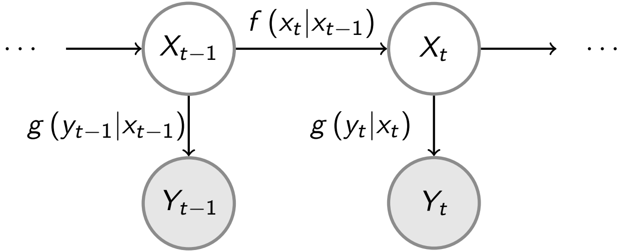
Hidden Markov Models and Bayesian Inference

Sinan Yıldırım
Sabancı University, Turkey

June 13, 2023

Hidden Markov models



The joint law of all the variables of the HMM up to time n

$$p(x_{1:n}, y_{1:n}) = \underbrace{\eta(x_1) \prod_{t=2}^{n} f(x_t|x_{t-1})}_{\text{hidden Markov process}} \underbrace{\prod_{t=1}^{n} g(y_t|x_t)}_{\text{observations}}$$

Some distributions of interest

The joint law of all the variables of the HMM up to time n

$$p(x_{1:n}, y_{1:n}) = \underbrace{\eta(x_1) \prod_{t=2}^{n} f(x_t|x_{t-1})}_{\text{hidden Markov process}} \underbrace{\prod_{t=1}^{n} g(y_t|x_t)}_{\text{observations}}$$

Marginal likelihood the observations up to time n which can be derived as

$$p(y_{1:n}) = \int p(x_{1:n}, y_{1:n}) dx_{1:n}.$$

Posterior distribution of $X_{1:n}$ given $Y_{1:n} = y_{1:n}$, which is obtained by using the Bayes' theorem

$$p(x_{1:n}|y_{1:n}) = \frac{p(x_{1:n}, y_{1:n})}{p(y_{1:n})}$$

Bayesian inference - General

Observation y, with likelihood $p(y|x)$

Parameter x, with prior $p(x)$

Want to estimate x based on y

Bayesian inference

Evaluate the posterior distribution

$$p(x|y) = \frac{p(x)p(y|x)}{p(y)} \propto p(x)p(y|x)$$

A finite state-space HMM for weather conditions

- $X_t \in \{1, 2\}$ denotes the state of the atmospheric condition in terms of pressure; "Low" (1) or "High" (2)

  Initial and transition probabilities:

  $$\eta = [\eta(1), \eta(2)], \quad F = \begin{bmatrix} 0.3 & 0.7 \\ 0.2 & 0.8 \end{bmatrix}$$

  where $F(i, j) = \mathbb{P}(X_{t+1} = j | X_t = i) = f(j|i)$.
- $Y_t \in \{1, 2, 3\}$: Day is "Dry" (1), "Cloudy" (2), "Rainy" (3),

  $$G = \begin{bmatrix} 0.3 & 0.4 & 0.3 \\ 0.6 & 0.3 & 0.1 \end{bmatrix}$$

  where $G(i, j) = \mathbb{P}(Y_t = j | X_t = i) = g(j|i)$.

Linear Gaussian HMM

We have $\{X_t, Y_t\}$, where $X_t \in \mathbb{R}^{d_x}$, and $Y_t \in \mathbb{R}^{d_y}$

$$X_1 \sim \mathcal{N}(\mu_1, \Sigma_1), \quad X_t = AX_{t-1} + U_t, \quad U_t \sim \mathcal{N}(0, S), \quad t > 1$$
$$Y_t = BX_t + V_t, \quad V_t \sim \mathcal{N}(0, R),$$

In terms of transition and observation densities

$$\eta(x_1) = \mathcal{N}(x_1; \mu_1, \Sigma_1), \quad f(x_t|x_{t-1}) = \mathcal{N}(x_t; Ax_{t-1}, S),$$
$$g(y_t|x_t) = \mathcal{N}(y_t; Bx_t, R).$$

A partially observed moving target - Target dynamics

$X_t = (V_t, P_t)$ with

- $V_t = (V_t(1), V_t(2))$ velocity
- $P_t = (P_t(1), P_t(2))$ position

State dynamics

$$V_1(i) \sim \mathcal{N}(0, \sigma_{bv}^2), \quad P_1(i) \sim \mathcal{N}(0, \sigma_{bp}^2), \quad i = 1, 2,$$
$$V_t(i) = aV_{t-1}(i) + U_t(i), \quad P_t(i) = P_{t-1}(i) + \Delta V_{t-1}(i) + Z_t(i), \quad i = 1, 2.$$

where $U_t \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_v^2)$ and $Z_t(i) \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_p^2)$.
Transition density

$$f(x_t|x_{t-1}) = \phi(Fx_{t-1}, \Sigma_x), \quad F = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & a & 0 & 0 \\ \Delta & 0 & 1 & 0 \\ 0 & \Delta & 0 & 1 \end{bmatrix}, \Sigma_x = \begin{bmatrix} \sigma_v^2 & 0 & 0 & 0 \\ 0 & \sigma_v^2 & 0 & 0 \\ 0 & 0 & \sigma_p^2 & 0 \\ 0 & 0 & 0 & \sigma_p^2 \end{bmatrix}$$

A partially observed moving target - Observation process

At each time t three distance measurements $(R_{t,1}, R_{t,2}, R_{t,3})$ with

$$R_{t,i} = [(P_t(1) - S_i(1))^2 + (P_t(2) - S_i(2))^2]^{1/2}, \quad i = 1, 2, 3,$$

from three different sensors are collected in Gaussian noise with variance σ_y^2 and these measurements form $Y_t = (Y_{t,1}, Y_{t,2}, Y_{t,3})$

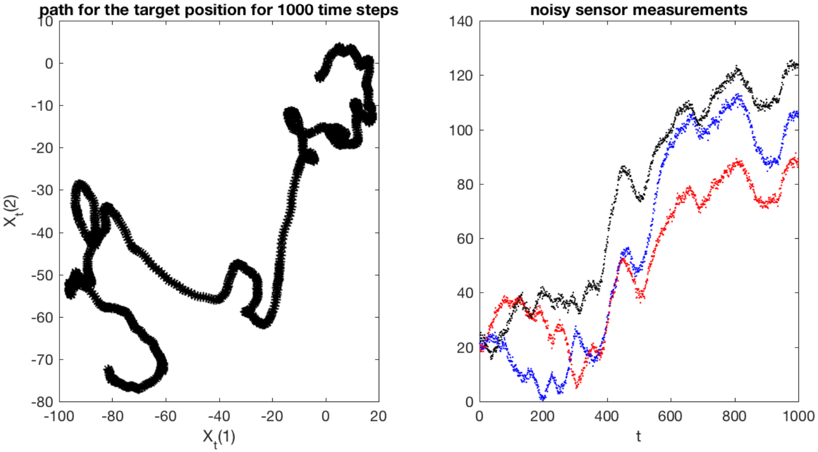
$$Y_{t,i} = R_{t,i} + E_{t,i}, \quad E_{t,i} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_y^2), \quad i = 1, 2, 3.$$

so that

$$g(y_t | x_t) = \prod_{i=1}^{3} \mathcal{N}(y_{t,i}; r_{t,i}, \sigma_y^2).$$

This is an example to a non-linear HMM due to the non-linearity in its observation dynamics.

Example - target tracking



Bayesian optimal filtering and smoothing

Goal: Estimate the hidden process $\{X_t\}_{t\geq 1}$ given observations $\{Y_t\}_{t\geq 1}$ up to time n.

