) &= p(\boldsymbol{x}_{t+1}\mid\boldsymbol{y}_{1:t+1})\\ &= N\left(\boldsymbol{x}_{t+1}\mid\boldsymbol{A}\boldsymbol{m}_t+\boldsymbol{K}_{t+1}(\boldsymbol{y}_{t+1}-\boldsymbol{C}\boldsymbol{A}\boldsymbol{m}_t),(\boldsymbol{I}-\boldsymbol{K}_{t+1}\boldsymbol{C})\boldsymbol{P}_t\right)\end{aligned}$$

Filter dist. at t+1

where $\boldsymbol{K}_{t+1} = \boldsymbol{P}_t\boldsymbol{C}^T\left(\boldsymbol{C}\boldsymbol{P}_t\boldsymbol{C}^T+\boldsymbol{R}\right)^{-1}$

Summary: Linear Kalman Filter

- Input : $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_t \mid \boldsymbol{m}_t, \boldsymbol{V}_t)$ and $\boldsymbol{y}_{\mathrm{t+1}}$
- Prediction: $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_{t+1} \mid \boldsymbol{A}\boldsymbol{m}_t, \boldsymbol{P}_t)$

  - where $\boldsymbol{P}_t = \boldsymbol{A}\boldsymbol{V}_t\boldsymbol{A}^T + \boldsymbol{Q}$

- Update :

$$p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t+1}) = N(\boldsymbol{x}_{t+1} \mid \boldsymbol{A}\boldsymbol{m}_t + \boldsymbol{K}_{t+1}(\boldsymbol{y}_{t+1} - \boldsymbol{C}\boldsymbol{A}\boldsymbol{m}_t), (\boldsymbol{I} - \boldsymbol{K}_{t+1}\boldsymbol{C})\boldsymbol{P}_t)$$

  - where $\boldsymbol{K}_{t+1} = \boldsymbol{P}_t\boldsymbol{C}^T(\boldsymbol{C}\boldsymbol{P}_t\boldsymbol{C}^T + \boldsymbol{R})^{-1}$

- Output : $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t+1}) = \mathrm{N}(\boldsymbol{x}_{t+1} \mid \boldsymbol{m}_{t+1}, \boldsymbol{V}_{t+1})$

  - where
$$\begin{cases} \boldsymbol{m}_{t+1} = \boldsymbol{A}\boldsymbol{m}_t + \boldsymbol{K}_{t+1}(\boldsymbol{y}_{t+1} - \boldsymbol{C}\boldsymbol{A}\boldsymbol{m}_t) \\ \boldsymbol{V}_{t+1} = (\boldsymbol{I} - \boldsymbol{K}_{t+1}\boldsymbol{C})\boldsymbol{P}_t \end{cases}$$

Derivation of RTS Smoother (1)

- Goal : Backward recursive equation of smoothed dist.

$$p\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}\right) = \mathrm{N}\left(\boldsymbol{x}_{t+1} \mid \hat{\boldsymbol{m}}_{t+1}, \hat{\boldsymbol{V}}_{t+1}\right)$$ (Smoothed dist. at $t+1$)

$$p\left(\boldsymbol{x}_{t} \mid \boldsymbol{y}_{1:T}\right) = \mathrm{N}\left(\boldsymbol{x}_{t} \mid \hat{\boldsymbol{m}}_{t}, \hat{\boldsymbol{V}}_{t}\right)$$ (Smoothed dist. at t)

Derive $\hat{\boldsymbol{m}}_{t}, \hat{\boldsymbol{V}}_{t}$ from $\hat{\boldsymbol{m}}_{t+1}, \hat{\boldsymbol{V}}_{t+1}$

Derivation of Rauch-Tung-Striebel (RTS) Smoother *

* RTS smoother is also called Kalman smoother

If the joint distribution of $\boldsymbol{x}_\mathrm{t}$ and $\boldsymbol{x}_{\mathrm{t+1}}$ given $\boldsymbol{y}_{1:\mathrm{T}}$ is obtained, we can get marginal distribution of $\boldsymbol{x}_\mathrm{t}$ using the property of Gaussian

If we know

$$p\left(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}\right) = \mathrm{N}\left(\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} \end{bmatrix} \middle| \begin{bmatrix} \hat{\boldsymbol{m}}_t \\ \hat{\boldsymbol{m}}_{t+1} \end{bmatrix}, \begin{bmatrix} \hat{\boldsymbol{V}}_t & \hat{\boldsymbol{V}}_{t,t+1} \\ \hat{\boldsymbol{V}}_{t+1,t} & \hat{\boldsymbol{V}}_{t+1} \end{bmatrix}\right)$$

Joint Gaussian

$$\Downarrow$$

$$p\left(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:T}\right) = \mathrm{N}\left(\boldsymbol{x}_t \mid \hat{\boldsymbol{m}}_t , \hat{\boldsymbol{V}}_t\right)$$

Marginal

So, try to obtain the joint distribution $p\left(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}\right)$

Derivation of RTS Smoother

$$p(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}) = p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:T}) \cdot p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}) \qquad \text{Bayes rule}$$

$$= \underbrace{p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:t})} \cdot \underbrace{p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})}_{\text{Smoothed dist. at t+1}} \qquad \text{Markov property}$$

By the way,

$$p(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t}) = \mathrm{N}\left(\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} \end{bmatrix} \middle| \begin{bmatrix} \boldsymbol{m}_t \\ \boldsymbol{A}\boldsymbol{m}_t \end{bmatrix}, \begin{bmatrix} \boldsymbol{V}_t & \boldsymbol{V}_t\boldsymbol{A}^T \\ \boldsymbol{A}\boldsymbol{V}_t & \boldsymbol{P}_t \end{bmatrix} \right)$$

So,

$$p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:t}) = \mathrm{N}\left(\boldsymbol{x}_t \,\middle|\, \boldsymbol{m}_t + \boldsymbol{J}_t(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{m}_t), \boldsymbol{V}_t - \boldsymbol{J}_t\boldsymbol{A}\boldsymbol{V}_t\right)$$

where $\boldsymbol{J}_t = \boldsymbol{V}_t\boldsymbol{A}^T\boldsymbol{P}_t^{-1}$

Derivation of RTS Smoother

Now, we have

$$p\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}\right) = \mathrm{N}\left(\boldsymbol{x}_{t+1} \mid \hat{\boldsymbol{m}}_{t+1}, \hat{\boldsymbol{V}}_{t+1}\right)$$

$$\begin{aligned} p\left(\boldsymbol{x}_{t} \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:T}\right) &= p\left(\boldsymbol{x}_{t} \mid \boldsymbol{x}_{t+1}, \boldsymbol{y}_{1:t}\right) \\ &= \mathrm{N}\Big(\underset{\boldsymbol{x}_t \text{ is}}{\boldsymbol{x}_t} \,\Big|\, \underbrace{\boldsymbol{m}_t + \boldsymbol{J}_t\left(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{m}_t\right)}_{\text{a linear transform of } \boldsymbol{x}_{t+1}}, \underbrace{\boldsymbol{V}_t - \boldsymbol{J}_t \boldsymbol{A} \boldsymbol{V}_t}_{\text{plus Gaussian noize}}\Big) \end{aligned}$$

We can obtain the joint distribution !

