STK-4051/9051 Computational Statistics Spring 2022
Sequential Monte Carlo (some retake)

Instructor: Odd Kolbjørnsen, oddkol@math.uio.no

Target: $f(x_i)$
Get sample from: $g(x_i)$
Weigth: $w_i = f(x_i)/g(x_i)$

Last time:

- Properly weighted sample
  - normalized weights work
- Effective sample size
  - indication of quality of sample
- Sampling importance resampling (SIR)
  - Resampling proportional to w_i => new weigths: $\propto 1$
  - Resampling proportional to u_i => new weigths: $\propto w_i/u_i$

- Sampling vs weigthing vs resampling

Resampling more generally

- Assume sample and weigths $(x_i, w_i), i = 1, \ldots n$
- When we resample m samples we might get x_i several times
- If we resample $\tilde{x}_j \sim p(x_i) \propto w_i$
  - On average: $E\{\#\text{times } x_i \text{ is sampled}\} = m \cdot w_i$
  - The weigth of sample $\tilde{x}_j \propto 1$

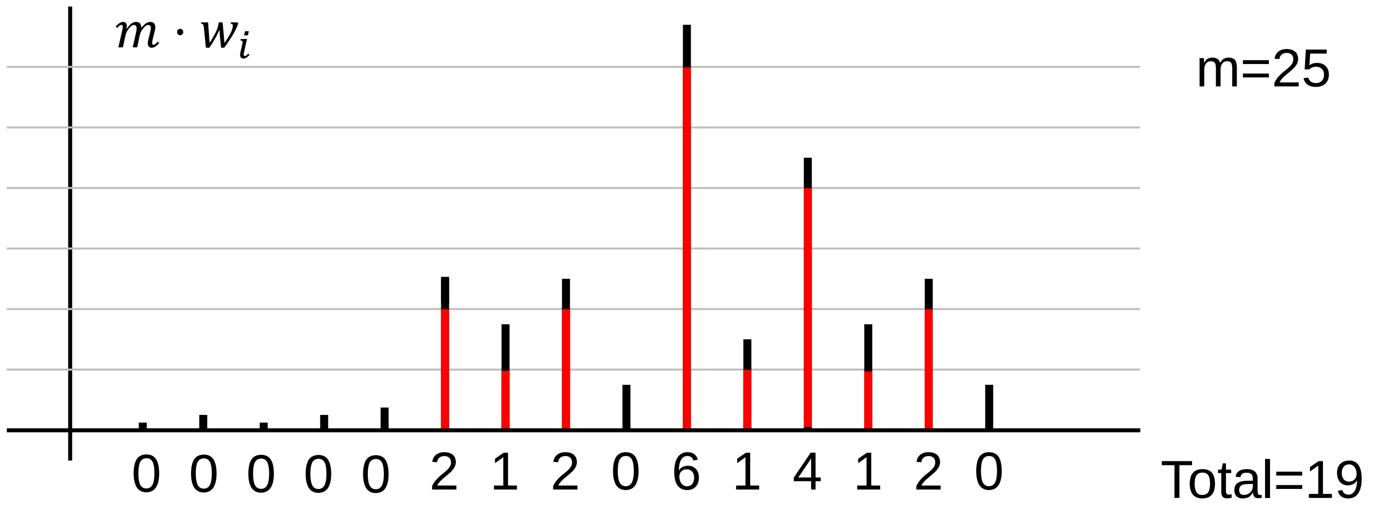
Point 1) If we resample $\tilde{x}_j \sim q(x_i) \propto u_i$,
then weigth of sample $\tilde{x}_j \propto w_k/u_k$ where $\{k\colon \tilde{x}_j = x_k\}$

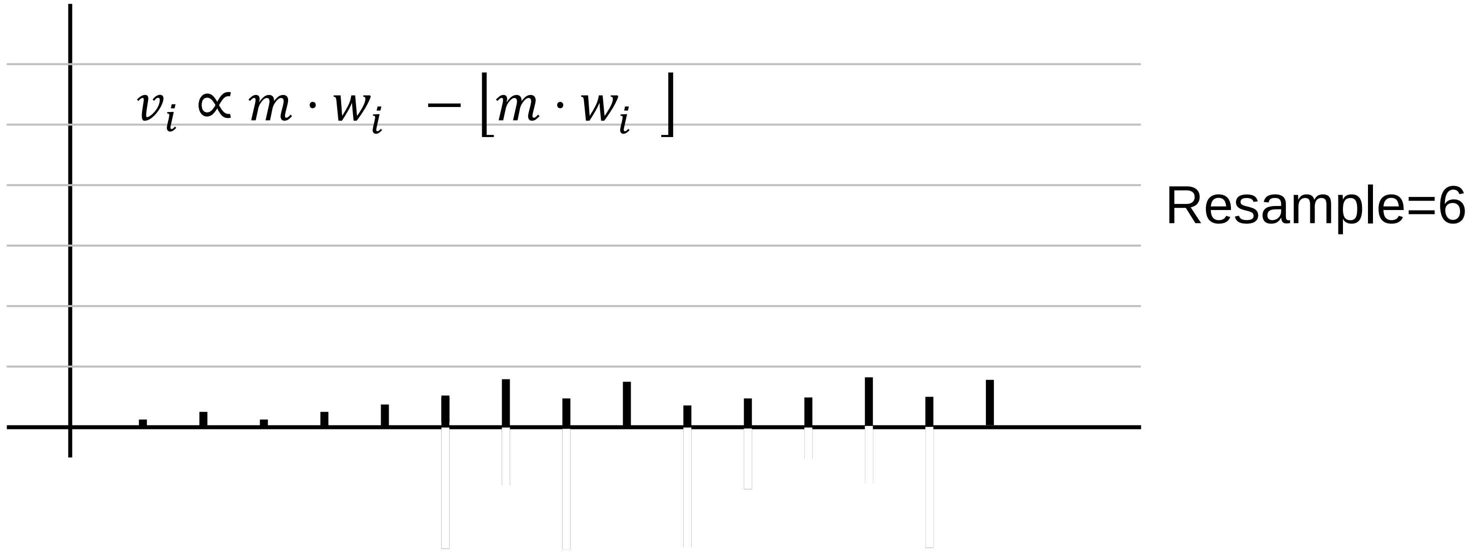
Point 2) We want to minimize the variability due to randomness

- If $m \cdot w_i$ is larger than an integer (say n_i) we can include x_i at least n_i times and leave the rest to randomness
- $n_i = \lfloor m \cdot w_i \rfloor$

Illustration of optimal resampling

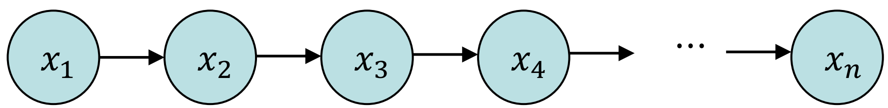
(resample weigth: $1/N$)