The sequence of posterior distributions

$$p(x_{1:t}|y_{1:t}), \quad t \geq 1$$

For $t' > t$ we have

$$p(x_{1:t'}|y_{1:t}) = p(x_{1:t}|y_{1:t}) \prod_{\tau=t+1}^{t'} f(x_\tau|x_{\tau-1});$$

For $t' < t$,

$$p(x_{1:t'}|y_{1:t}) = \int p(x_{1:t}|y_{1:t}) dx_{t'+1:t}.$$

Filtering, prediction, smoothing

Consider the posterior distribution

$$p(x_k | y_{1:n})$$

- Filtering: $k = n$
- Prediction: $k > n$
- Smoothing: $k < n$

Expectations of functions $\varphi : \mathcal{X} \to \mathbb{R}$ of X_k given $y_{1:n}$

$$\mathbb{E}\left[\varphi(X_k) | Y_{1:n} = y_{1:n}\right] = \int \varphi(x_k) p(x_k | y_{1:n}) dx_k .$$

Forward filtering (and prediction):

Initialization:

$$p(x_1|y_1) = \frac{\eta(x_1)g(y_1|x_1)}{\int \eta(x_1')g(y_1|x_1')dx_1}.$$

Given y_t,

Filtering at time $t-1 \rightarrow$ Prediction for time t

$$p(x_t|y_{1:t-1}) = \int f(x_t|x_{t-1})p(x_{t-1}|y_{1:t-1})dx_{t-1}.$$

Prediction for time $t \rightarrow$ Filtering at time t

$$p(x_t|y_{1:t}) = \frac{g(y_t|x_t)p(x_t|y_{1:t-1})}{\int p(x_t'|y_{1:t-1})g(y_t|x_t')dx_t'}.$$

Backward smoothing

Start with $p(x_n|y_{1:n-1})$.
For $t = n, n-1, \ldots, 1$, update from $p(x_{t+1}|y_{1:n})$ to $p(x_t|y_{1:n})$:

$$p(x_t|y_{1:n}) = \int p(x_{t+1}|y_{1:n}) p(x_t|x_{t+1}, y_{1:n}) dx_{t+1}.$$

Given X_{t+1}, X_t is conditionally independent from the rest of the future variables. Hence

$$p(x_t|x_{t+1}, y_{1:n}) = \frac{p(x_t|y_{1:t}) f(x_{t+1}|x_t)}{p(x_{t+1}|y_{1:t})}$$

Backward smoothing

$$p(x_t|y_{1:n}) = \int p(x_{t+1}|y_{1:n}) \frac{p(x_t|y_{1:t}) f(x_{t+1}|x_t)}{p(x_{t+1}|y_{1:t})} dx_{t+1}$$

Finite state-space HMM

$X_t \in \mathcal{X} = \{1, \dots, k\}$.

Define

- **Filtering probabilities:**

$$\alpha_t(i) := \mathbb{P}(X_t = i | Y_{1:t} = y_{1:t}), \quad i = 1, \dots, k, \quad t = 1, \dots, n$$

- **Prediction probabilities:**

$$\beta_t(i) := \mathbb{P}(X_t = i | Y_{1:t-1} = y_{1:t-1}), \quad i = 1, \dots, k, \quad t = 1, \dots, n$$

- **Smoothing probabilities:**

$$\gamma_t(i) := \mathbb{P}(X_t = i | Y_{1:n} = y_{1:n}), \quad i = 1, \dots, k, \quad t = 1, \dots, n$$

FFBS for HMM

Forward filtering

for $t = 1, \dots, n$ **do**

Prediction: If $t = 1$, set $\beta_1(i) = \eta(i)$, $i = 1, \dots, k$; else

$$\beta_t(i) = \sum_{j=1}^{k} \alpha_{t-1}(j) f(i|j), \quad i = 1, \dots, k.$$

Filtering:

$$\alpha_t(i) = \frac{\beta_t(i) g(y_t|i)}{\sum_{j=1}^{k} \beta_t(j) g(y_t|j)}, \quad i = 1, \dots, k.$$

Backward smoothing

for $t = n, \dots, 1$ **do**

Smoothing: If $t = n$, set $\gamma_n(i) = \alpha_n(i)$, $i = 1, \dots, k$; else

$$\gamma_t(i) = \sum_{j=1}^{k} \gamma_{t+1}(j) \frac{\alpha_t(i) f(j|i)}{\beta_{t+1}(j)}, \quad i = 1, \dots, k.$$

Linear Gaussian HMM (LG-HMM)

Recall the linear Gaussian HMM

$$X_1 \sim \mathcal{N}(\mu_1, \Sigma_1), \quad X_t = AX_{t-1} + U_t, \quad U_t \sim \mathcal{N}(0, S), \quad t > 1$$
$$Y_t = BX_t + V_t, \quad V_t \sim \mathcal{N}(0, R)$$

The filtering, prediction, and smoothing distributions are all Gaussian:

$$\begin{aligned} p(x_t|y_{1:t}) &= \mathcal{N}(x_t; \mu_{t|t}, P_{t|t}), && t = 1, \dots, n, \\ p(x_t|y_{1:t-1}) &= \mathcal{N}(x_t; \mu_{t|t-1}, P_{t|t-1}), && t = 1, \dots, n, \\ p(x_t|y_{1:n}) &= \mathcal{N}(x_t; \mu_{t|n}, P_{t|n}), && t = 1, \dots, n. \end{aligned}$$

The mean and the covariance of these distributions are tractable.

Kalman Filtering: Forward filtering in LG-HMM

for $t = 1, \dots, n$ **do**

Prediction:

if $t = 1$ **then**

Set $\mu_{1|0} = \mu_1$, $P_{1|0} = \Sigma_1$

else

$$\mu_{t|t-1} = A\mu_{t-1|t-1},$$
$$P_{t|t-1} = AP_{t-1|t-1}A^T + S$$

Filtering:

$$P^y_{t|t-1} = BP_{t|t-1}B^T + R,$$
$$\mu^y_{t|t-1} = B\mu_{t|t-1}, \quad P^{xy}_{t|t-1} = P_{t|t-1}B^T$$
$$\mu_{t|t} = \mu_{t|t-1} + P^{xy}_{t|t-1}{P^y_{t|t-1}}^{-1}(y_t - \mu^y_{t|t-1})$$
$$P_{t|t} = P_{t|t-1} - P^{xy}_{t|t-1}{P^y_{t|t-1}}^{-1}{P^{xy}_{t|t-1}}^T$$

Backward smoothing in LG-HMM

Start with $\mu_{n|n}$ and $P_{n|n}$.

for $t = n-1, \ldots, 1$ **do**

$$\Gamma_{t|t+1} = P_{t|t} A^T P_{t+1|t}^{-1}$$
$$\mu_{t|n} = \mu_{t|t} + \Gamma_{t|t+1}(\mu_{t|n} - \mu_{t+1|t})$$
$$P_{t|n} = P_{t|t} + \Gamma_{t|t+1}(P_{t+1|n} - P_{t+1|t})\Gamma_{t|t+1}^T$$

Particle Filtering

HMM: Target posterior distributions

Joint distribution

$$p(x_{1:n}, y_{1:n}) = \eta(x_1) \prod_{t=2}^{n} f(x_t|x_{t-1}) \prod_{t=1}^{n} g(y_t|x_t)$$

Posterior distribution of $x_{1:n}$ given $y_{1:n}$

$$p(x_{1:n}|y_{1:n}) = \frac{p(x_{1:n}, y_{1:n})}{p(y_{1:n})} \propto p(x_{1:n}, y_{1:n})$$

Goal: Sequentially estimate $p(x_{1:n}|y_{1:n})$.