$$p\left(\boldsymbol{x}_{t}, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T}\right)$$

$$= \mathrm{N}\left(\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} \end{bmatrix} \middle| \begin{bmatrix} \boldsymbol{m}_t + \boldsymbol{J}_t\left(\hat{\boldsymbol{m}}_{t+1} - \boldsymbol{A}\boldsymbol{m}_t\right) \\ \hat{\boldsymbol{m}}_{t+1} \end{bmatrix}, \begin{bmatrix} \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1} \boldsymbol{J}_t^{T} + \boldsymbol{V}_t - \boldsymbol{J}_t \boldsymbol{A} \boldsymbol{V}_t & \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1} \\ \hat{\boldsymbol{V}}_{t+1} \boldsymbol{J}_t^{T} & \hat{\boldsymbol{V}}_{t+1} \end{bmatrix}\right)$$

Derivation of RTS Smoother

Joint distribution

$\mathrm{cov}[\boldsymbol{x}_t, \boldsymbol{x}_{t+1}]$

$$p(\boldsymbol{x}_t, \boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:T})$$

$$= \mathrm{N}\left(\begin{bmatrix} \boldsymbol{x}_t \\ \boldsymbol{x}_{t+1} \end{bmatrix} \middle| \begin{bmatrix} \boldsymbol{m}_t + \boldsymbol{J}_t(\hat{\boldsymbol{m}}_{t+1} - \boldsymbol{A}\boldsymbol{m}_t) \\ \hat{\boldsymbol{m}}_{t+1} \end{bmatrix}, \begin{bmatrix} \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1} \boldsymbol{J}_t^T + \boldsymbol{V}_t - \boldsymbol{J}_t \boldsymbol{A} \boldsymbol{V}_t & \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1} \\ \hat{\boldsymbol{V}}_{t+1} \boldsymbol{J}_t^T & \hat{\boldsymbol{V}}_{t+1} \end{bmatrix} \right)$$

Marginal distribution = Smoothed dist. at *t*

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:T}) = \mathrm{N}\left(\boldsymbol{x}_t \mid \hat{\boldsymbol{m}}_t, \hat{\boldsymbol{V}}_t\right)$$

$$= \mathrm{N}\left(\boldsymbol{x}_t \mid \boldsymbol{m}_t + \boldsymbol{J}_t(\hat{\boldsymbol{m}}_{t+1} - \boldsymbol{A}\boldsymbol{m}_t), \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1} \boldsymbol{J}_t^T + \boldsymbol{V}_t - \boldsymbol{J}_t \boldsymbol{A} \boldsymbol{V}\right)$$

$$\begin{cases} \hat{\boldsymbol{m}}_t = \boldsymbol{m}_t + \boldsymbol{J}_t(\hat{\boldsymbol{m}}_{t+1} - \boldsymbol{A}\boldsymbol{m}_t) \\ \hat{\boldsymbol{V}}_t = \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1} \boldsymbol{J}_t^T + \boldsymbol{V}_t - \boldsymbol{J}_t \boldsymbol{A} \boldsymbol{V} = \boldsymbol{V}_t + \boldsymbol{J}_t(\hat{\boldsymbol{V}}_{t+1} - \boldsymbol{P}_t)\boldsymbol{J}_t^T \end{cases}$$

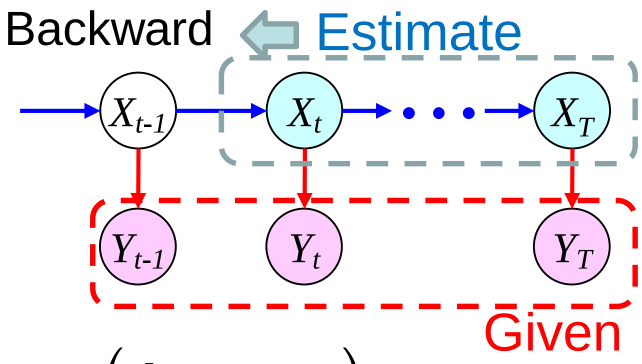
(Summary) RTS Smoother

- Assume filtered dist. $p(\mathbf{x}_t \mid \mathbf{y}_{1:t}) = \mathrm{N}(\mathbf{x}_t \mid \mathbf{m}_t, \mathbf{V}_t)$ have been computed by Kalman filtering
- At terminal time, $p(\boldsymbol{x}_T \mid \boldsymbol{y}_{1:T}) = \mathrm{N}(\boldsymbol{x}_T \mid \boldsymbol{m}_T, \boldsymbol{V}_T) = \mathrm{N}(\boldsymbol{x}_T \mid \hat{\boldsymbol{m}}_T, \hat{\boldsymbol{V}}_T)$
- For t=T-1 to 1, repeat the following computation

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:T}) = \mathrm{N}\left(\boldsymbol{x}_t \mid \hat{\boldsymbol{m}}_t, \hat{\boldsymbol{V}}_t\right)$$

where

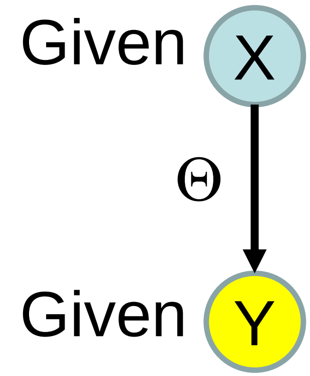
$$\begin{cases} \hat{\boldsymbol{m}}_t = \boldsymbol{m}_t + \boldsymbol{J}_t(\hat{\boldsymbol{m}}_{t+1} - \boldsymbol{A}\boldsymbol{m}_t) \\ \hat{\boldsymbol{V}}_t = \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1} \boldsymbol{J}_t^T + \boldsymbol{V}_t - \boldsymbol{J}_t \boldsymbol{A} \boldsymbol{V} = \boldsymbol{V}_t + \boldsymbol{J}_t\left(\hat{\boldsymbol{V}}_{t+1} - \boldsymbol{P}_t\right)\boldsymbol{J}_t^T \end{cases}$$



Maximum likelihood estimation

Case of Supervised Learning

- Assume generative models
  - Θ : parameters
- Classification (e.g., Bayesian classifier):
  - x: class (categorical scalar)
  - y: measurements (real / categorical vector)
- Regression (e.g., Linear Regression):
  - x: input (real vector)
  - y: output (real scalar/vector)
- Data : $D = \{x_i , y_i\}$ (i=1,..,N), Assume i.i.d.
- Log-likelihood function:

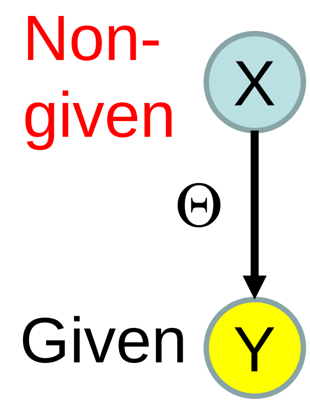


$$l(\Theta \mid D) \equiv \ln p(Y \mid X, \Theta) = \sum_{i=1}^{N} \ln p(y_i \mid x_i, \Theta)$$