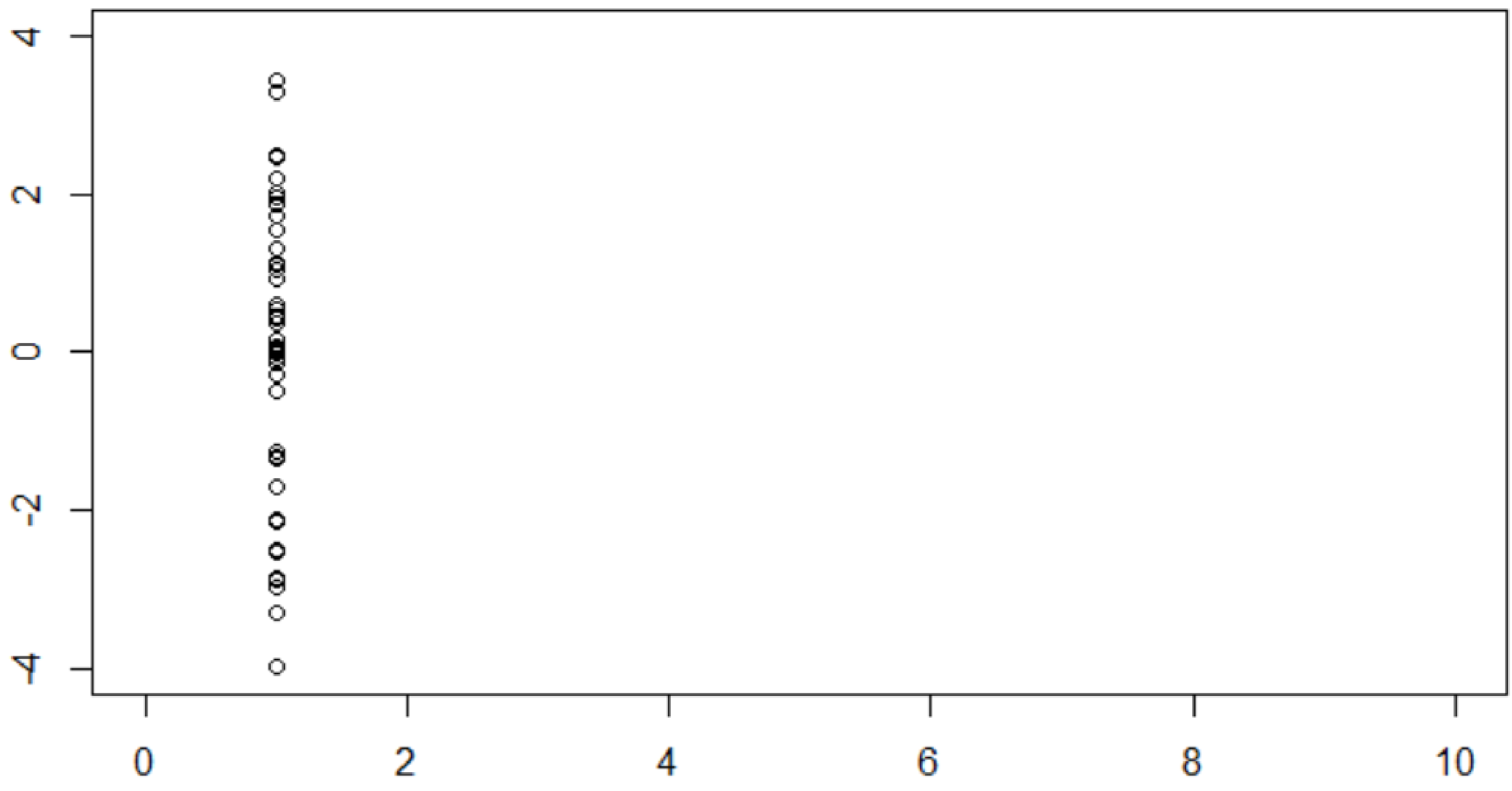
Recap

- Sequential Mote Carlo $x_t \sim f(x_t|x_{t-1})$,
  - $f(\boldsymbol{x}) = f(x_1)f(x_2|x_1)f(x_3|x_2)\cdots f(x_n|x_{n-1})$
  - $g(\boldsymbol{x}) = g(x_1)g(x_2|x_1)g(x_3|x_2)\cdots g(x_n|x_{n-1})$



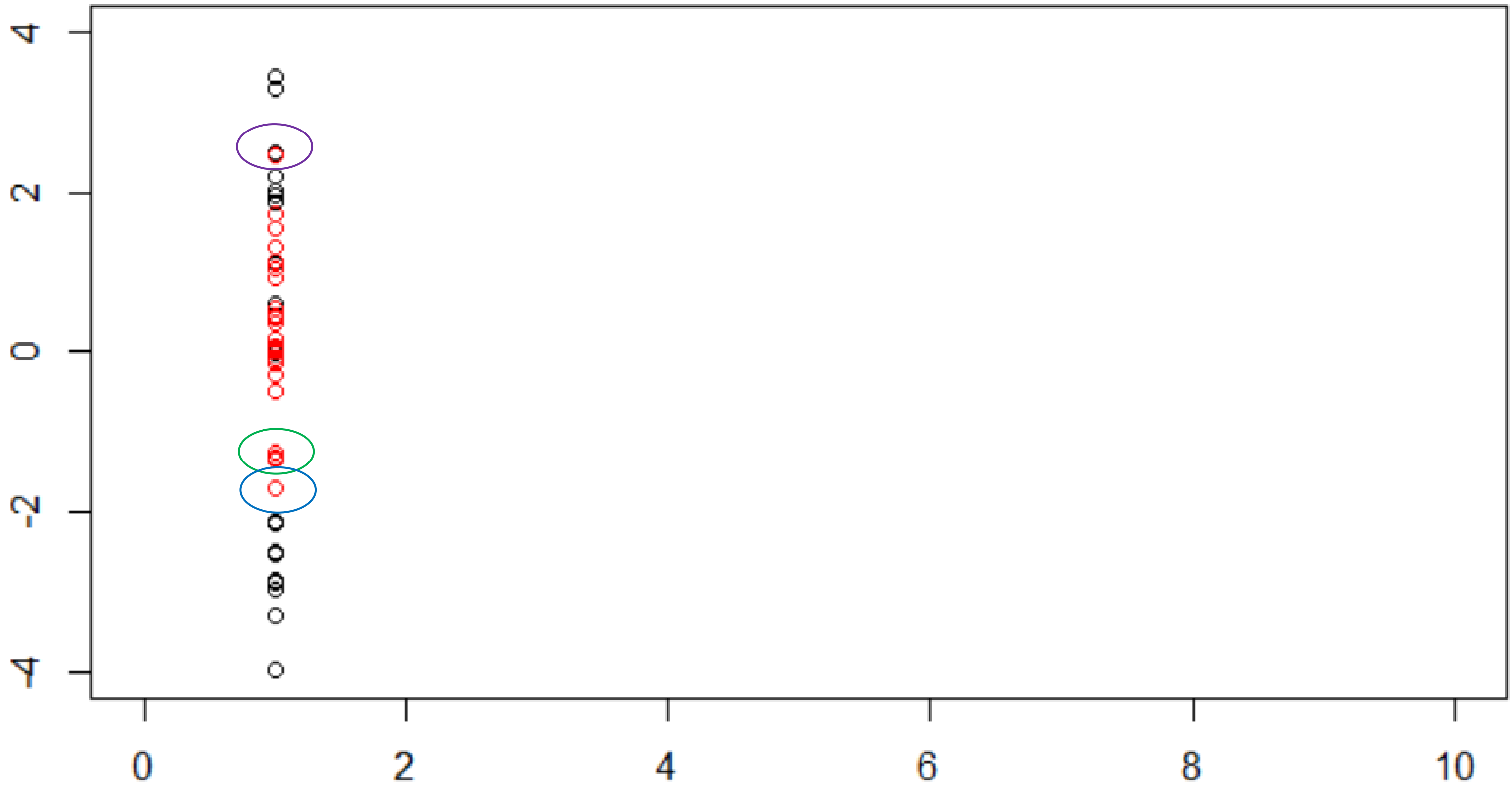
  - $x_t^i \sim g\left(x_t^i \middle| x_{t-1}^i\right)$, $w_t^i = w_{t-1}^i \dfrac{f\left(x_t^i \middle| x_{t-1}^i\right)}{g\left(x_t^i \middle| x_{t-1}^i\right)}$
  - Problem: weigth accumulation, $N_{\text{eff}} \to 1$
    - $Var\left(w_t^i\right) \geq Var(w_{t-1}^i)$
  - Partial solution: Resampling
- This is where I think I lost you last time…