Sequential Importance Sampling

Sequential importance sampling

Target distribution: $p(x_{1:n}|y_{1:n}) \propto p(x_{1:n}, y_{1:n})$
Want to approximate with a discrete distribution

$$p(x_{1:n}|y_{1:n}) \approx \sum_{i=1}^{N} W_n^{(i)} \delta_{X_{1:n}^{(i)}}(x_{1:n})$$

For this, we need a proposal distribution

$$Q_n(x_{1:n}|y_{1:n}) = q_1(x_1) \prod_{t=1}^{n} q_t(x_t|x_{1:t-1})$$

The weight function:

$$w_n(x_{1:n}) = \frac{p(x_{1:n}, y_{1:n})}{Q_n(x_{1:n})}.$$

Recursion on the weight function:

$$w_n(x_{1:n}) = w_{n-1}(x_{1:n-1}) \frac{f(x_n|x_{n-1})g(y_n|x_n)}{q_n(x_n|x_{1:n-1})}.$$

With N particles $X_{1:n}^{(i)} \sim Q_n(x_{1:n})$,

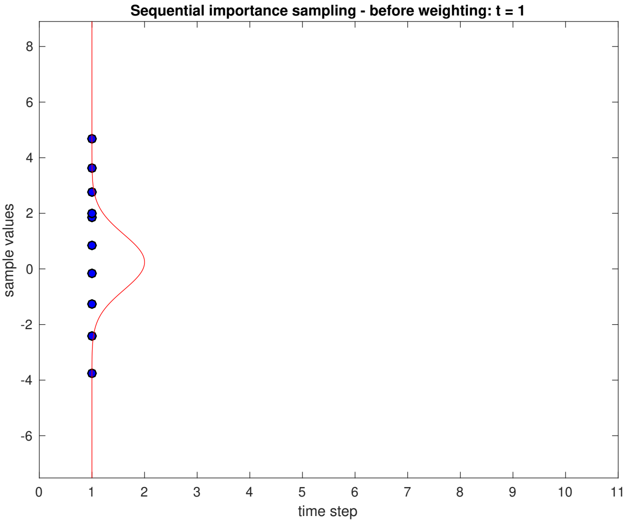
$$W_n^{(i)} = \frac{w_n(X_{1:n}^{(i)})}{\sum_{i=1}^{N} w_n(X_{1:n}^{(i)})}.$$

Sequential importance sampling

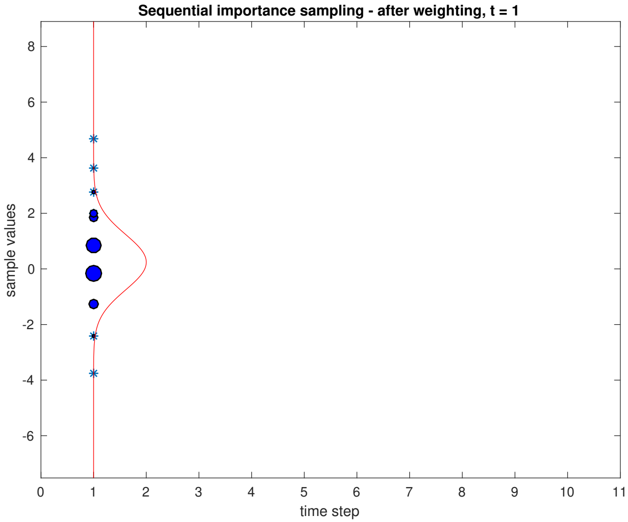
For $n = 1, 2, \ldots;$

- For $i = 1, \ldots, N$,
  - If $n = 1$, draw $X_1^{(i)} \sim q_1(\cdot|y_1)$ and calculate
  $$w_1(X_1^{(i)}) = \frac{\eta(X_1^{(i)}) g(y_1|X_1^{(i)})}{q_1(X_1^{(i)})}.$$
  - If $n \geq 2$, draw $X_n^{(i)} \sim q_n(\cdot|X_{1:n-1}^{(i)})$, set $X_{1:n}^{(i)} = (X_{1:n-1}^{(i)}, X_n^{(i)})$ and calculate
  $$w_n(X_{1:n}^{(i)}) = w_{n-1}(X_{1:n-1}^{(i)}) \frac{f(X_n^{(i)}|X_{n-1}^{(i)}) g(y_n|X_n^{(i)})}{q_n(X_n^{(i)}|X_{1:n-1}^{(i)})}.$$
- Importance weights: For $i = 1, \ldots, N$, calculate
$$W_n^{(i)} = \frac{w_n(X_{1:n}^{(i)})}{\sum_{i=1}^N w_n(X_{1:n}^{(i)})}.$$