Easy to maximize

Case of Unsupervised Learning

- Clustering (e.g., Gaussian mixture):
  - x: cluster (categorical scalar)
  - y: measurements (real / categorical vector)
- Dimensionality reduction
  - e.g., Factor analysis, Probabilistic PCA
  - x: factor (real vector), low-dimensional
  - y: measurements (real vector), high-dim.
- Data : D = $\{\boldsymbol{y}_\mathrm{i}\}$ (i=1,..,N), i.i.d. assumption
- Log-likelihood function:



$$l(\Theta \mid D) \equiv \ln p(Y \mid \Theta) = \ln \int p(Y, X \mid \Theta) dX$$

$$= \sum_{i=1}^{N} \ln \int p(y_i \mid x_i, \Theta) p(x_i \mid \Theta) dx_i$$

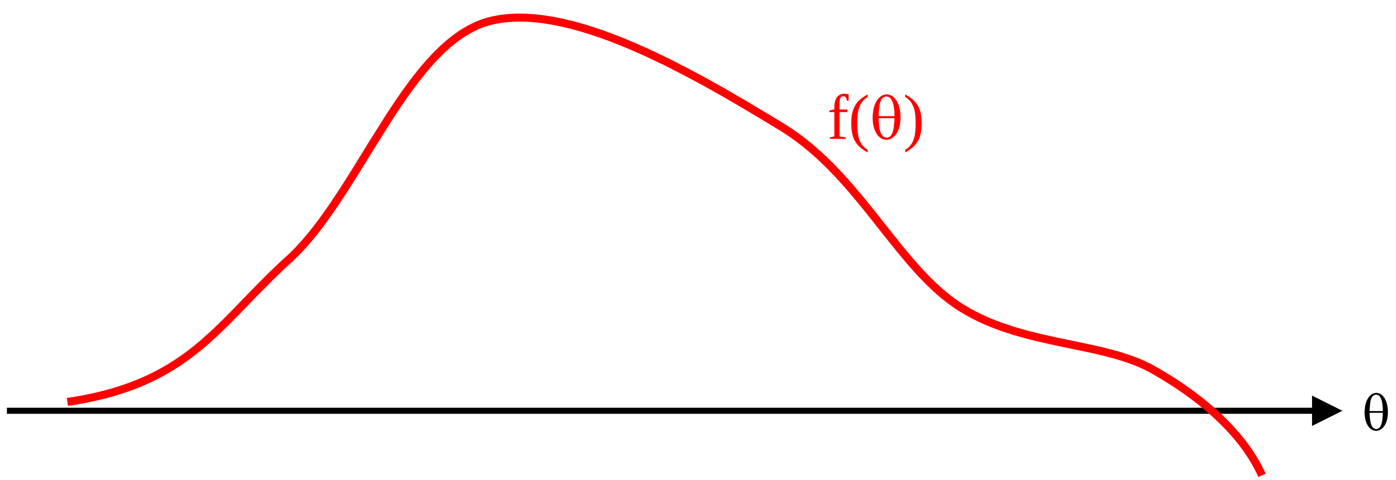
Complicated !

Hard to maximize

Minorization-Maximization Algorithm

Purpose of Minorization-Maximization (MM) Algorithm

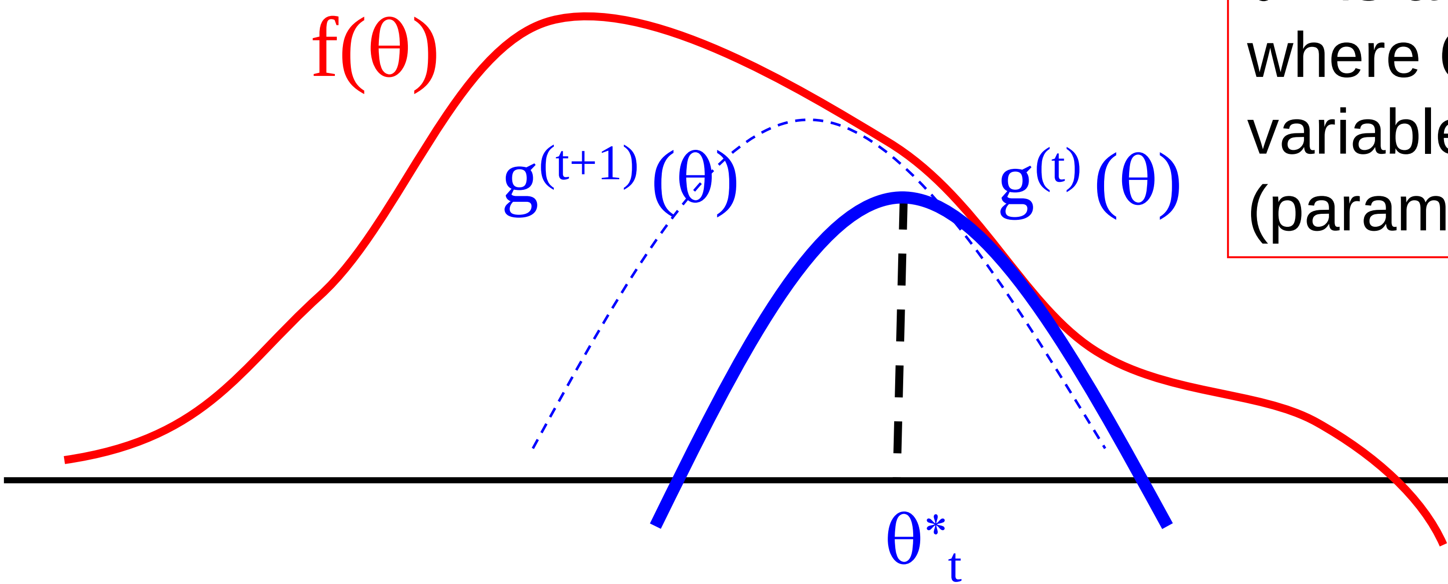
- Assume you want to maximize a function $f(\theta)$ with respect to θ
- But it is hard to maximize $f(\theta)$ directly..
  - because it is difficult to compute the 1st and 2nd derivatives
  - because θ is very high-dimensonal
  - Newton's method cannot be applied



Basic Idea of MM Algorithm

- Instead, consider maximizing a sequence of easier (surrogate) functions $\{ g^{(t)}(\theta) \}$ $(t=1,2,..)$
  - For example, quadratic functions are optimized easily
- But, what conditions should $\{ g^{(t)}(\theta) \}$ satisfy ?
- Obviously, $g^{(t)}(\theta)$ depends on the current solution ⇨ $g^{(t)}(\theta) = g(\theta | \theta^{t})$

$\theta^{(t)}$ is a constant where θ is a variable (parameter)