Proposal = sample from $g(x_1)$



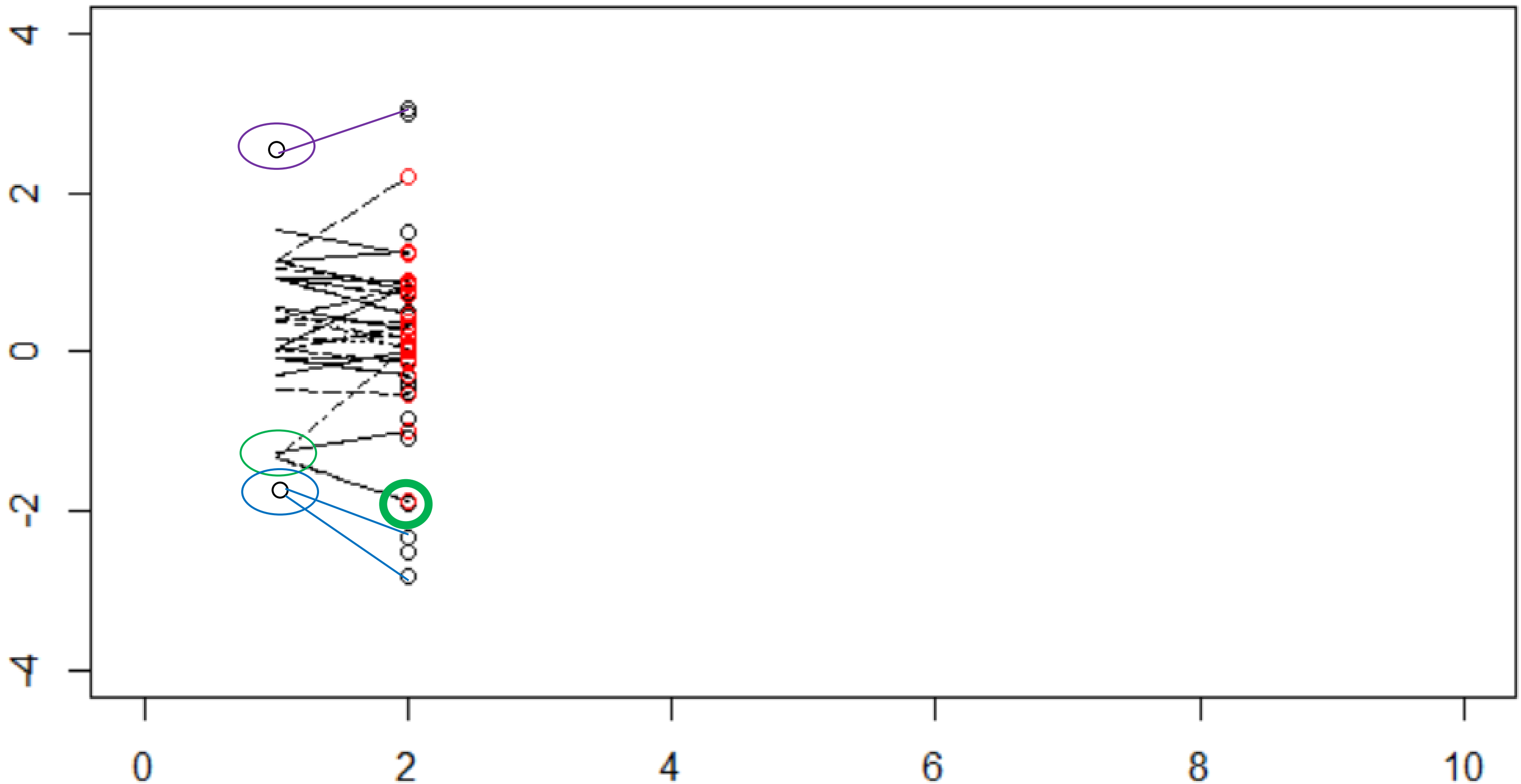
Compute weigths and resample =red

Red is a good approximation to $f(x_1)$



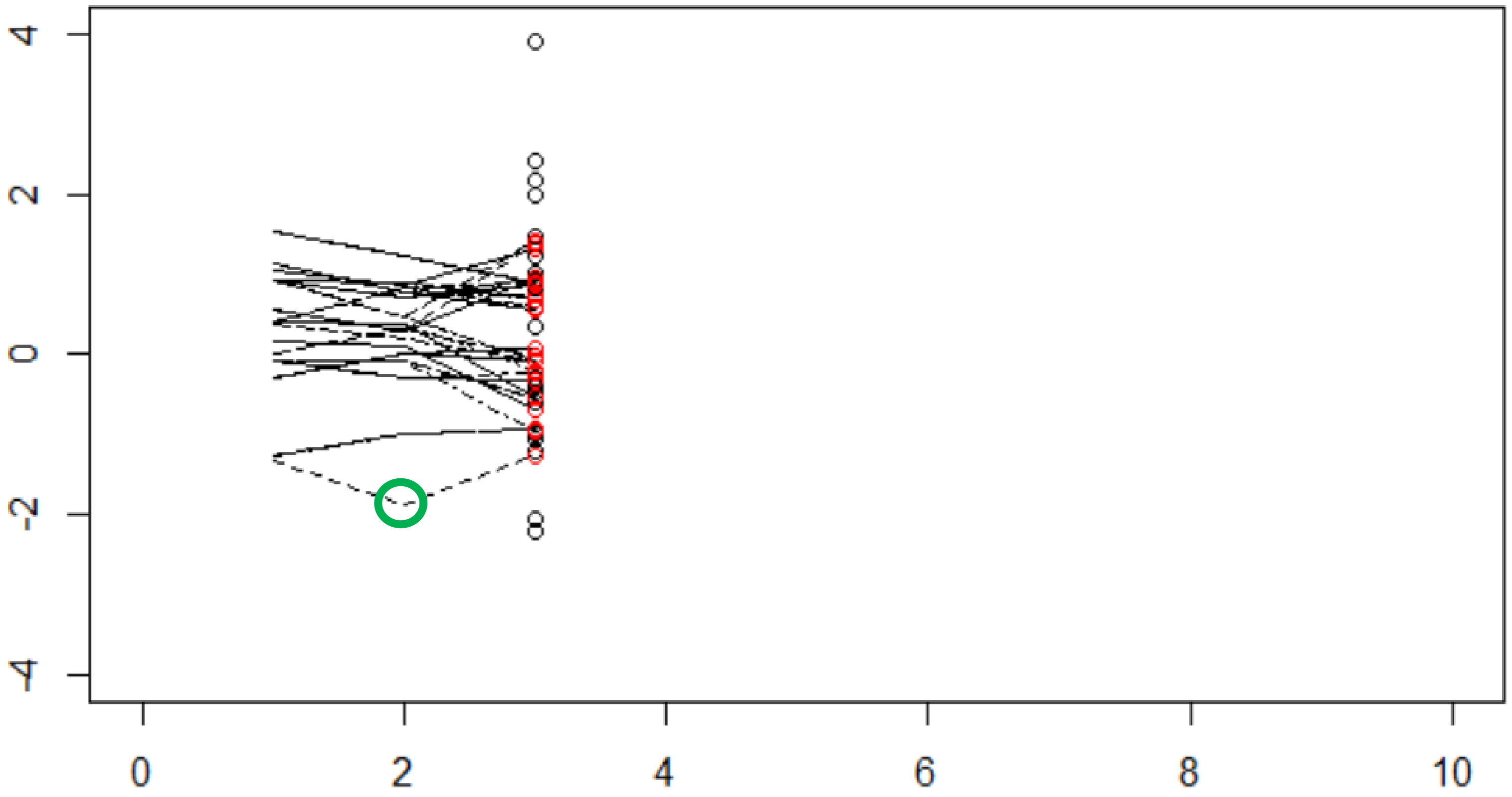
Step 2

Red is a good approximation to $p(x_2)$



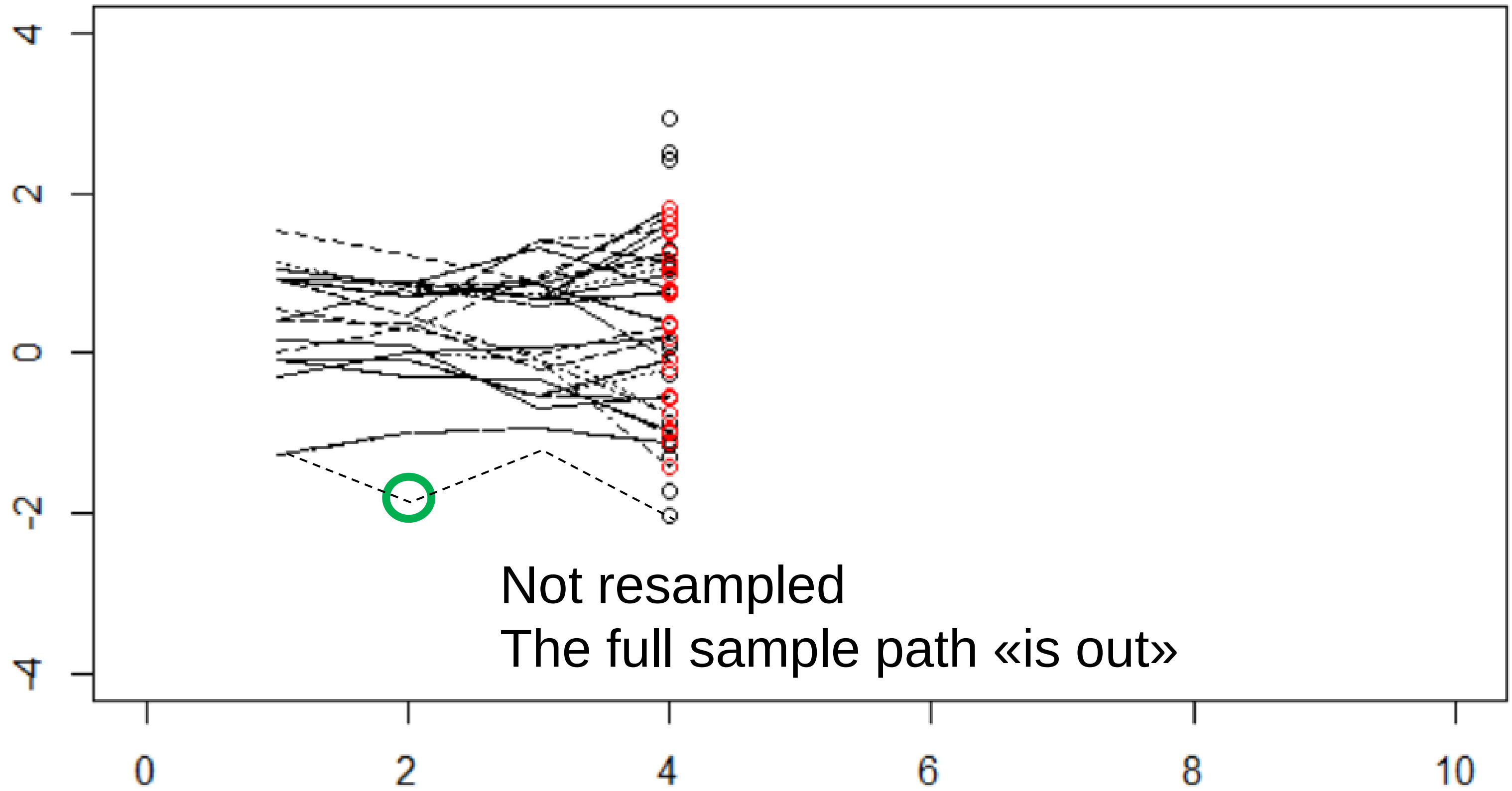
Step 3

Red is a good approximation to $p(x_3)$



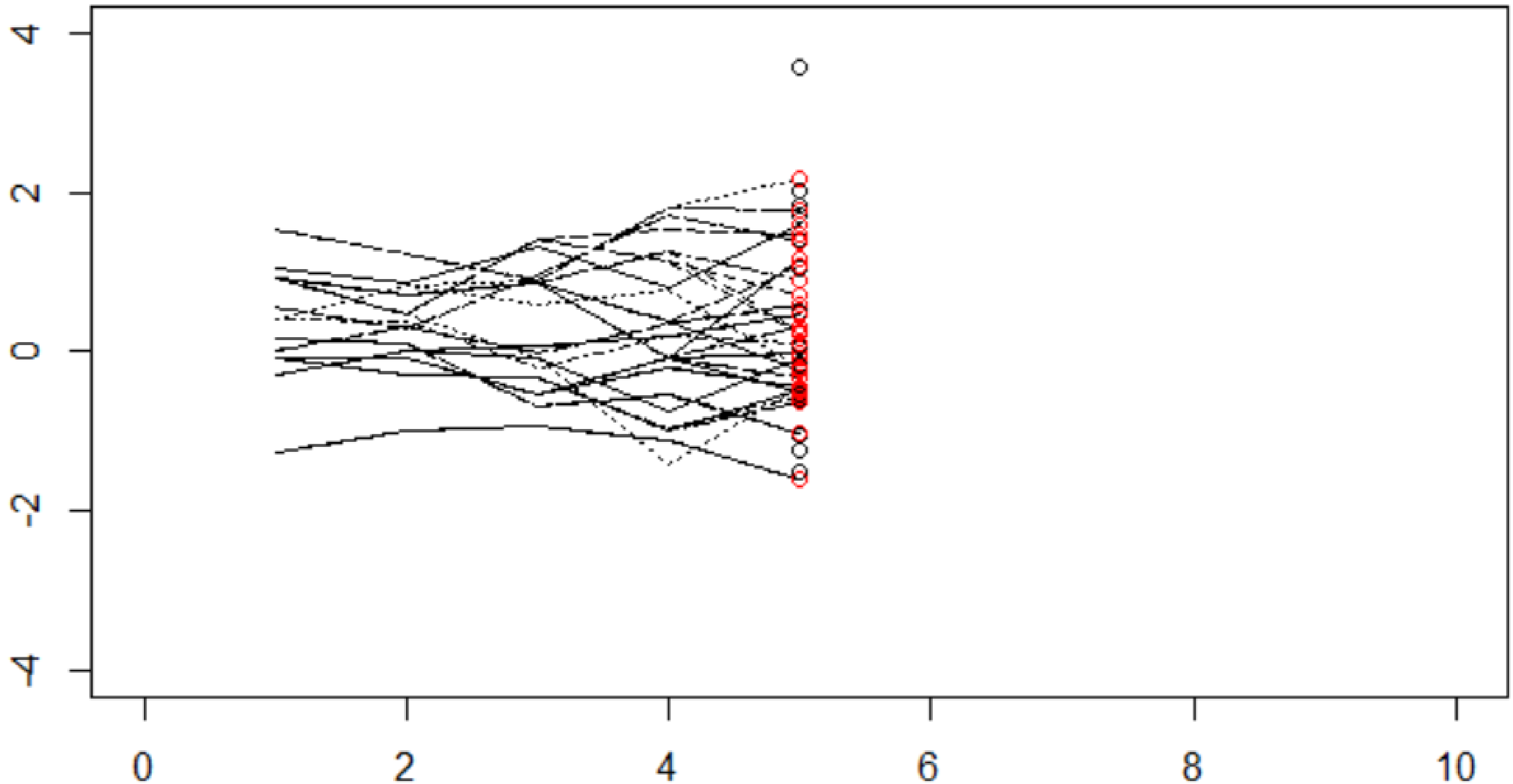
Step 4

Red is a good approximation to $p(x_4)$



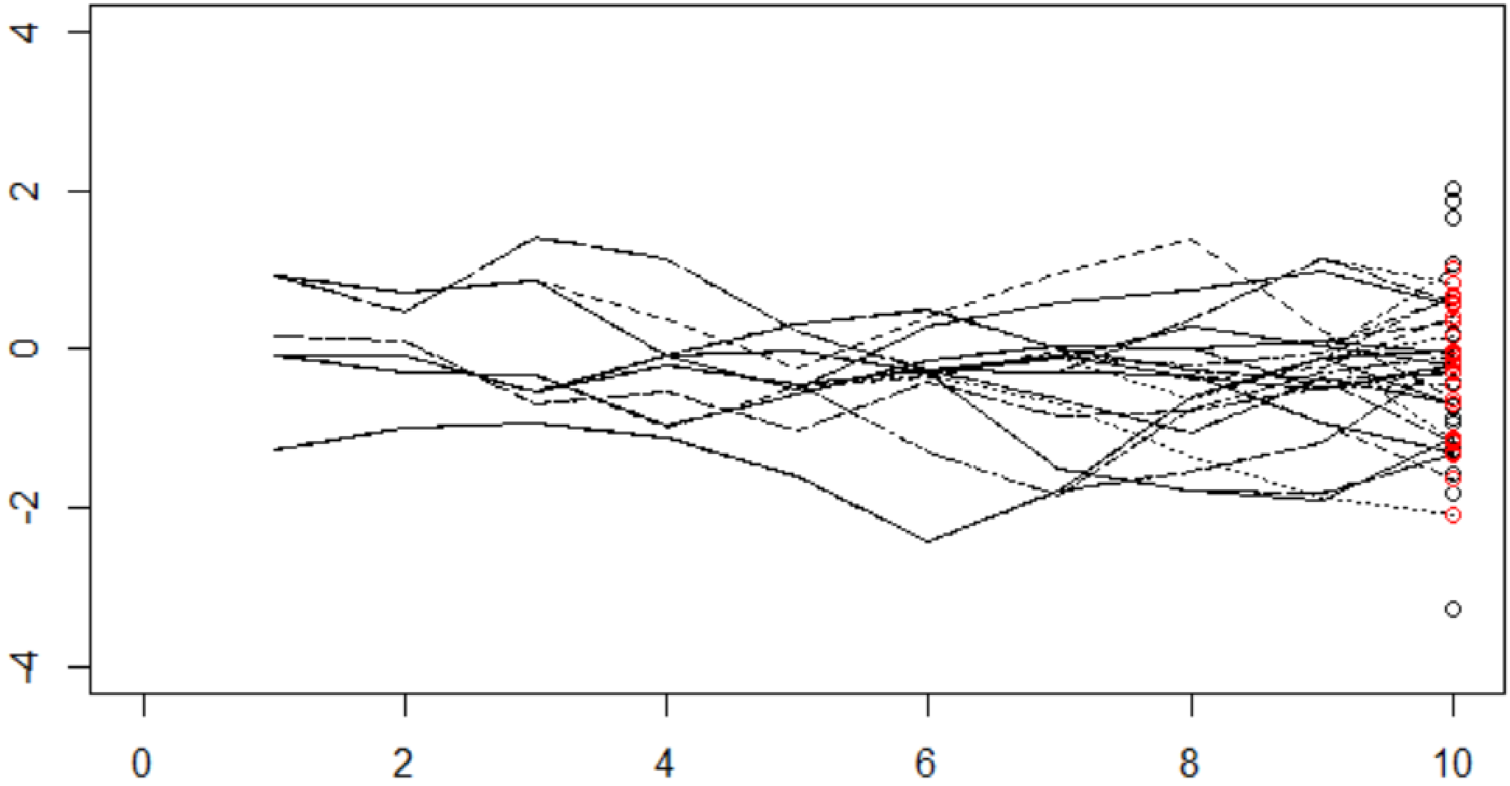
Step 5

Red is a good approximation to $p(x_5)$

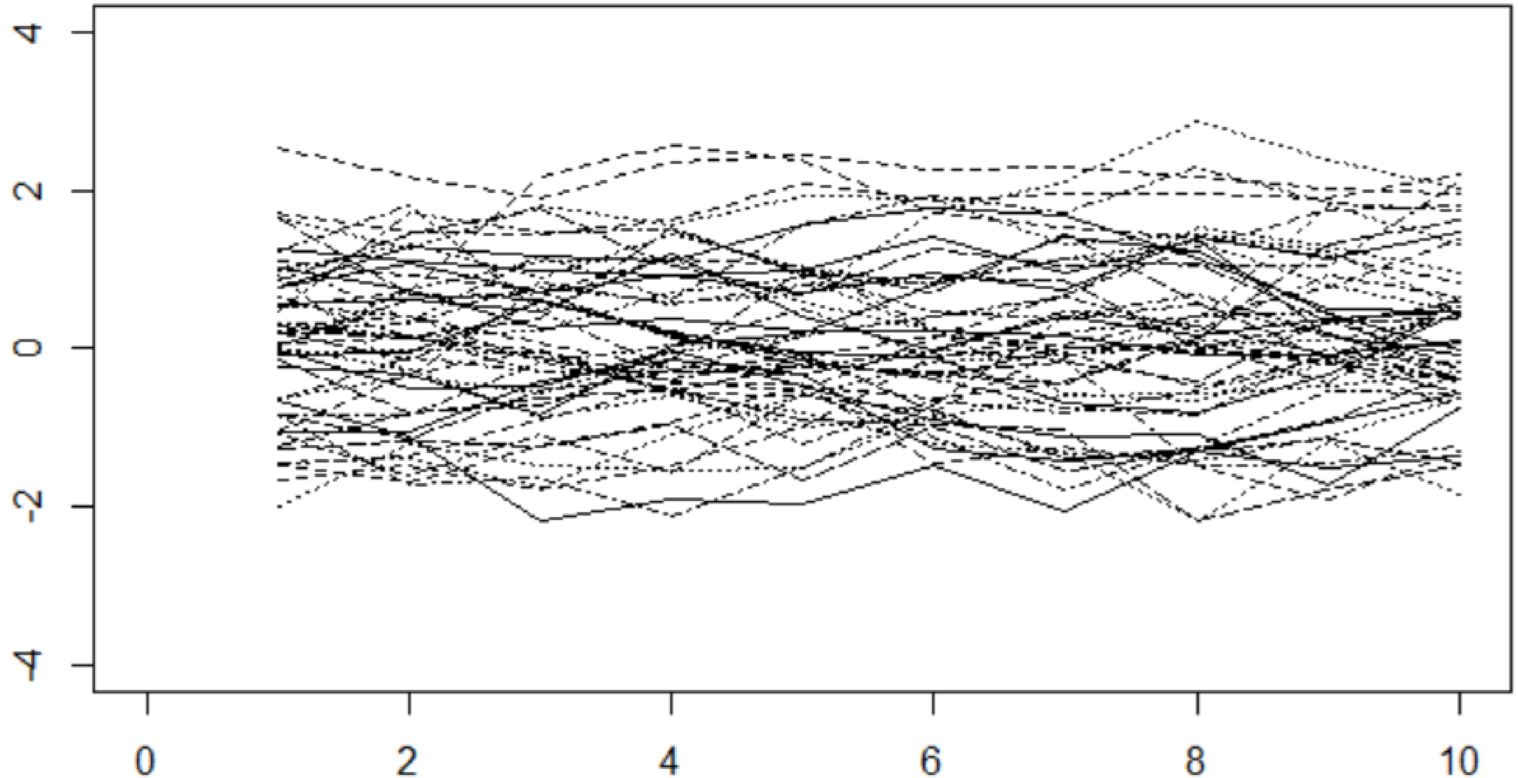


50 samples using resampling

Red is a good approximation to $p(x_{10})$



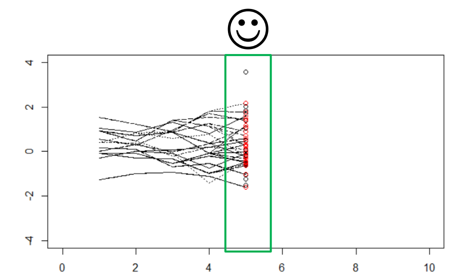
How 50 samples should look



When is what a good approximation?

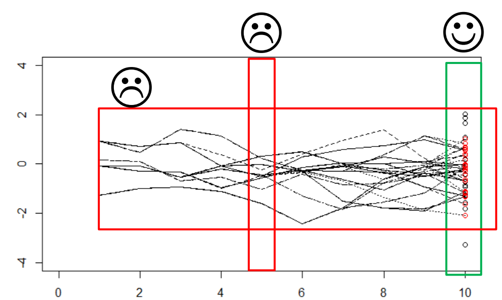
Sequential Monte Carlo particle filter:

- Sampling along the way is a good approximation for the marginal distribution at each step, $f(x_t)$

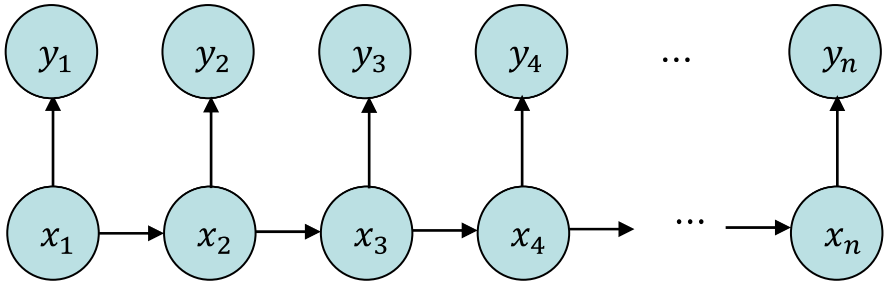


- At the end you can trace back the sample path for the final sample, this could have been an approximation for a sample from full distribution, but it is degenerated. Few samples in the beginning of the path. $f(x_1, \ldots, x_t, \ldots, x_n)$ is not good (as seen)