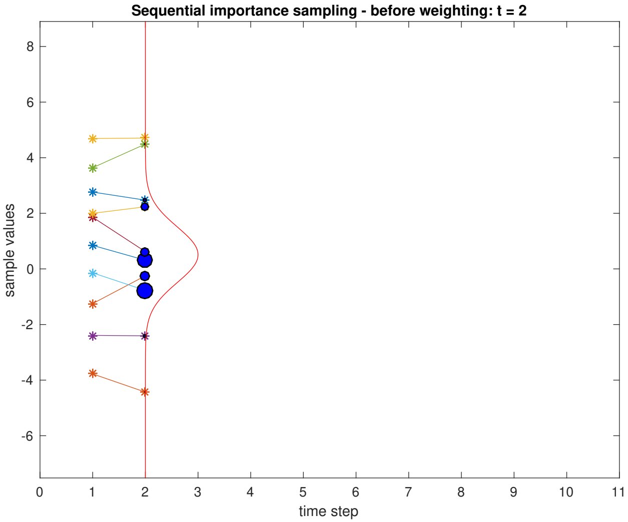
SIS - Weight degeneracy problem



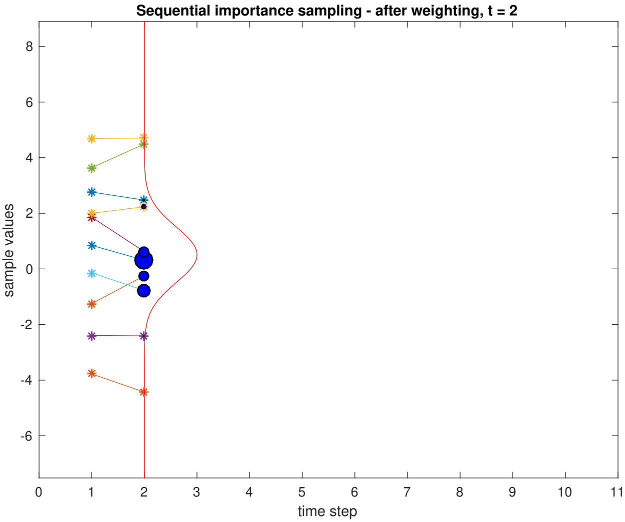
SIS - Weight degeneracy problem



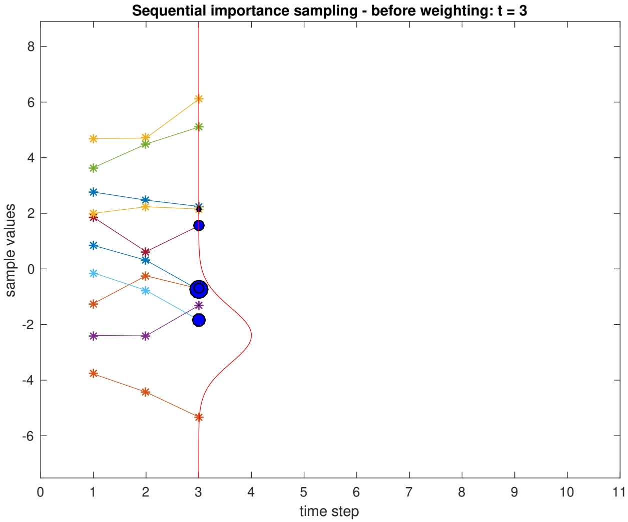
SIS - Weight degeneracy problem



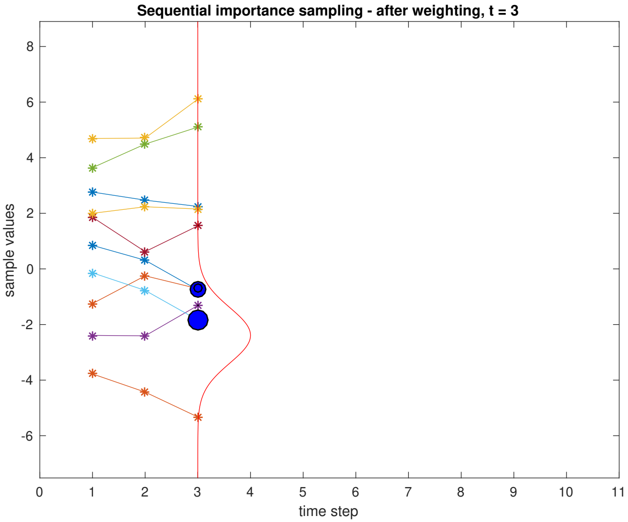
SIS - Weight degeneracy problem



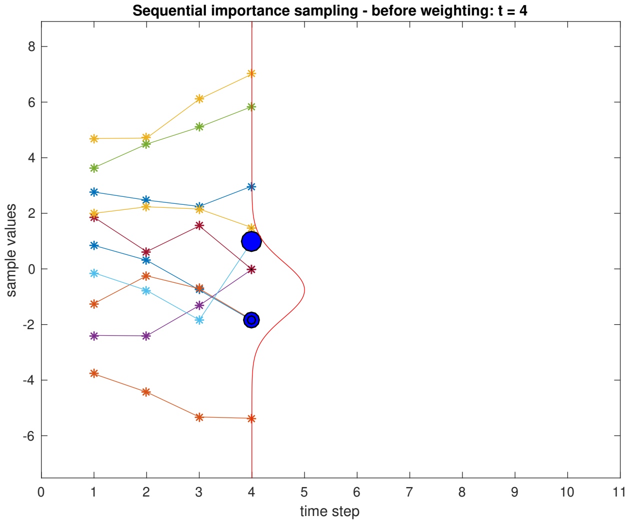
SIS - Weight degeneracy problem



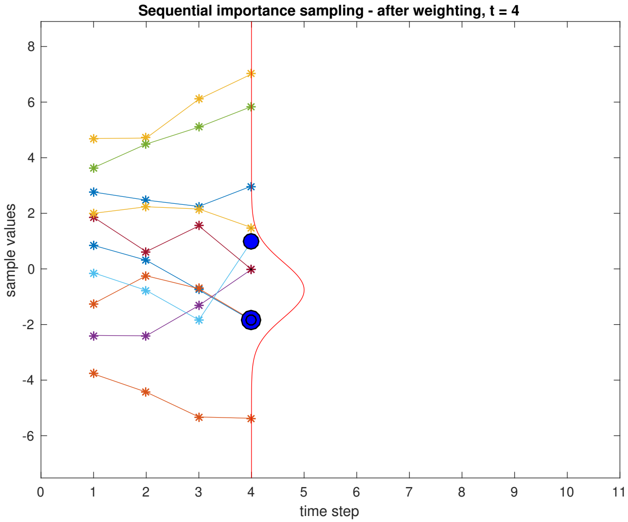
SIS - Weight degeneracy problem



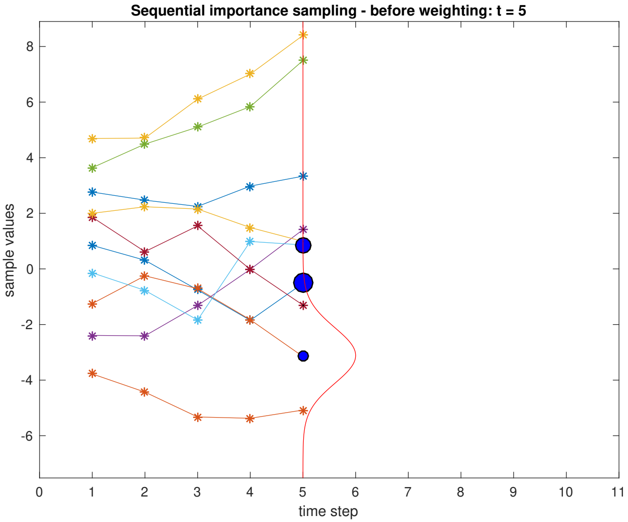
SIS - Weight degeneracy problem



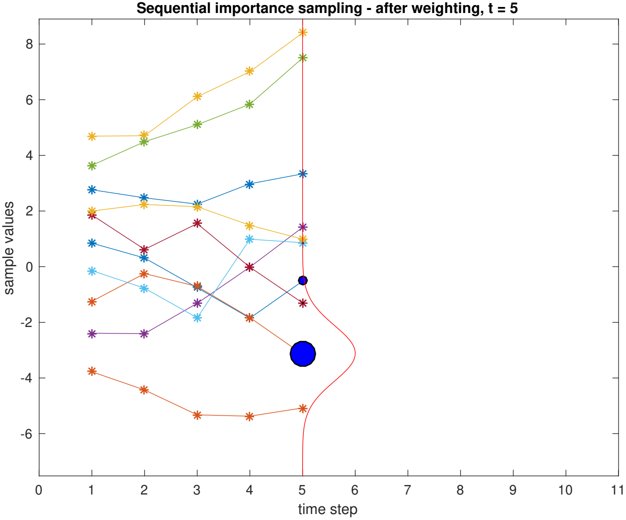
SIS - Weight degeneracy problem



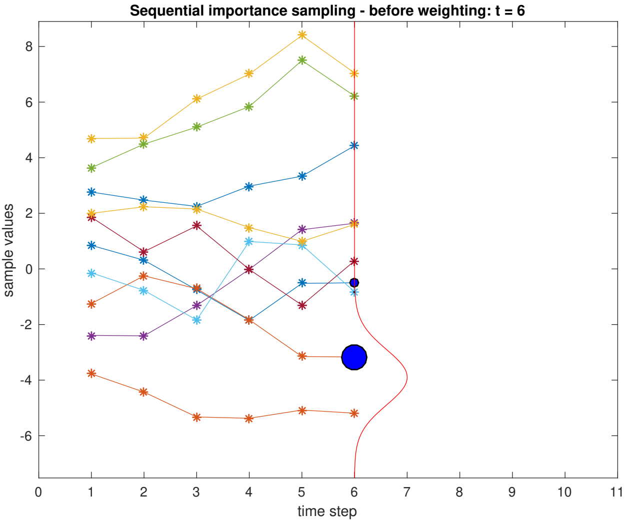
SIS - Weight degeneracy problem



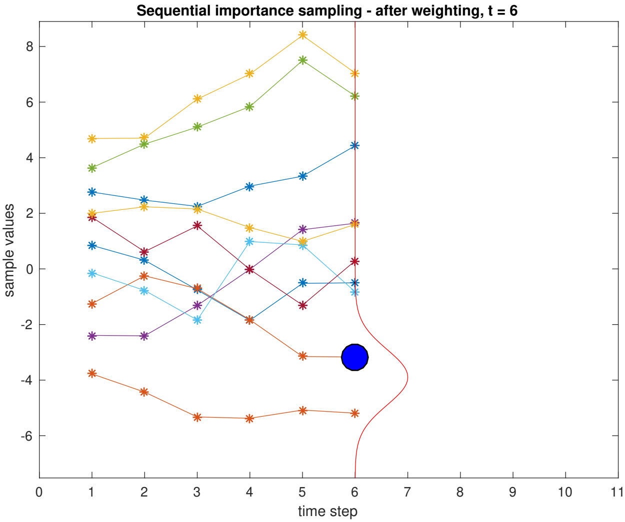
SIS - Weight degeneracy problem



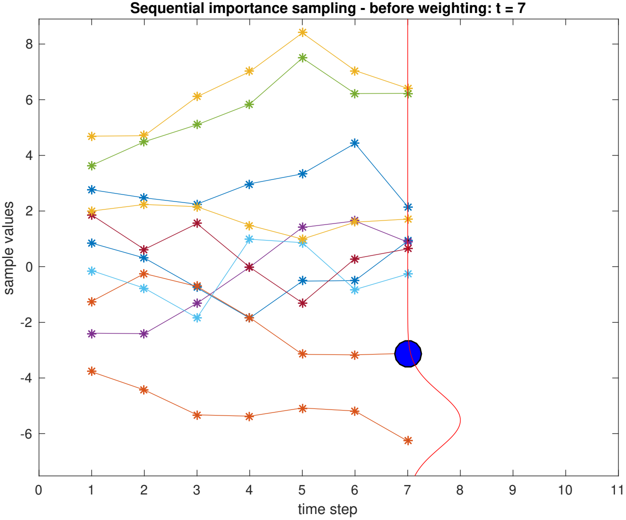
SIS - Weight degeneracy problem