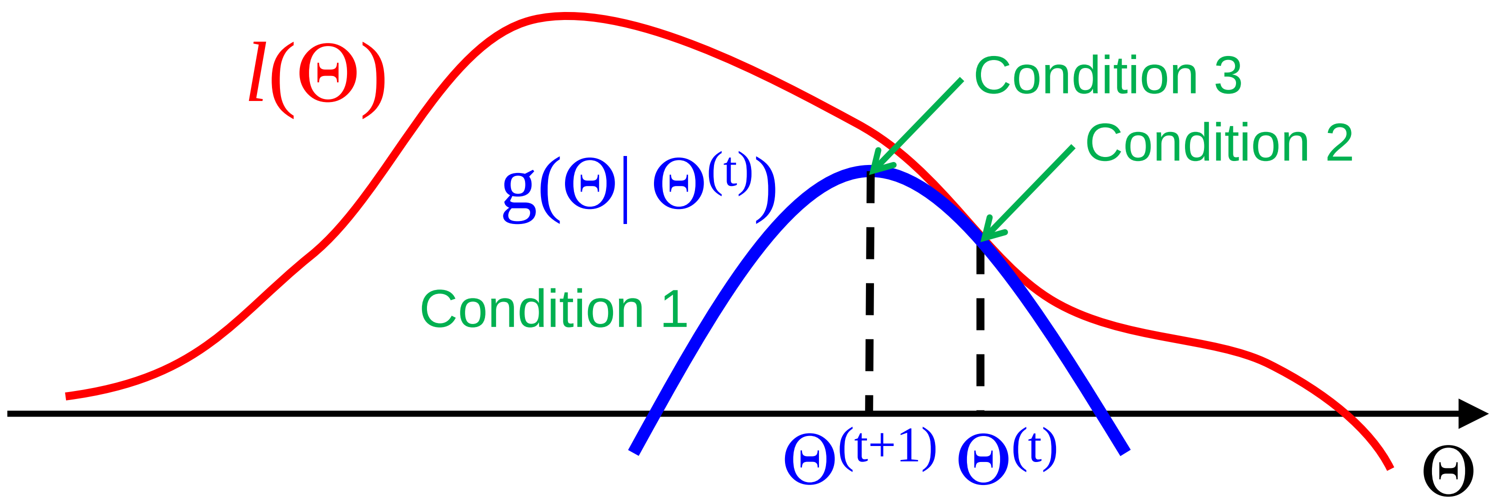
Conditions for Surrogate Functions

[Condtion 1] $g(\theta | \theta^{(t)})$ must be a lower bound of $f(\theta)$

i.e., $f(\theta) \geq g(\theta | \theta^{(t)})$ for any θ

[Condtion 2] $g(\theta^{(t)} | \theta^{(t)}) = f(\theta^{(t)})$

[Condtion 3] $g(\Theta | \Theta^{(t)})$ can be easily maximized



Minorization-Maximization method

- Assume these 3 conditions are satisfied, then determine $\Theta^{(t+1)}$ as,

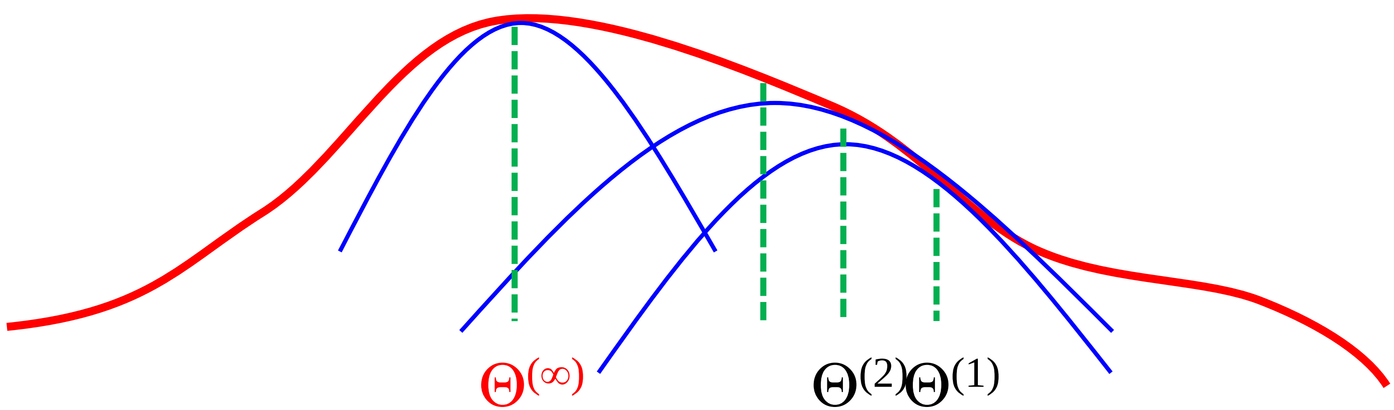
$$\Theta^{(t+1)} = \arg\max_{\Theta} g(\Theta|\Theta^{(t)})$$

- Then the following inequality holds,

$$l(\Theta^{(t+1)}) \geq g(\Theta^{(t+1)}|\Theta^{(t)}) \geq g(\Theta^{(t)}|\Theta^{(t)}) = l(\Theta^{(t)})$$

Condition 1 $\quad \Theta^{(t+1)} = \arg\max_{\Theta} g(\Theta|\Theta^{(t)}) \quad$ Condition 2

- This leads to $l(\Theta^{(1)}) \leq l(\Theta^{(2)}) \leq \ldots \leq l(\Theta^{(\infty)})$ (Local) Maximum



Jensen’s inequality

AM-GM Inequality
(相加相乗平均の不等式)

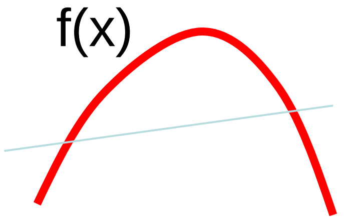
- $x_1, x_2, \ldots, x_\mathrm{n}$ are non-negative real numbers
- AM : Arithmetic Mean $\frac{1}{n}\left(x_1 + x_2 + \cdots + x_n\right)$
- GM : Geometric Mean $\sqrt[n]{x_1 \cdot x_2 \cdot \cdots \cdot x_n}$
- AM is greater than or equals to GM

$$\frac{1}{n}\left(x_1 + x_2 + \cdots + x_n\right) \geq \sqrt[n]{x_1 \cdot x_2 \cdot \cdots \cdot x_n}$$

- Equality holds if and only if $x_1 = x_2 = \cdots = x_n$
- Taking the logarithm of both sides,

$$\underbrace{\log \frac{1}{n}\sum_{i=1}^{n} x_i}_{\text{log-sum}} \geq \underbrace{\frac{1}{n}\sum_{i=1}^{n} \log x_i}_{\text{sum-log}} \xrightarrow{\text{Assume x is a r.v.}} \log \mathrm{E}\left[x\right] \geq \mathrm{E}\left[\log x\right]$$

Jensen's Inequality



- Assume X is a random variable and $f(X)$ is a concave function, then Jensen's inequality below holds

$$f(E[X]) \geq E[f(X)]$$

- If $f(X)$ is strictly concave, equality holds if and only if $X = \text{const.}$
- Now consider X is a discrete r.v. and $p(X=x_i) = q_i$ and $f(X) = \log X$ (log is strictly concave)

$$\mathrm{f}(\mathrm{E}[x]) = \log\left(\sum_{i=1}^{M} q_i \cdot x_i\right) \geq \mathrm{E}[\mathrm{f}(x)] = \sum_{i=1}^{M} q_i \cdot \log x_i$$