- If you try to use the final sample to get $f(x_t)$, this is in general not a good sample



- Sequential Monte Carlo for Hidden Markov Model
  - Want to sample from $f(x_t) \propto p(x_t^i | x_{t-1}^i)p(y_t | x_t^i)$



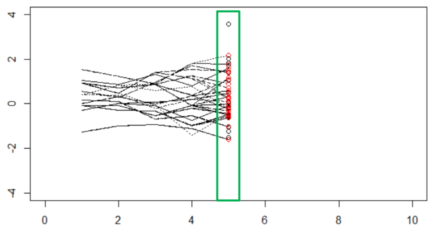
  - Bootstrap filter $x_t^i \sim g(x_t | x_{t-1}^i)$, $w_t^i = w_{t-1}^i p(y_t | x_t^i)$

  - General $x_t^i \sim g\left(x_t | x_{t-1}^i, y_n\right)$, $w_t^i = w_{t-1}^i \frac{p(x_t^i | x_{t-1}^i)p(y_t | x_t^i)}{g\left(x_t^i | x_{t-1}^i, y_n\right)}$

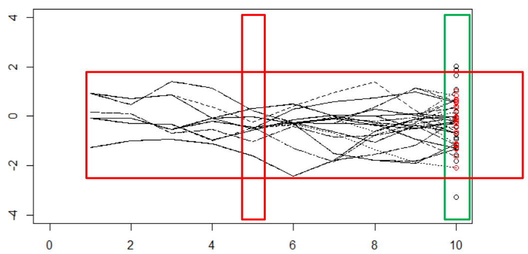
- Same problems as before:
  - Weigth accumulation, $N_{\text{eff}} \rightarrow 1$
  - Resampling → Degeneracy

When is what a good approximation?

Sequential Monte Carlo bootstrap filter:



- Sampling along the way is a good approximation for the marginal distribution at each step, $f(x_t|y_{1:t})$
- At the end you can trace back the sample path for the final sample, this could have been an approximation for a sample from full distribution, but it is degenerated. Few samples in the beginning of the path. $f(x_{1:n}|y_{1:n})$ is not good (as seen)

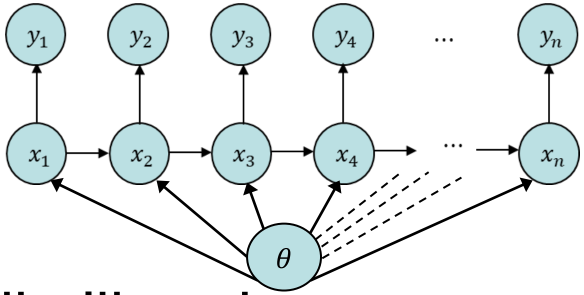


- If you try to use the final sample to get $f(x_t|y_{1:n})$, this is not a good sample
- So you solve the filter problem, $f(x_t|y_{1:t})$ **not** the smoothing problem, $f(x_t|y_{1:n})$ **not** the full conditioning problem $f(x_{1:n}|y_{1:n})$