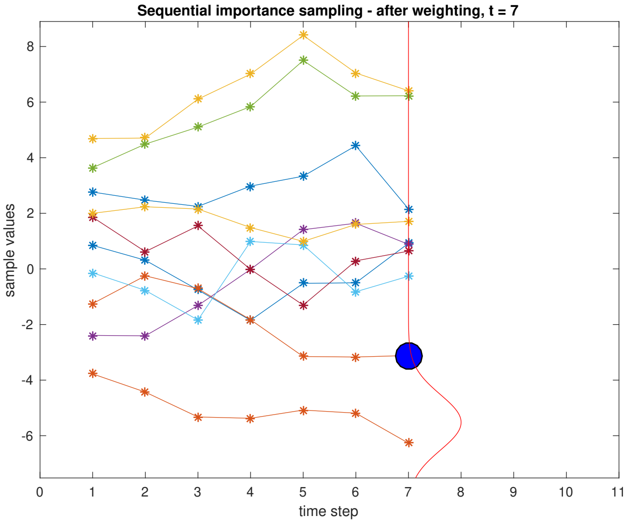
SIS - Weight degeneracy problem



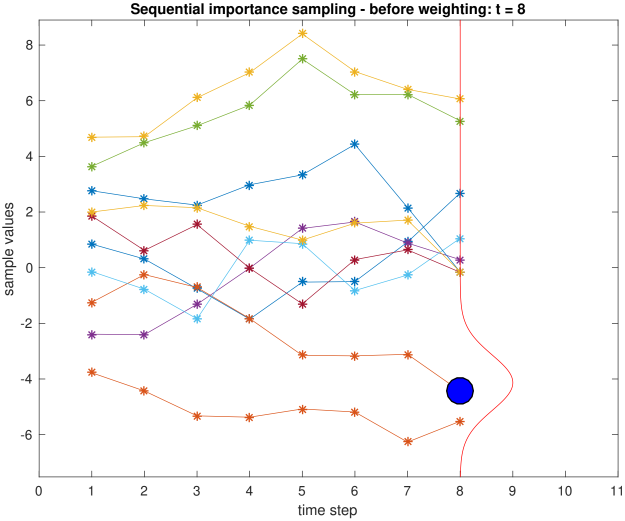
SIS - Weight degeneracy problem



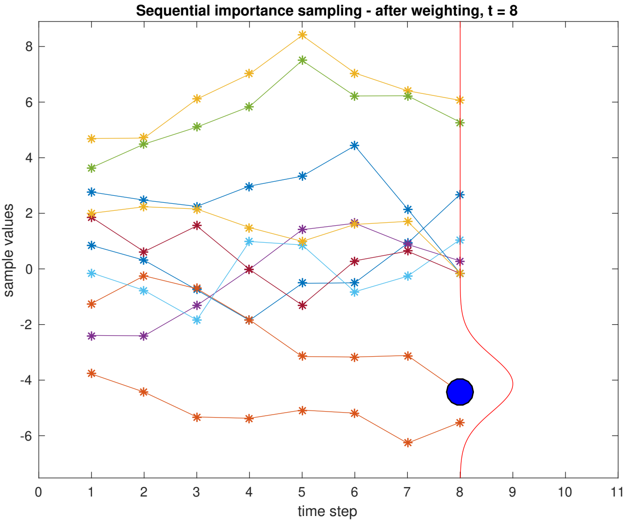
SIS - Weight degeneracy problem



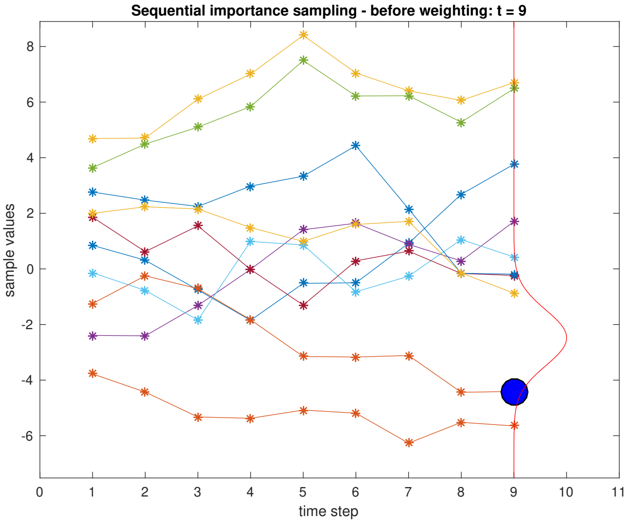
SIS - Weight degeneracy problem



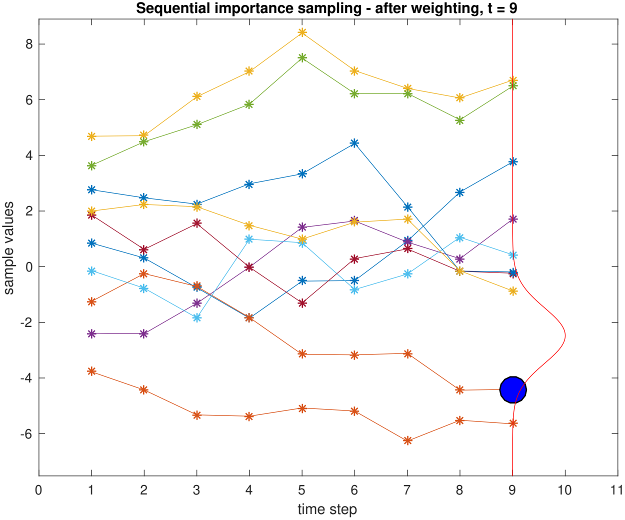
SIS - Weight degeneracy problem



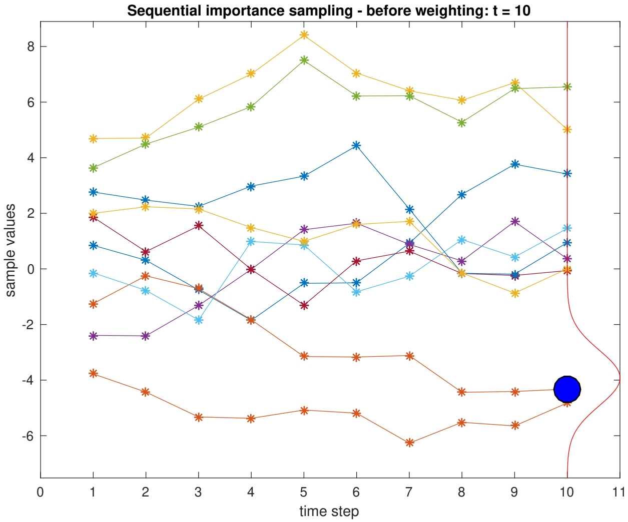
SIS - Weight degeneracy problem



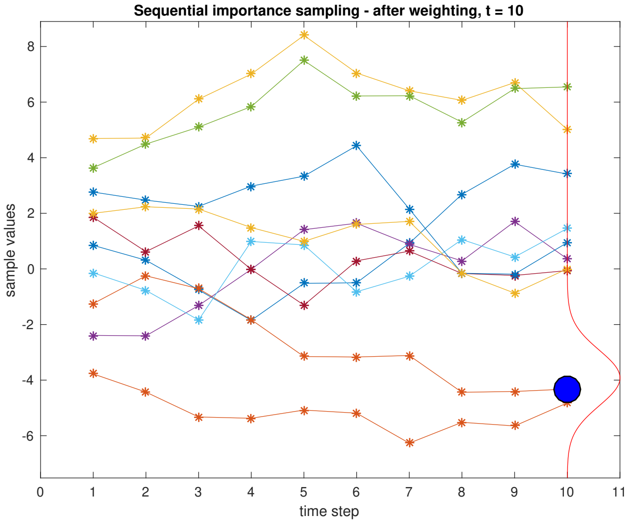
SIS - Weight degeneracy problem



SIS - Weight degeneracy problem



SIS - Weight degeneracy problem

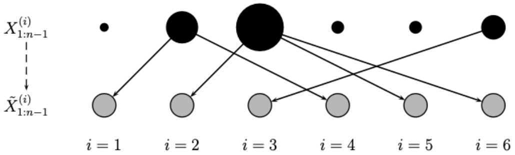


SIS + Resampling → SIS-R a.k.a. Particle filter

Resampling: Assume at time $n-1$ we have particles $X_{1:n-1}^{(1)}, \ldots, X_{1:n-1}^{(N)}$ with weights $W_{n-1}^{(1)}, \ldots, W_{n-1}^{(N)}$.