Equality holds when $x_1 = x_2 = \cdots = x_M$

EM algorithm

Apply Jensen's Inequality to Log-likelihood Function

- Get back to the maximization of log-likelihood function of unsupervised learning problem
  - Consider X is discrete r.v. to be simple

$$l(\Theta) \equiv \ln p(Y \mid \Theta) = \ln \sum_X p(Y, X \mid \Theta)$$

- Consider q(X) is a probabilistic distribution of X

$$l(\Theta) = \ln \sum_X p(Y, X \mid \Theta) = \ln \sum_X q(X) \frac{p(Y, X \mid \Theta)}{q(X)}$$

$$= \ln \mathrm{E}_{q(X)}\left[\frac{p(Y, X \mid \Theta)}{q(X)}\right]$$

$$\geq \mathrm{E}_{q(X)}\left[\ln \frac{p(Y, X \mid \Theta)}{q(X)}\right]$$

Jensen's Inequality

Equality Condition and Lower Bound

Equality holds when $\dfrac{p(Y, X \mid \Theta)}{q(X)} = \text{const.}$

$$q(X) \propto p(Y, X \mid \Theta) = p(X \mid Y, \Theta) \quad \left(\because \sum_X q(X) = 1 \right)$$

Now, define the lower bound function $g(\Theta \mid \Theta^{(t)})$:

$$g\left(\Theta \mid \Theta^{(t)}\right) \equiv \mathrm{E}_{p\left(X \mid Y, \Theta^{(t)}\right)}\left[\ln \frac{p(Y, X \mid \Theta)}{p\left(X \mid Y, \Theta^{(t)}\right)} \right]$$

[Condition 1] $l(\Theta) \geq g(\Theta \mid \Theta^{(t)})$ for all Θ ⇨ Satisfied

[Condition 2] $g(\Theta^{(t)} \mid \Theta^{(t)}) = l(\Theta^{(t)})$ ⇨ Satisfied

[Condition 3] $g(\Theta \mid \Theta^{(t)})$ can be easily maximized

Maximizing Lower Bound

Log-likelihood Lower bound

$$l(\Theta) \geq \mathrm{g}\left(\Theta \mid \Theta^{(t)}\right)$$ ⇨ Maximize w.r.t. Θ

$$\mathrm{g}\left(\Theta \mid \Theta^{(t)}\right) = \mathrm{E}_{p\left(X \mid Y, \Theta^{(t)}\right)}\left[\ln \frac{p(Y, X \mid \Theta)}{p\left(X \mid Y, \Theta^{(t)}\right)}\right]$$

$$= \underbrace{\mathrm{E}_{p\left(X \mid Y, \Theta^{(t)}\right)}\left[\boxed{\ln p(Y, X \mid \Theta)}\right]}_{\text{Function of } \Theta} - \underbrace{\mathrm{E}_{p\left(X \mid Y, \Theta^{(t)}\right)}\left[\ln p\left(X \mid Y, \Theta^{(t)}\right)\right]}_{\text{Constant w.r.t. } \Theta}$$

Called "complete" log likelihood

Function of Θ Constant w.r.t. Θ

|||

$\mathrm{Q}\left(\Theta \mid \Theta^{(t)}\right)$ Expected complete log-likelihood

In practice, we maximize Q(Θ|Θ $^{(t)}$) at each iteration

$$\Theta^{(t+1)} \leftarrow \arg\max_{\Theta} \mathrm{Q}\left(\Theta \mid \Theta^{(t)}\right)$$

Summary: EM Algorithm in General

Given:

- Data : Y
- Initial parameter values: $\Theta^{(0)}$

Repeat until convergence

1. [E-step] Compute posterior dist. $q^*(X) = p\left(X \mid Y, \Theta^{(t)}\right)$

   and $Q\left(\Theta \mid \Theta^{(t)}\right) = \mathrm{E}_{q^*(X)}\left[\ln p\left(Y, X \mid \Theta\right)\right]$

   Expected complete log-likelihood

2. [M-step] Maximize $Q(\Theta|\Theta^{(t)})$ w.r.t. Θ

   $$\Theta^{(t+1)} \leftarrow \arg\max_{\Theta} Q\left(\Theta \mid \Theta^{(t)}\right)$$

3. $t \leftarrow t + 1$

EM Algorithm for LDS

Problem of Learning LDS

Linear (Gaussian) Dynamical System

$$\boldsymbol{x}_t = \boldsymbol{A}\boldsymbol{x}_{t-1} + \boldsymbol{w}_t$$
$$\boldsymbol{y}_t = \boldsymbol{C}\boldsymbol{x}_t + \boldsymbol{v}_t$$
$$\boldsymbol{w}_t \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{Q})$$
$$\boldsymbol{v}_t \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{R})$$

- Given :
  - Observation sequence : $\boldsymbol{y}_{1:\mathrm{T}}$
- Find :
  - System matrices : $\boldsymbol{A}$, $\boldsymbol{B}$
  - Noise covariance matrices: $\boldsymbol{Q}$, $\boldsymbol{R}$
  - Initial state distribution: $p(\boldsymbol{x}_1) = \mathrm{N}(\boldsymbol{x}_1 \mid \boldsymbol{m}_0, \boldsymbol{V}_0)$
  - State sequence: $\boldsymbol{x}_{1:\mathrm{T}}$
    - Mean vectors and covariance matrices : $\boldsymbol{m}_{1:\mathrm{T}}$ and $\boldsymbol{V}_{1:\mathrm{T}}$

Likelihood Function of LDS

- Model parameters : $\Theta = \{\boldsymbol{A}\ ,\ \boldsymbol{B}\ ,\ \boldsymbol{Q}\ ,\ \boldsymbol{R}\ ,\ \boldsymbol{m}_0,\ \boldsymbol{V}_0\}$
- Data : $D = \{\boldsymbol{y}_{1:T}\}$
- Log-likelihood function:

$$l(\Theta \mid D) \equiv \ln p(\boldsymbol{y}_{1:T} \mid \Theta) = \ln \int p(\boldsymbol{y}_{1:T}, \boldsymbol{x}_{1:T} \mid \Theta) d\boldsymbol{x}_{1:T}$$

$$= \underbrace{\ln \int}_{\text{Integral in log !}} p(\boldsymbol{y}_{1:T} \mid \boldsymbol{x}_{1:T}, \Theta) p(\boldsymbol{x}_{1:T} \mid \Theta) d\boldsymbol{x}_{1:T}$$

Hard to optimize (maximize)

Much Easier Version
If we could observe state sequence

- Given :
  - Observation sequence : $\boldsymbol{y}_{1:T}$
  - State sequence: $\boldsymbol{x}_{1:T}$
    - Mean vectors and covariance matrices : $\boldsymbol{m}_{1:T}$ and $\boldsymbol{V}_{1:T}$
- Find :
  - System matrices : $\boldsymbol{A}$, $\boldsymbol{B}$
  - Noise covariance matrices: $\boldsymbol{Q}$, $\boldsymbol{R}$
  - Initial state distribution: $p(\boldsymbol{x}_1) = \mathrm{N}(\boldsymbol{x}_1 \mid \boldsymbol{m}_0, \boldsymbol{V}_0)$