The difference is which data you condition to

Today

- Inference in Sequential Markov models



- Maximum likelihood
  - Likelihood by sampling (bootstrap filter)

$\theta = ?$

$$p(\mathbf{y}_{1:t}|\theta) = p(y_1|\theta)\prod_{s=2}^{t} \underbrace{p(y_s|\mathbf{y}_{1:s-1};\theta)} \xrightarrow{s=2,\dots,t} \approx \sum_{i=1}^{N} w_{t-1}^{i} p(y_s|x_s^i;\theta)$$

- Bayesian approach
  - Direct approach
  - Dynamic approach
  - Sufficient statistics

$p(\theta)$ prior
Want $p(\theta|y_{1:n})$
(posterior)

SMC and maximum likelihood

- Interested in maximizing

$$L_t(\theta) = p(\mathbf{y}_{1:t}|\theta) = \int_{\mathbf{x}_{1:t}} p(\mathbf{y}_{1:t}|\mathbf{x}_{1:t};\theta)p(\mathbf{x}_{1:t}|\theta)d\mathbf{x}_{1:t}.$$

- Main problem: Calculation of the likelihood function (and possibly the score function in order to do optimization)
- Main approach: Use that

$$p(\mathbf{y}_{1:t}|\theta) = p(y_1|\theta)\prod_{s=2}^{t} p(y_s|\mathbf{y}_{1:s-1};\theta)$$

and

$$\begin{aligned} p(y_s|\mathbf{y}_{1:s-1}) &= \int_{x_s} p(x_s|\mathbf{y}_{1:s-1})p(y_s|x_s;\theta)dx_s \\ &\approx \sum_{i=1}^{N} w_{t-1}^i p(y_s|x_s^i;\theta) \end{aligned}$$

where $x_s^i \sim p(x_s|x_{s-1}^i)$ (Bootstrap filter).