Draw N new particles among $X_{1:n-1}^{(1)}, \ldots, X_{1:n-1}^{(N)}$ according to their weights independently.

$$P(\widetilde{X}_{1:n-1}^{(i)} = X_{1:n-1}^{(j)}) = W_{n-1}^{(j)}, \quad i, j = 1, \ldots, N.$$



Proceed with the resampled particles $\widetilde{X}_{1:n-1}^{(1)}, \ldots, \widetilde{X}_{1:n-1}^{(N)}$ with equal weights $1/N$

Particle filter for HMM

For $n = 1$;

For $i = 1, \ldots, N$ draw $X_1^{(i)} \sim q_1(\cdot)$ and calculate $W_1^{(i)} \propto \frac{\eta(X_1^{(i)}) g(y_1 | X_1^{(i)})}{q_1(X_1^{(i)})}$.

For $n = 2, 3, \ldots,$

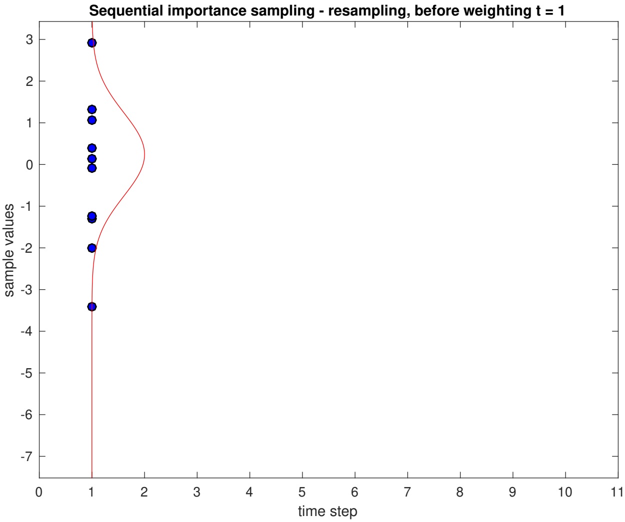
- Generate $\widetilde{X}_{1:n-1}^{(1)}, \ldots, \widetilde{X}_{1:n-1}^{(N)}$ by resampling:

$$\mathbb{P}(\widetilde{X}_{1:n-1}^{(i)} = X_{1:n-1}^{(j)}) = W_{n-1}^{(j)}, \quad i, j = 1, \ldots, N.$$

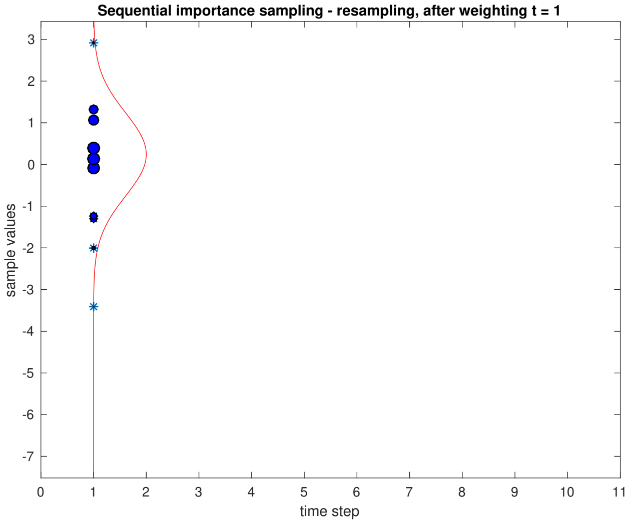
- For $i = 1, \ldots, N$, draw $X_n^{(i)} \sim q_n(\cdot | \widetilde{X}_{1:n-1}^{(i)})$, set $X_{1:n}^{(i)} = (\widetilde{X}_{1:n-1}^{(i)}, X_n^{(i)})$.
- Weights

$$W_n^{(i)} \propto \frac{f(X_n^{(i)} | \widetilde{X}_{n-1}^{(i)}) g(y_n | X_n^{(i)})}{q_n(X_n^{(i)} | \widetilde{X}_{1:n-1}^{(i)})}.$$