Likelihood Function of Easier Version

- Model parameters : $\Theta = \{\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R}, \boldsymbol{m}_0, \boldsymbol{V}_0\}$
- Data : $D = \{\boldsymbol{y}_{1:T}, \boldsymbol{x}_{1:T}\}$
- Log-likelihood function:

$$l(\Theta \mid D) \equiv \underbrace{\ln p(\boldsymbol{y}_{1:T}, \boldsymbol{x}_{1:T} \mid \Theta)}_{\text{complete data log-likelihood}} = \ln p(\boldsymbol{y}_{1:T} \mid \boldsymbol{x}_{1:T}, \Theta) p(\boldsymbol{x}_{1:T} \mid \Theta)$$

$$= \ln\left(\prod_{t=1}^{T} p(\boldsymbol{y}_t \mid \boldsymbol{x}_t, \Theta) \cdot p(\boldsymbol{x}_1 \mid \Theta) \prod_{t=1}^{T-1} p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \Theta)\right)$$

$$= \ln p(\boldsymbol{x}_1 \mid \Theta) + \sum_{t=1}^{T-1} \ln p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \Theta) + \sum_{t=1}^{T} \ln p(\boldsymbol{y}_t \mid \boldsymbol{x}_t, \Theta)$$

Likelihood Function of Easier Version (cont.)

$$l(\Theta \mid D) = \ln p(\boldsymbol{x}_1 \mid \Theta) + \sum_{t=1}^{T-1} \ln p(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \Theta) + \sum_{t=1}^{T} \ln p(\boldsymbol{y}_t \mid \boldsymbol{x}_t, \Theta)$$

$$= \ln N(\boldsymbol{x}_1 \mid \boldsymbol{m}_0, \boldsymbol{V}_0) + \sum_{t=1}^{T-1} \ln N(\boldsymbol{x}_{t+1} \mid \boldsymbol{A}\boldsymbol{x}_t, \boldsymbol{Q}) + \sum_{t=1}^{T} \ln N(\boldsymbol{y}_t \mid \boldsymbol{C}\boldsymbol{x}_t, \boldsymbol{R})$$

$$\begin{aligned} = -\frac{1}{2}\Bigg\{ & (\boldsymbol{x}_1 - \boldsymbol{m}_0)^T \boldsymbol{V}_0^{-1} (\boldsymbol{x}_1 - \boldsymbol{m}_0) + \sum_{t=1}^{T-1} (\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t)^T \boldsymbol{Q}^{-1} (\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t) \\ & + \sum_{t=1}^{T} (\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t)^T \boldsymbol{R}^{-1} (\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t) + \ln|\boldsymbol{V}_0| + (T-1)\ln|\boldsymbol{Q}| + T \ln|\boldsymbol{R}| \Bigg\} \\ & + \text{const.} \end{aligned}$$

MLE Solution for Easier Version

Setting derivatives of $l(\Theta|D)$ w.r.t. parameters to 0

$$\frac{\partial l(\Theta)}{\partial \boldsymbol{m}_0} = \boldsymbol{V}_0^{-1}(\boldsymbol{x}_1 - \boldsymbol{m}_0) = \boldsymbol{0} \quad \Rightarrow \quad \hat{\boldsymbol{m}}_0 = \boldsymbol{x}_1$$

$$\frac{\partial l(\Theta)}{\partial \boldsymbol{V}_0} = -\frac{1}{2}\boldsymbol{V}_0^{-1}\left\{(\boldsymbol{x}_1 - \boldsymbol{m}_0)(\boldsymbol{x}_1 - \boldsymbol{m}_0)^T \boldsymbol{V}_0^{-1} - \boldsymbol{I}\right\} = \boldsymbol{0} \quad \Rightarrow \quad \hat{\boldsymbol{V}}_0 = \boldsymbol{0}$$

$$\frac{\partial l(\Theta)}{\partial \boldsymbol{A}} = \boldsymbol{Q}^{-1}\sum_{t=1}^{T-1}(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t)\boldsymbol{x}_t^{\ T} = \boldsymbol{0} \quad \Rightarrow \quad \hat{\boldsymbol{A}} = \left(\sum_{t=1}^{T-1}\boldsymbol{x}_{t+1}\boldsymbol{x}_t^{\ T}\right)\left(\sum_{t=1}^{T-1}\boldsymbol{x}_t\boldsymbol{x}_t^{\ T}\right)^{-1}$$

$$\frac{\partial l(\Theta)}{\partial \boldsymbol{Q}} = -\frac{1}{2}\boldsymbol{Q}^{-1}\left\{\sum_{t=1}^{T-1}(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t)(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t)^T \boldsymbol{Q}^{-1} - (T-1)\boldsymbol{I}\right\} = \boldsymbol{0}$$

$$\Rightarrow \quad \hat{\boldsymbol{Q}} = \tfrac{1}{T-1}\sum_{t=1}^{T-1}(\boldsymbol{x}_{t+1} - \hat{\boldsymbol{A}}\boldsymbol{x}_t)(\boldsymbol{x}_{t+1} - \hat{\boldsymbol{A}}\boldsymbol{x}_t)^T$$

$$\frac{\partial l(\Theta)}{\partial \boldsymbol{C}} = \boldsymbol{R}^{-1}\sum_{t=1}^{T}(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t)\boldsymbol{x}_t^{\ T} = \boldsymbol{0} \quad \Rightarrow \quad \hat{\boldsymbol{C}} = \left(\sum_{t=1}^{T}\boldsymbol{y}_t\boldsymbol{x}_t^{\ T}\right)\left(\sum_{t=1}^{T}\boldsymbol{x}_t\boldsymbol{x}_t^{\ T}\right)^{-1}$$

$$\frac{\partial l(\Theta)}{\partial \boldsymbol{R}} = -\frac{1}{2}\boldsymbol{R}^{-1}\left\{\sum_{t=1}^{T}(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t)(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t)^T \boldsymbol{R}^{-1} - T\cdot\boldsymbol{I}\right\} = \boldsymbol{0}$$

$$\Rightarrow \quad \hat{\boldsymbol{R}} = \tfrac{1}{T}\sum_{t=1}^{T}(\boldsymbol{y}_t - \hat{\boldsymbol{C}}\boldsymbol{x}_t)(\boldsymbol{y}_t - \hat{\boldsymbol{C}}\boldsymbol{x}_t)^T$$

Can be solved analytically

(Review) EM Algorithm in General

Given:

Observation sequence

- Data : Y $= \boldsymbol{y}_{1:T}$
- Initial parameter values: $\Theta^{(0)} = \{\boldsymbol{A}^{(0)}, \boldsymbol{B}^{(0)}, \boldsymbol{Q}^{(0)}, \boldsymbol{R}^{(0)}, \boldsymbol{m}_0^{(0)}, \boldsymbol{V}_0^{(0)}\}$

Repeat until convergence

Smoothed dist.