- Poyiadjis et al. (2011): Algorithms for calculating the score function and information (matrix) recursively

Maximum likelihood in SMC

To evaluate for a given θ

$$p(\mathbf{y}_{1:t}|\theta) = p(y_1|\theta)\prod_{s=2}^{t} p(y_s|\mathbf{y}_{1:s-1};\theta)$$

$$p(y_s|y_{1:s-1},\theta) = \int_{x_s} p(x_s|y_{1:s-1},\theta)p(y_s|x_s,\theta)dx_s$$

$$= \int_{x_s}\int_{x_{s-1}} p(x_s,x_{s-1}|y_{1:s-1},\theta)p(y_s|x_s,\theta)dx_{s-1}dx_s$$

$$= \int_{x_s}\int_{x_{s-1}} p(x_s|x_{s-1},\theta)\underbrace{p(x_{s-1}|y_{1:s-1},\theta)}p(y_s|x_s,\theta)dx_{s-1}dx_s$$

We sample this
To get x_s^i

We have this for $s-1$ with:
(x_{s-1}^i, w_{s-1}^i), $i = 1,\dots,M$

We get the weight
update for x_s^i
$u_s^i = p\left(y_s \middle| x_s^i, \theta\right)$

Sample for s: $\boxed{(x_s^i, u_s^i w_{s-1}^i),\ i = 1,\dots,M}$

SMC and Bayesian parameter estimation

- Assume

$$
\begin{aligned}
X_1 &\sim p(x_1;\theta)\\
X_t &\sim p(x_t|x_{t-1};\theta)\\
Y_t &\sim p(y_t|x_t;\theta)\\
\theta &\sim p(\theta)
\end{aligned}
$$

- Aim now: Simulate from $p(x_t,\theta|\mathbf{y}_{1:t})$
- Three approaches
  - Direct use of SMC
  - Introducing dynamics in θ
  - Using sufficient statistics

Direct use of SMC

- Assume at time $t-1$ the existence of a properly weighted sample $\{(x_{t-1}^i, \theta^i, w_{t-1}^i)\}$ with respect to $p(x_{t-1}, \theta|\mathbf{y}_{1:t-1})$.
- We have

$$p(x_t, \theta|\mathbf{y}_{1:t-1}) = \int_{x_{t-1}} p(x_t|x_{t-1}, \theta)p(x_{t-1}, \theta|\mathbf{y}_{1:t-1})dx_{t-1}$$
$$\approx \sum_{i=1}^N w_{t-1}^i p(x_t|x_{t-1}^i, \theta^i)\delta_\theta(\theta^i)$$

and

$$p(x_t, \theta|\mathbf{y}_{1:t}) \approx c \cdot \sum_{i=1}^N w_{t-1}^i p(x_t|x_{t-1}^i, \theta^i)\delta_\theta(\theta^i)p(y_t|x_t, \theta^i)$$

- Updated samples $\{(\theta^i, x_t^i, w_t^i)\}$:
  1. Simulate $x_t^i \sim p(x_t|x_{t-1}^i, \theta^i)$
  2. Update the weights by $w_t^i \propto w_{t-1}^i p(y_t|x_t^i, \theta^i)$

Is direct use of SMC properly weighted?

- Proposal:

$$\theta^i \sim g(\theta) \qquad x_s^i \sim p(x_s|x_{s-1}^i, \theta^i), \quad s = 1, ..., t$$

- Weights at time $t = 1$:

$$w_1^i = \frac{p(\theta^i)p(x_1^i|\theta^i)p(y_1|x_1^i, \theta^i)}{g(\theta^i)p(x_1^i|\theta^i)} = \frac{p(\theta^i)p(y_1|x_1^i, \theta^i)}{g(\theta^i)}$$

giving properly weighted samples at time 1.

- At time t:

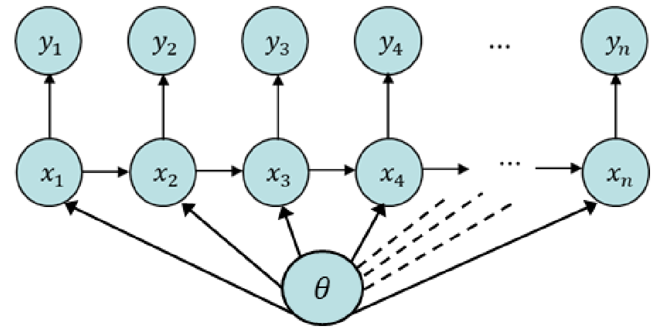
$$\begin{aligned}
w_t^i =& \frac{p(\theta^i)p(x_1^i|\theta^i)p(y_1|x_1^i, \theta^i) \prod_{s=2}^t p(x_s^i|x_{s-1}^i, \theta^i)p(y_s|x_s^i, \theta^i)}{g(\theta^i)p(x_1^i|\theta^i) \ \prod_{s=2}^t p(x_s^i|x_{s-1}^i, \theta^i)} \\
=& \frac{p(\theta^i)p(x_1^i|\theta^i)p(y_1|x_1^i, \theta^i) \prod_{s=2}^t p(y_s|x_s^i, \theta^i)}{g(\theta^i)p(x_1^i|\theta^i)} \\
=& \frac{p(\theta^i)p(x_1^i|\theta^i)p(y_1|x_1^i, \theta^i) \prod_{s=2}^{t-1} p(y_s|x_s^i, \theta^i)}{g(\theta^i)p(x_1^i|\theta^i)} p(y_t|x_t^i, \theta^i) \\
=& w_{t-1}^i p(y_t|x_t{}^i, \theta^i)
\end{aligned}$$

- Main problem: Now we need to resample $(\theta, \boldsymbol{x}_{1:t})$.
  Will result in degeneracy when $p(\theta, x_t|\boldsymbol{y}_{1:t})$ is of interest.

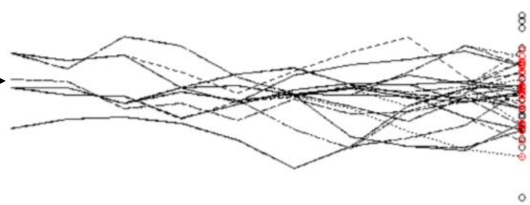
Issue: Direct approach

Direct approach samples theta initially and
is never able to resample it thus at each resampling
it is a thinning/depletion of unique elements

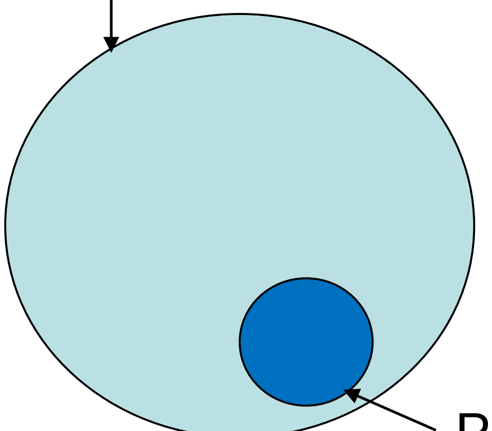
$$p(x_t, \theta|\mathbf{y}_{1:t}) \approx c \cdot \sum_{i=1}^{N} w_{t-1}^{i} p(x_t|x_{t-1}^{i}, \theta^{i})\delta_{\theta}(\theta^{i})p(y_t|x_t, \theta^{i})$$



θ is sampled
here

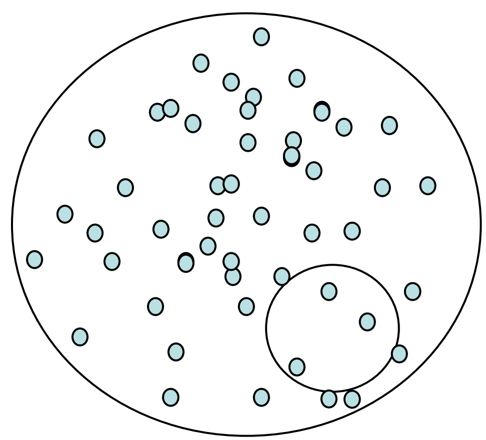


Prior



Posterior

Sample from prior



Lemmings data

- Interested in the dynamics of the lemmings populations
- From church books: Binary records on lemmings years or not.
- Define $x_t = \log(N_t)$, N_t population size at year t
- Model

$$\begin{aligned} x_t =& a x_{t-1} + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma^2) \\ y_t \sim& \text{Binom}\left(1, \tfrac{\exp(x_t)}{1+\exp(x_t)}\right) \end{aligned}$$