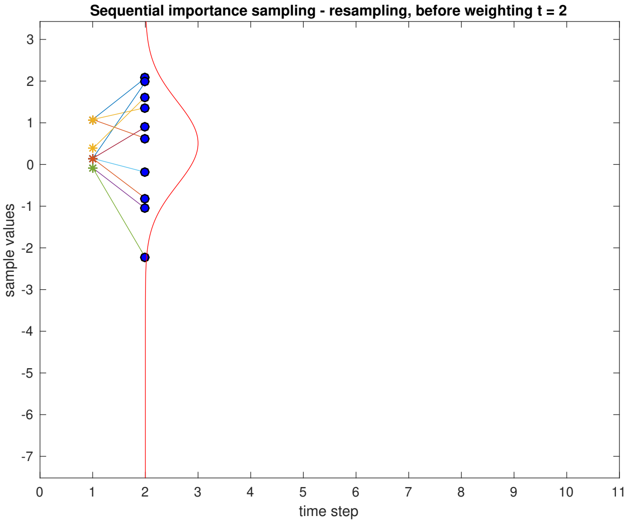
Particle filter



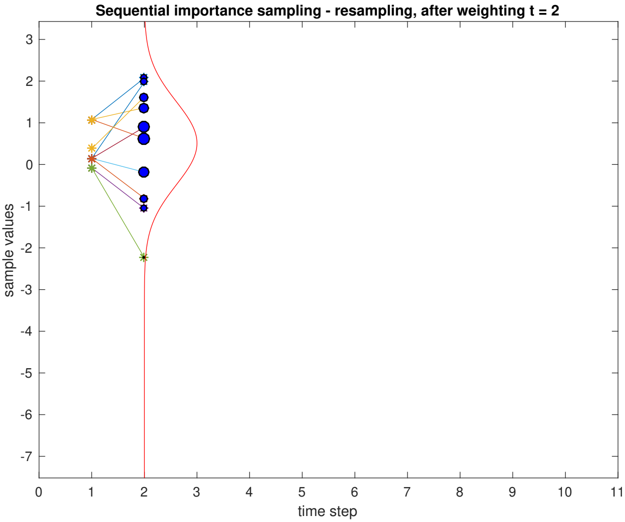
Particle filter



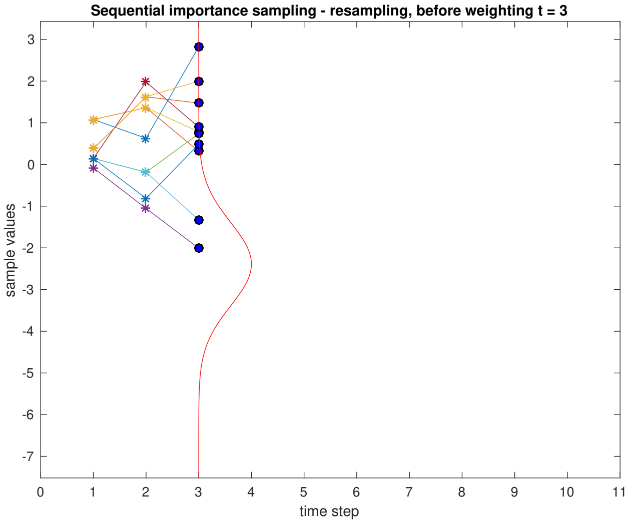
Particle filter



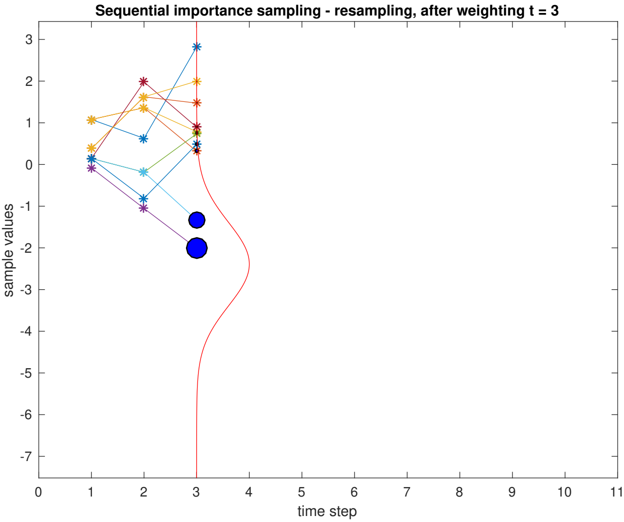
Particle filter



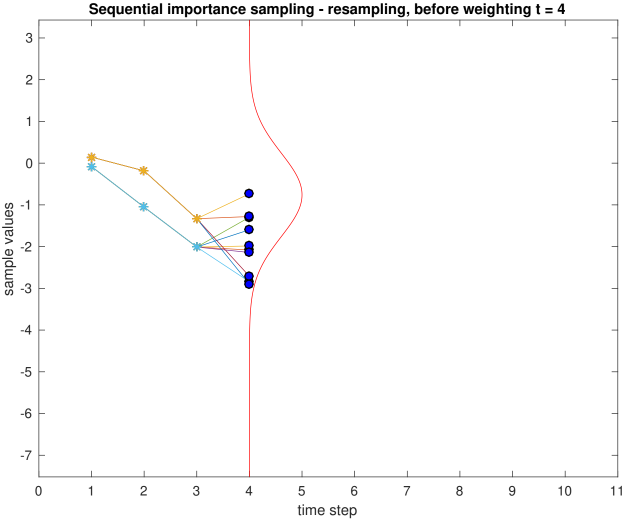
Particle filter



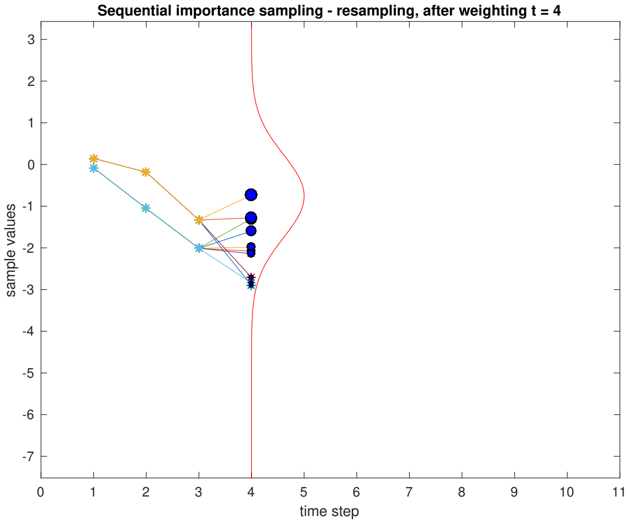
Particle filter



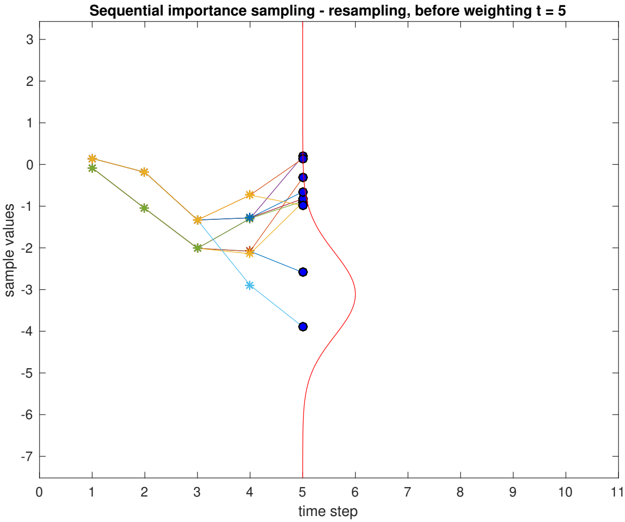
Particle filter



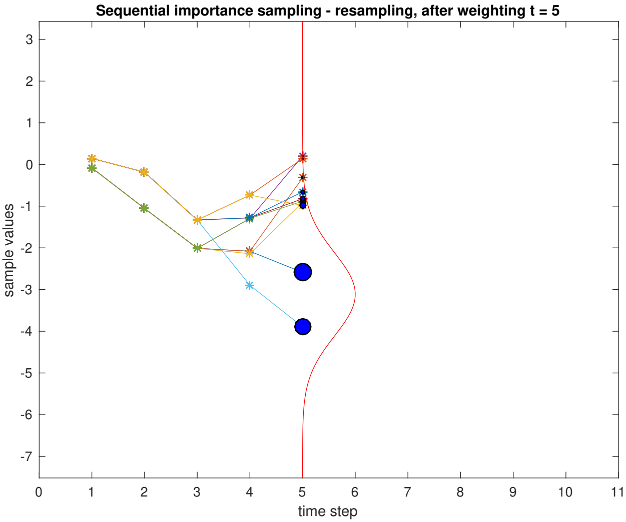
Particle filter



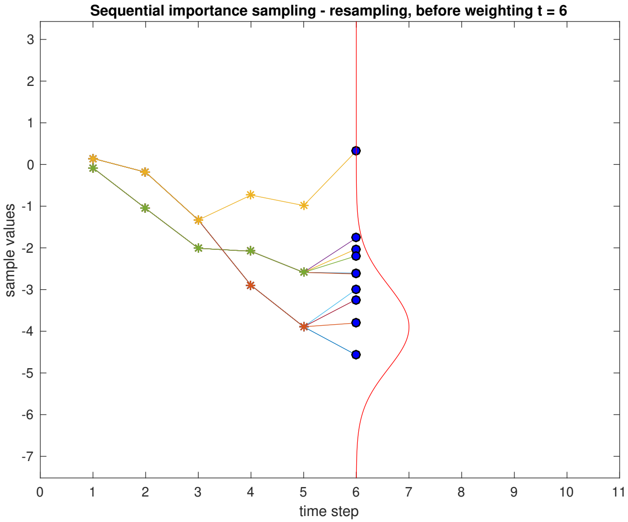
Particle filter



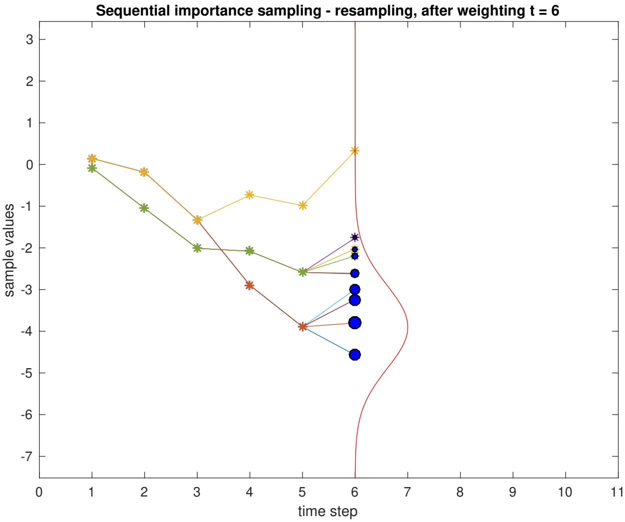