1. [E-step] Compute posterior dist. $q^*(X) = p\left(X \mid Y, \Theta^{(t)}\right)$

   and $\mathrm{Q}\left(\Theta \mid \Theta^{(t)}\right) = \mathrm{E}_{q^*(X)}\left[\ln p(Y, X \mid \Theta)\right]$

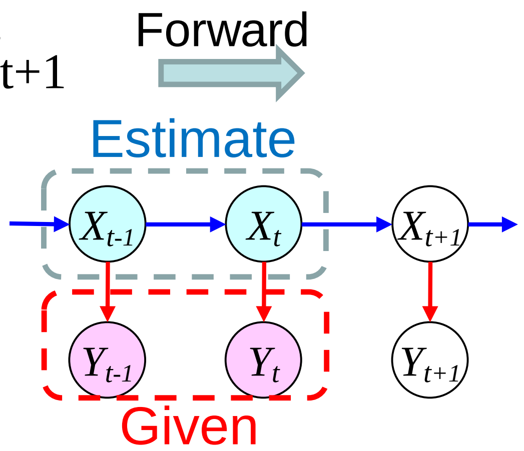
2. [M-step] Maximize $\mathrm{Q}(\Theta|\Theta^{(t)})$ w.r.t. Θ

$$\Theta^{(t+1)} \leftarrow \arg\max_{\Theta} \mathrm{Q}\left(\Theta \mid \Theta^{(t)}\right)$$

3. $t \leftarrow t+1$

E-step(1): Linear Kalman Filter

- Input : $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_t \mid \boldsymbol{m}_t, \boldsymbol{V}_t)$ and $\boldsymbol{y}_{t+1}$



- Prediction: $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t}) = \mathrm{N}(\boldsymbol{x}_{t+1} \mid \boldsymbol{A}\boldsymbol{m}_t, \boldsymbol{P}_t)$

  - where $\boldsymbol{P}_t = \boldsymbol{A}\boldsymbol{V}_t\boldsymbol{A}^T + \boldsymbol{Q}$

- Update :

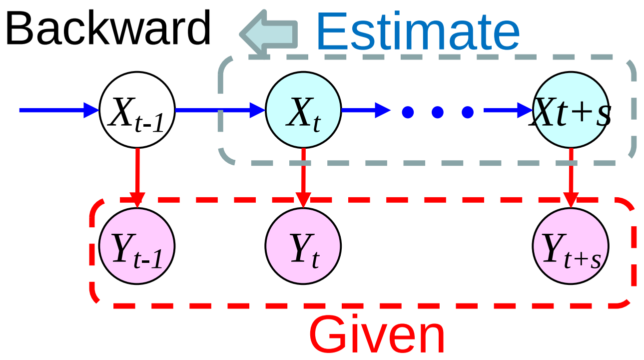
$$p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t+1}) = N\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{A}\boldsymbol{m}_t + \boldsymbol{K}_{t+1}(\boldsymbol{y}_{t+1} - \boldsymbol{C}\boldsymbol{A}\boldsymbol{m}_t), (\boldsymbol{I} - \boldsymbol{K}_{t+1}\boldsymbol{C})\boldsymbol{P}_t\right)$$

  - where $\boldsymbol{K}_{t+1} = \boldsymbol{P}_t\boldsymbol{C}^T\left(\boldsymbol{C}\boldsymbol{P}_t\boldsymbol{C}^T + \boldsymbol{R}\right)^{-1}$

- Output : $p(\boldsymbol{x}_{t+1} \mid \boldsymbol{y}_{1:t+1}) = \mathrm{N}(\boldsymbol{x}_{t+1} \mid \boldsymbol{m}_{t+1}, \boldsymbol{V}_{t+1})$

  - where
$$\begin{cases} \boldsymbol{m}_{t+1} = \boldsymbol{A}\boldsymbol{m}_t + \boldsymbol{K}_{t+1}(\boldsymbol{y}_{t+1} - \boldsymbol{C}\boldsymbol{A}\boldsymbol{m}_t) \\ \boldsymbol{V}_{t+1} = (\boldsymbol{I} - \boldsymbol{K}_{t+1}\boldsymbol{C})\boldsymbol{P}_t \end{cases}$$

E-step(2) : RTS Smoothing



- Assume filtered dist.
  $$p(\mathbf{x}_t \mid \mathbf{y}_{1:t}) = \mathrm{N}(\mathbf{x}_t \mid \mathbf{m}_t, \mathbf{V}_t)$$
  have been computed by Kalman filtering
- At terminal time, $p(\boldsymbol{x}_T \mid \boldsymbol{y}_{1:T}) = \mathrm{N}(\boldsymbol{x}_T \mid \boldsymbol{m}_T, \boldsymbol{V}_T) = \mathrm{N}\left(\boldsymbol{x}_T \mid \hat{\boldsymbol{m}}_T, \hat{\boldsymbol{V}}_T\right)$
- For t=T-1 to 1, repeat the following computation

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{1:T}) = \mathrm{N}\left(\boldsymbol{x}_t \,\middle|\, \hat{\boldsymbol{m}}_t, \hat{\boldsymbol{V}}_t\right)$$

where
$$\begin{cases} \hat{\boldsymbol{m}}_t = \boldsymbol{m}_t + \boldsymbol{J}_t(\hat{\boldsymbol{m}}_{t+1} - \boldsymbol{A}\boldsymbol{m}_t) \\ \hat{\boldsymbol{V}}_t = \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1} \boldsymbol{J}_t^T + \boldsymbol{V}_t - \boldsymbol{J}_t \boldsymbol{A} \boldsymbol{V} = \boldsymbol{V}_t + \boldsymbol{J}_t\left(\hat{\boldsymbol{V}}_{t+1} - \boldsymbol{P}_t\right)\boldsymbol{J}_t^T \end{cases}$$

and covariance between $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t+1}$ given $\boldsymbol{y}_{1:T}$:

$$\mathrm{cov}[\boldsymbol{x}_t, \boldsymbol{x}_{t+1}] = \boldsymbol{J}_t \hat{\boldsymbol{V}}_{t+1}$$

We require this later

E-step (3) : Miscllaneous Expectations

In M-step, we will require the following expectations:

$$\begin{cases} \mathrm{E}_{q^{(t)}(\boldsymbol{x}_{1:T})}\left[\boldsymbol{x}_t\right] = \hat{\boldsymbol{m}}_t \\ \mathrm{E}_{q^{(t)}(\boldsymbol{x}_{1:T})}\left[\boldsymbol{x}_t\,\boldsymbol{x}_t{}^T\right] = \mathrm{var}\left(\boldsymbol{x}_t\right) + \hat{\boldsymbol{m}}_t\hat{\boldsymbol{m}}_t{}^T = \hat{\boldsymbol{V}}_t + \hat{\boldsymbol{m}}_t\hat{\boldsymbol{m}}_t{}^T \\ \mathrm{E}_{q^{(t)}(\boldsymbol{x}_{1:T})}\left[\boldsymbol{x}_t\,\boldsymbol{x}_{t+1}{}^T\right] = \mathrm{cov}\left[\boldsymbol{x}_t, \boldsymbol{x}_{t+1}\right] + \hat{\boldsymbol{m}}_t\hat{\boldsymbol{m}}_{t+1}{}^T = \boldsymbol{J}_t\hat{\boldsymbol{V}}_{t+1} + \hat{\boldsymbol{m}}_t\hat{\boldsymbol{m}}_{t+1}{}^T \end{cases}$$

where, $q^{(t)}\left(\boldsymbol{x}_{1:T}\right) = p\left(\boldsymbol{x}_{1:T} \mid \boldsymbol{y}_{1:T}, \Theta^{(t)}\right)$

is the joint smoothed distribution of state sequence, given observation sequence and estimated parameter at t-th iteration