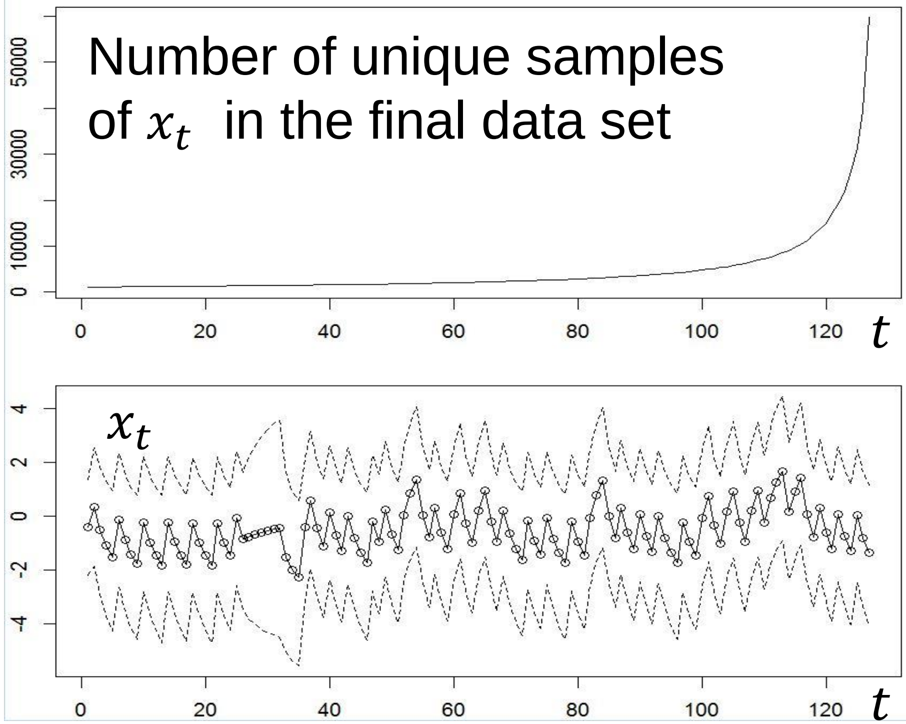
- Of interest: $p(x_t|\boldsymbol{y}_{1:t})$, $p(a|\boldsymbol{y}_{1:t})$
- `SMC_lin_bin.R`, `SMC_lin_bin_fixed.R`
- `SMC_lin_bin_parest_direct.R`

```
#SMC with parameter estimation
#Estimating parameters simultaneously using direct method
sig2=sig^2
sig2.a = 2
#Initialization
x.sim[1,]=rnorm(N,0,sig)
w = dbinom(y[1],1,exp(x.sim[1,])/(1+exp(x.sim[1,])))
#Resample
ind = sample(1:N,N,replace=T,prob=w)
x.sim[1,] = x.sim[1,ind]
w = rep(1/N,N)
x.hat[1,1] = mean(x.sim[1,])
x.hat[1,2:3] = quantile(x.sim[1,],c(0.025,0.975))
a.sim = rnorm(N,0,sqrt(sig2.a))
a.hat = matrix(nrow=nT,ncol=3)
a.hat[1,1] = mean(a.sim)
a.hat[1,2:3] = quantile(a.sim,c(0.025,0.975))
```

```
N.unique = rep(NA,nT)
for(i in 1:nT)
  N.unique[i] = length(unique(x.sim[i,]))
```

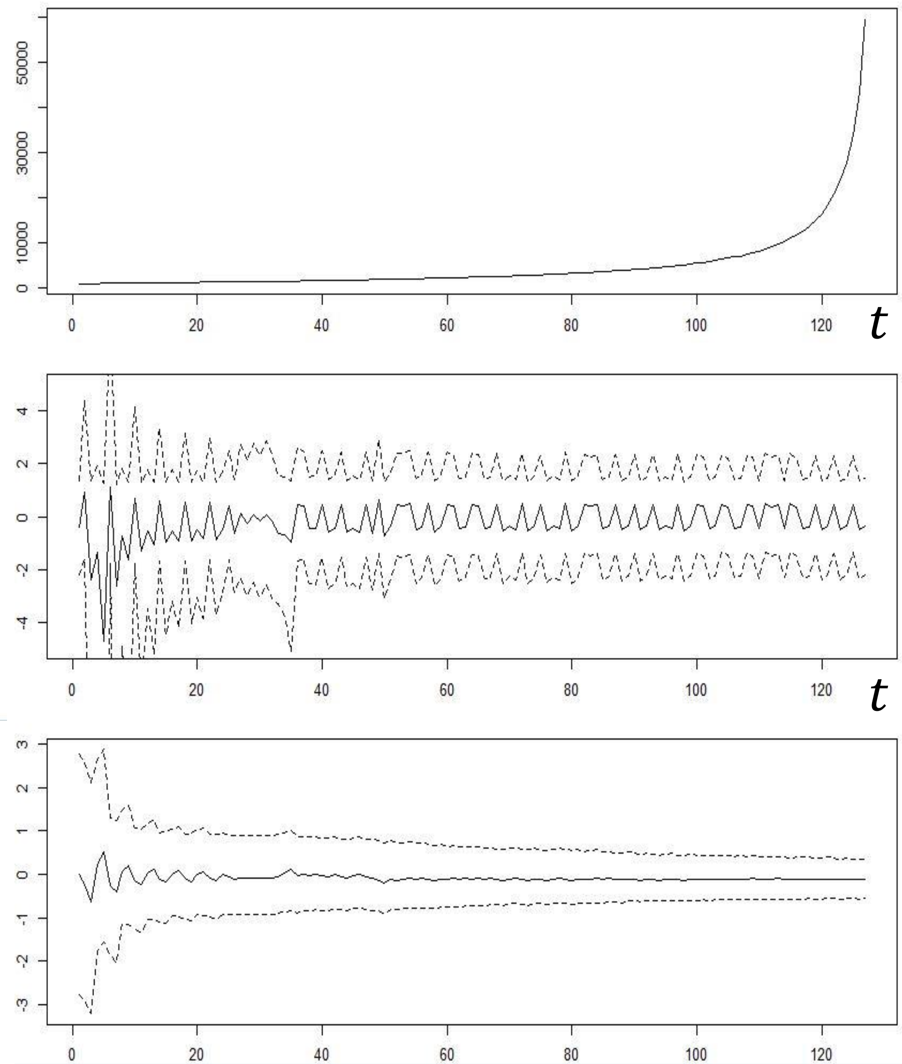
```
for(i in 2:nT)
{
  x.sim[i,]=rnorm(N,a.sim*x.sim[i-1,],sig)
  if(!is.na(y[i]))
    w = w*dbinom(y[i],1,exp(x.sim[i,])/(1+exp(x.sim[i,])))
  #Resample
  ind = sample(1:N,N,replace=T,prob=w)
  x.sim[1:i,] = x.sim[1:i,ind]          #Note: Resampling the whole path!
  a.sim = a.sim[ind]
  w = rep(1/N,N)
  x.hat[i,1] = mean(x.sim[i,])
  x.hat[i,2:3] = quantile(x.sim[i,],c(0.025,0.975))
  a.hat[i,1] = mean(a.sim)
  a.hat[i,2:3] = quantile(a.sim,c(0.025,0.975))
}
```

SMC_lin_bin_fixed.R



a=0.9

- SMC_lin_bin_parest_direct.R



Introducing dynamics in θ

- Liu and West (2001): Assume θ is (slowly) changing with time:

$$\theta_t = \theta_{t-1} + \zeta_t, \quad \zeta_t \sim N(0, q)$$

- Focus on $p(x_t, \theta_t | \mathbf{y}_{1:t})$.
- Assume a weighted sample $\{(x_{t-1}^i, \theta_{t-1}^i, w_{t-1}^i)\}$

$$\begin{aligned} p(x_t, \theta_t | \mathbf{y}_{1:t-1}) &= \int_{x_{t-1}} p(x_t | x_{t-1}, \theta_t) p(\theta_t | \theta_{t-1}) p(x_{t-1}, \theta_{t-1} | \mathbf{y}_{1:t-1}) dx_{t-1} d\theta_{t-1} \\ &\approx \sum_{i=1}^{N} w_{t-1}^i p(x_t | x_{t-1}^i, \theta_t) p(\theta_t | \theta_{t-1}^i) \end{aligned}$$

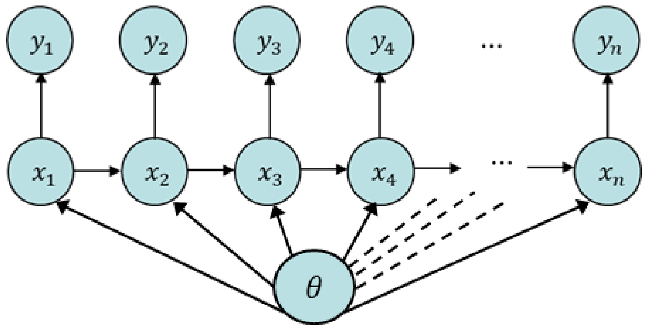
$$p(x_t, \theta_t | \mathbf{y}_{1:t}) \approx c \cdot \sum_{i=1}^{N} w_{t-1}^i p(x_t | x_{t-1}^i, \theta_t) p(\theta_t | \theta_{t-1}^i) p(y_t | x_t, \theta_t).$$

- Update samples to$\{(\theta_t^i, x_t^i, w_t^i)\}$ by
  1. Simulate $\theta_t^i \sim p(\theta_t | \theta_{t-1}^i)$,
  2. Simulate $x_t^i \sim p(x_t | x_{t-1}^i, \theta_t^i)$
  3. Update the weights by $w_t^i \propto w_{t-1}^i p(y_t | x_t^i, \theta_t^i)$.