Particle filter



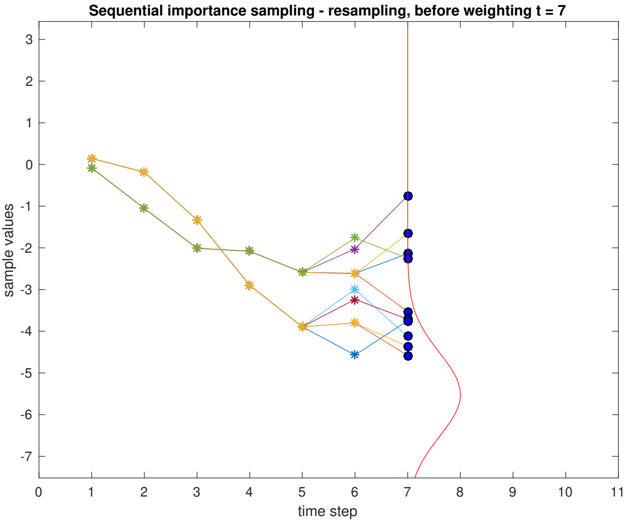
Particle filter



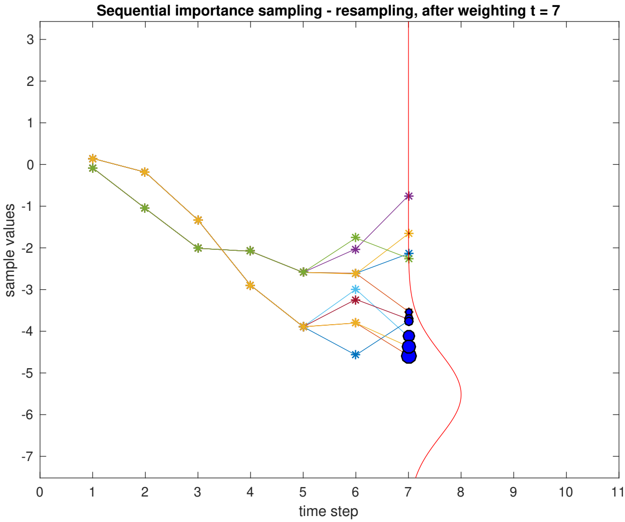
Particle filter



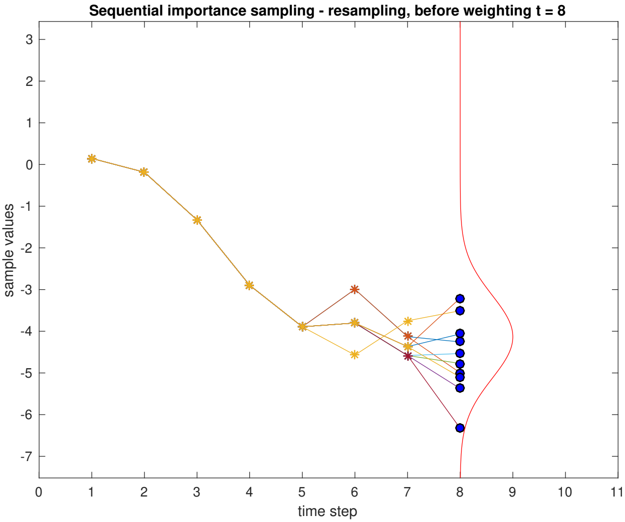
Particle filter



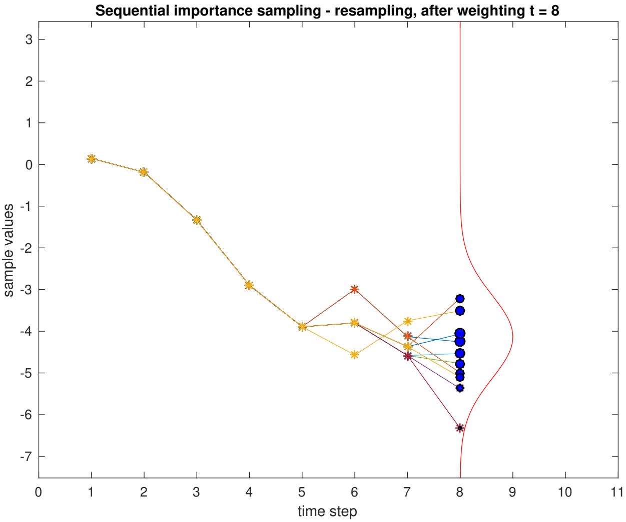
Particle filter



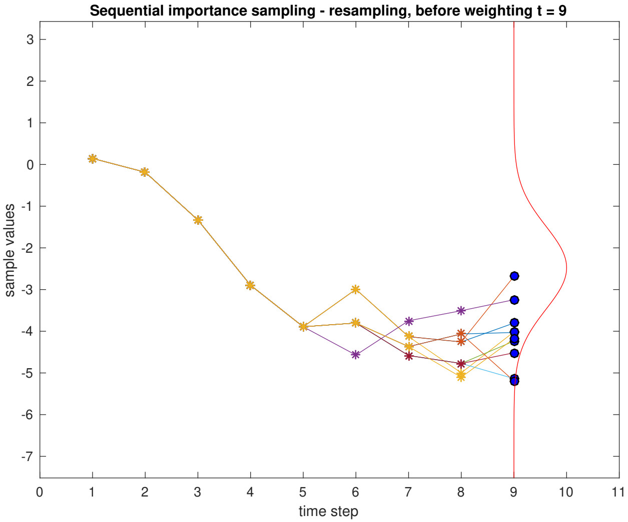
Particle filter



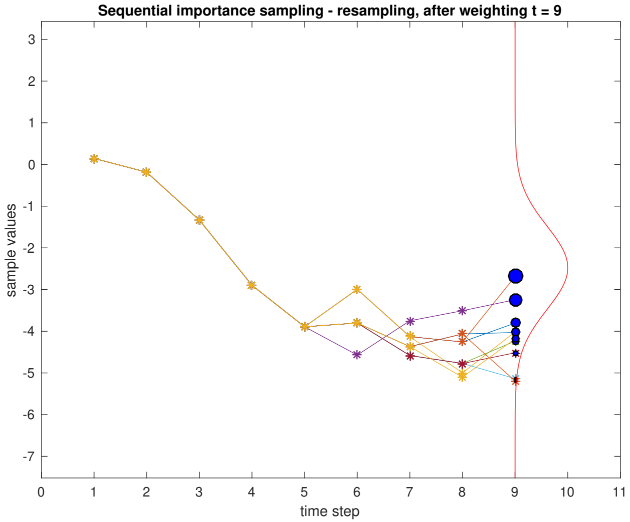
Particle filter



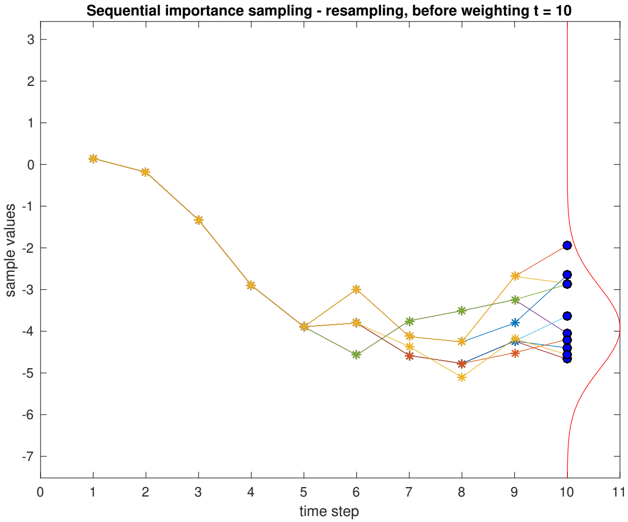
Particle filter



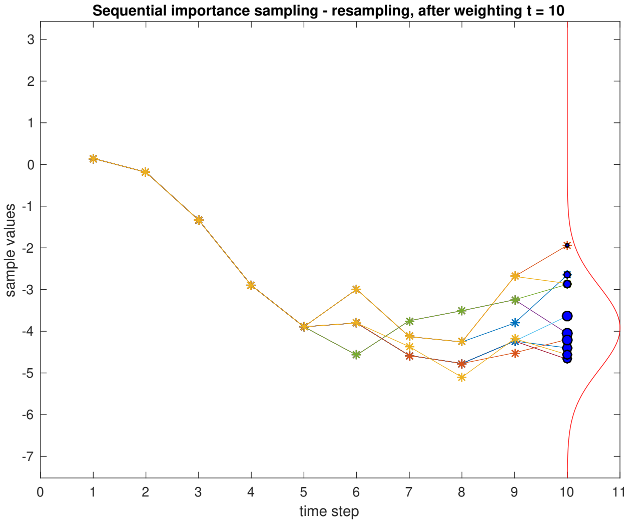
Particle filter



Particle filter



Particle filter



Thank you!