M-step(1) : Complete Log-Likelihood

We've got already the complete log-likelihood

$$\ln p\left(\boldsymbol{y}_{1:T}, \boldsymbol{x}_{1:T} \mid \Theta\right)$$

$$= -\frac{1}{2}\Bigg\{\left(\boldsymbol{x}_1 - \boldsymbol{m}_0\right)^T \boldsymbol{V}_0^{-1}\left(\boldsymbol{x}_1 - \boldsymbol{m}_0\right) + \sum_{t=1}^{T-1}\left(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t\right)^T \boldsymbol{Q}^{-1}\left(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t\right)$$

$$+ \sum_{t=1}^{T}\left(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t\right)^T \boldsymbol{R}^{-1}\left(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t\right) + \ln\left|\boldsymbol{V}_0\right| + \left(T-1\right)\ln\left|\boldsymbol{Q}\right| + T\ln\left|\boldsymbol{R}\right|\Bigg\}$$

$$+ \text{const.}$$

$$\mathrm{Q}\left(\Theta \mid \Theta^{(t)}\right) = \mathrm{E}_{q^*(X)}\left[\ln p\left(\boldsymbol{y}_{1:T}, \boldsymbol{x}_{1:T} \mid \Theta\right)\right]$$

Compute the expectation of this function w.r.t. state sequence $\{\boldsymbol{x}_{1:\mathrm{T}}\}$ and maximize w.r.t. parameters $\Theta = \{\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R}, \boldsymbol{m}_0, \boldsymbol{V}_0\}$!

M-step (2) : Update of Parameters

$$\frac{\partial \mathrm{Q}}{\partial \boldsymbol{m}_0} = \boldsymbol{V}_0^{-1}\left(\mathrm{E}[\boldsymbol{x}_1] - \boldsymbol{m}_0\right) = \boldsymbol{0} \quad \Rightarrow \quad \boldsymbol{m}_0^{(t+1)} = \mathrm{E}[\boldsymbol{x}_1] = \hat{\boldsymbol{m}}_1$$

$$\frac{\partial \mathrm{Q}}{\partial \boldsymbol{V}_0} = -\frac{1}{2}\boldsymbol{V}_0^{-1}\left\{\mathrm{E}\left[(\boldsymbol{x}_1 - \boldsymbol{m}_0)(\boldsymbol{x}_1 - \boldsymbol{m}_0)^T\right]\boldsymbol{V}_0^{-1} - \boldsymbol{I}\right\} = \boldsymbol{0} \quad \Rightarrow \quad \boldsymbol{V}_0^{(t+1)} = \mathrm{E}\left[\boldsymbol{x}_1\boldsymbol{x}_1^{\ T}\right] - \boldsymbol{m}_0\boldsymbol{m}_0^{\ T} = \hat{\boldsymbol{V}}_1$$

$$\frac{\partial \mathrm{Q}}{\partial \boldsymbol{A}} = -\tfrac{1}{2}\boldsymbol{Q}^{-1}\sum_{t=1}^{T-1}\left(\mathrm{E}\left[\boldsymbol{x}_{t+1}\boldsymbol{x}_t^{\ T}\right] - \boldsymbol{A}\,\mathrm{E}\left[\boldsymbol{x}_t\boldsymbol{x}_t^{\ T}\right]\right) = \boldsymbol{0} \quad \Rightarrow \quad \boldsymbol{A}^{(t+1)} = \left(\sum_{t=1}^{T-1}\mathrm{E}\left[\boldsymbol{x}_{t+1}\boldsymbol{x}_t^{\ T}\right]\right)\left(\sum_{t=1}^{T-1}\mathrm{E}\left[\boldsymbol{x}_t\boldsymbol{x}_t^{\ T}\right]\right)^{-1}$$

$$\frac{\partial \mathrm{Q}}{\partial \boldsymbol{Q}} = -\frac{1}{2}\boldsymbol{Q}^{-1}\left\{\sum_{t=1}^{T-1}\mathrm{E}\left[(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t)(\boldsymbol{x}_{t+1} - \boldsymbol{A}\boldsymbol{x}_t)^T\right]\boldsymbol{Q}^{-1} - (T-1)\boldsymbol{I}\right\} = \boldsymbol{0}$$

$$\Rightarrow \quad \boldsymbol{Q}^{(t+1)} = \tfrac{1}{T-1}\sum_{t=1}^{T-1}\left\{\mathrm{E}\left[\boldsymbol{x}_{t+1}\boldsymbol{x}_{t+1}^{\ T}\right] - \boldsymbol{A}^{(t+1)}\,\mathrm{E}\left[\boldsymbol{x}_t\boldsymbol{x}_{t+1}^{\ T}\right] - \mathrm{E}\left[\boldsymbol{x}_{t+1}\boldsymbol{x}_t^{\ T}\right]\boldsymbol{A}^{(t+1)T} + \boldsymbol{A}^{(t+1)}\,\mathrm{E}\left[\boldsymbol{x}_t\boldsymbol{x}_t^{\ T}\right]\boldsymbol{A}^{(t+1)T}\right\}$$

$$\frac{\partial \mathrm{Q}}{\partial \boldsymbol{C}} = \boldsymbol{R}^{-1}\sum_{t=1}^{T}\left(\boldsymbol{y}_t\,\mathrm{E}\left[\boldsymbol{x}_t^{\ T}\right] - \boldsymbol{C}\,\mathrm{E}\left[\boldsymbol{x}_t\boldsymbol{x}_t^{\ T}\right]\right) = \boldsymbol{0} \quad \Rightarrow \quad \boldsymbol{C}^{(t+1)} = \left(\sum_{t=1}^{T}\boldsymbol{y}_t\,\mathrm{E}\left[\boldsymbol{x}_t^{\ T}\right]\right)\left(\sum_{t=1}^{T}\mathrm{E}\left[\boldsymbol{x}_t\boldsymbol{x}_t^{\ T}\right]\right)^{-1}$$

$$\frac{\partial \mathrm{Q}}{\partial \boldsymbol{R}} = -\frac{1}{2}\boldsymbol{R}^{-1}\left\{\sum_{t=1}^{T}\mathrm{E}\left[(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t)(\boldsymbol{y}_t - \boldsymbol{C}\boldsymbol{x}_t)^T\right]\boldsymbol{R}^{-1} - T\cdot\boldsymbol{I}\right\} = \boldsymbol{0}$$

$$\Rightarrow \quad \boldsymbol{R}^{(t+1)} = \tfrac{1}{T}\sum_{t=1}^{T}\left\{\boldsymbol{y}_t\boldsymbol{y}_t^{\ T} - \boldsymbol{C}^{(t+1)}\,\mathrm{E}[\boldsymbol{x}_t]\boldsymbol{y}_t^{\ T} - \boldsymbol{y}_t\,\mathrm{E}\left[\boldsymbol{x}_t^{\ T}\right]\boldsymbol{C}^{(t+1)T} + \boldsymbol{C}^{(t+1)}\,\mathrm{E}\left[\boldsymbol{x}_t\boldsymbol{x}_t^{\ T}\right]\boldsymbol{C}^{(t+1)T}\right\}$$

Summary

- EM algorithm for learning linear dynamical systems was derived
- E-step (inference) is actually Kalman (RTS) smoother
- M-step is similar to ordinary maximum likelihood estimation for supervised learning
- This framework can be extended to many complicated (i.e., non-linear, switching, etc.) models
  - Often some approximation is necessary, though.
  - We will see such extensions next time