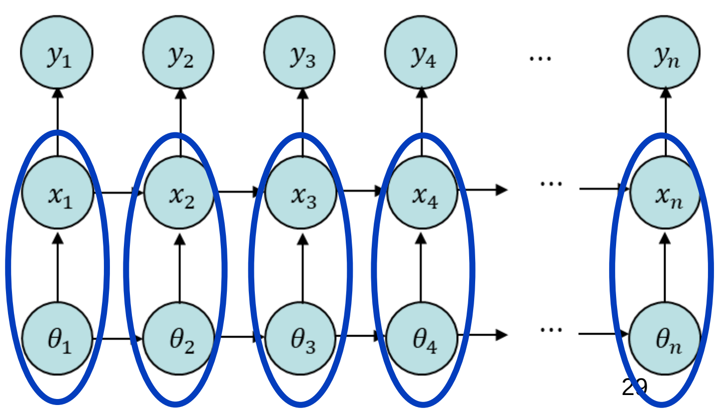
«θ is like x»

Issue: Dynamic approach

Dynamic approach. We can resample θ_t at each time step
This solves the problem of degeneration
(but we solve a different problem…)



Original problem



New problem

This problem is equivalet with the «no θ problem»

$$\tilde{x}_t = \underbrace{(x_t, \theta_t)}_{\boxed{\text{«Dynamic duo»}}}$$

Dynamics in θ continued

- New values $\{\theta_t^i\}$ are generated at each time point
- Main problem: Introduce extra variability in θ_t.
- Consequence: Estimation of θ_t mainly based on most recent observations
- The model

$$\theta_t = \theta_{t-1} + \zeta_t, \quad \zeta_t \sim N(0, q)$$

  might be reasonable
- New problem: Estimate the static parameter q.
- `SMC_lin_bin_parest_dyn.R`

Sufficient statistics

- Example:

$$x_t = ax_{t-1} + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma^2), \quad \sigma \text{ known for simplicity}$$

- The distribution $p(y_t|x_t)$ can be arbitrary (but not depending on θ).
- $\theta = a$ needs to be estimated. Assume a prior $a \sim N(\mu_a, \sigma_a^2)$.
- Can be shown:

$$p(a|\mathbf{x}_{1:t}) = N(\mu_{a|t}, \sigma_{a|t}^2)$$

where

$$\mu_{a|t} = \frac{\sigma_a^2 \sum_{s=2}^t x_s x_{s-1} + \sigma^2 \mu_a}{\sigma_a^2 \sum_{s=2}^t x_{s-1}^2 + \sigma^2}; \quad \sigma_{a|t}^2 = \frac{\sigma^2 \sigma_a^2}{\sigma_a^2 \sum_{s=2}^t x_{s-1}^2 + \sigma^2}.$$

- Main point: Given $\mathbf{x}_{1:t}$, the distribution of a (and simulation) is simple.
- $p(a|\mathbf{x}_{1:t})$ only depend on $S_{t,1} = \sum_{s=2}^t x_s x_{s-1}$ and $S_{t,2} = \sum_{s=2}^t x_{s-1}^2$
- Both terms can be recursively updated through

$$S_{t,1} = S_{t-1,1} + x_t x_{t-1}, \quad S_{t,2} = S_{t-1,2} + x_{t-1}^2.$$

SMC and sufficient statistics

- Assume $p(y_t|x_t)$ do not depend on θ.
- Assume $p(\theta|\mathbf{x}_{1:t}) = p(\theta|S_t)$, S_t sufficient statistic.
- Assume $S_t = h(S_{t-1}, x_{t-1}, x_t)$
- Fearnhead (2002) and Storvik (2002): Focus on $p(x_t, S_t|\mathbf{y}_{1:t})$, not $p(x_t, \theta|\mathbf{y}_{1:t})$.
- Assume a properly weighted sample $\{(x_{t-1}^i, S_{t-1}^i, w_{t-1}^i), i = 1, \ldots, N\}$ with respect to $p(x_{t-1}, S_{t-1}|\mathbf{y}_{1:t-1})$
- Similar recursions as before:

$$
\begin{aligned}
p(x_t, S_t|\mathbf{y}_{1:t-1}) =& \int_{x_{t-1}} p(x_t, S_t|x_{t-1}, S_{t-1})p(x_{t-1}, S_{t-1}|\mathbf{y}_{1:t-1})dx_{t-1}dS_{t-1} \\
\approx& \sum_{i=1}^N w_{t-1}^i p(x_t, S_t|x_{t-1}^i, S_{t-1}^i) \\
p(x_t, S_t|\mathbf{y}_{1:t}) \approx& c \cdot \sum_{i=1}^N w_{t-1}^i p(x_t, S_t|x_{t-1}^i, S_{t-1}^i)p(y_t|x_t).
\end{aligned}
$$

- Simulation from $p(x_t, S_t|x_{t-1}^i, S_{t-1}^i)$ (possible proposal function)
  1. Simulate $\theta^i \sim p(\theta|x_{t-1}^i, S_{t-1}^i) = p(\theta|S_{t-1}^i)$.
  2. Simulate $x_t^i \sim p(x_t|x_{t-1}^i, \theta^i)$.
  3. Put $S_t^i = h(S_{t-1}^i, x_{t-1}^i, x_t^i)$.

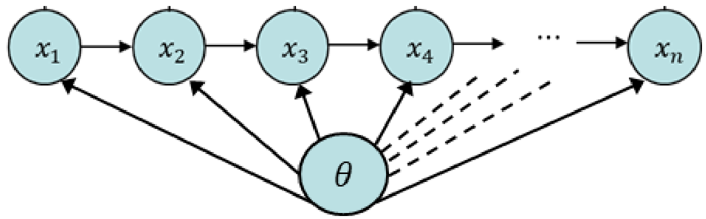
Algorithm Storvik filter

Algorithm 3 SMC with parameter updating

Simulate $\theta^i \sim p(\theta)$ for $i = 1, ..., N$. ▷ Initialization
Simulate $x_1^i \sim p(x_1|\theta^i)$ for $i = 1, ..., N$.
Put weights $w_1^i = p(y_1|x_1^i)$.
Put $S_1^i = 0$ for $i = 1, ..., N$.
for $t = 2, 3, \ldots$ **do** ▷ Sequential Monte Carlo
    Simulate $\theta^i \sim p(\theta|S_{t-1}^i)$ for $i = 1, ..., N$.
    Simulate $x_t^i \sim p(x_t|x_{t-1}^i, \theta^i)$ for $i = 1, ..., N$.
    Put weights $w_t^i = w_{t-1}^i p(y_t|x_t^i)$.
    Put $S_t^i = h(S_{t-1}^i, x_{t-1}^i, x_t^i)$.
    if $\hat{N}_{eff}$ is small **then** ▷ Resampling
        Resample (x_t^i, S_t^i) with probabilities proportional to w_t^i.
        Put $w_t^i = 1/N$.
    end if
end for

```
SMC_lin_bin_parest_suff.R
```

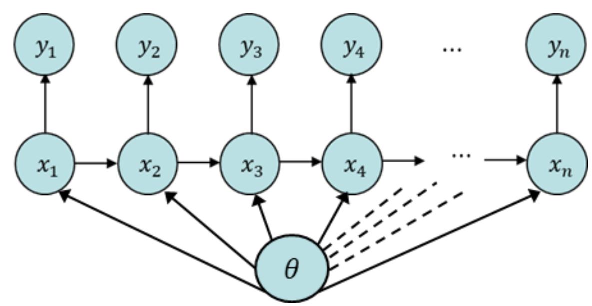
Approach using sufficient statistics



- We can update the statistics recursively
- We can compute sufficient statistics for θ

$$p(\theta|\mathbf{x}_{1:t}) = p(\theta|S_t),\ S_t \text{ sufficient statistic}$$

- This opens for resampling of theta ☺
- We can expand this to the hidden Markov model as long as the hidden variable is the one influenced by theta
- This keeps the original conditioning structure (as x_t “protect” θ from y_t)



SMC and sufficient statistics general

- Assume $p(y_t|x_t)$ do not depend on θ.
- Assume $p(\theta|\mathbf{x}_{1:t}) = p(\theta|S_t)$, S_t sufficient statistic.
- Assume $S_t = h(S_{t-1}, x_{t-1}, x_t)$
- Fearnhead (2002) and Storvik (2002): Focus on $p(\overbrace{x_t, S_t}^{\text{«Dynamic duo»}}|\mathbf{y}_{1:t})$, not $p(x_t, \theta|\mathbf{y}_{1:t})$.
- Assume a properly weighted sample $\{(x_{t-1}^i, S_{t-1}^i, w_{t-1}^i), i = 1, \dots, N\}$ with respect to $p(x_{t-1}, S_{t-1}|\mathbf{y}_{1:t-1})$
- Similar recursions as before:

$$p(x_t, S_t|\mathbf{y}_{1:t-1}) = \int_{x_{t-1}} p(x_t, S_t|x_{t-1}, S_{t-1})p(x_{t-1}, S_{t-1}|\mathbf{y}_{1:t-1})dx_{t-1}dS_{t-1}$$

$$\approx \sum_{i=1}^{N} w_{t-1}^i p(x_t, S_t|x_{t-1}^i, S_{t-1}^i)$$

$$p(x_t, S_t|\mathbf{y}_{1:t}) \approx c \cdot \sum_{i=1}^{N} w_{t-1}^i p(x_t, S_t|x_{t-1}^i, S_{t-1}^i)p(y_t|x_t).$$

- Simulation from $p(x_t, S_t|x_{t-1}^i, S_{t-1}^i)$ (possible proposal function)
  1. Simulate $\theta^i \sim p(\theta|x_{t-1}^i, S_{t-1}^i) = p(\theta|S_{t-1}^i)$.
  2. Simulate $x_t^i \sim p(x_t|x_{t-1}^i, \theta^i)$.
  3. Put $S_t^i = h(S_{t-1}^i, x_{t-1}^i, x_t^i)$.

Bootstrap filter
for the «dynamic duo»…

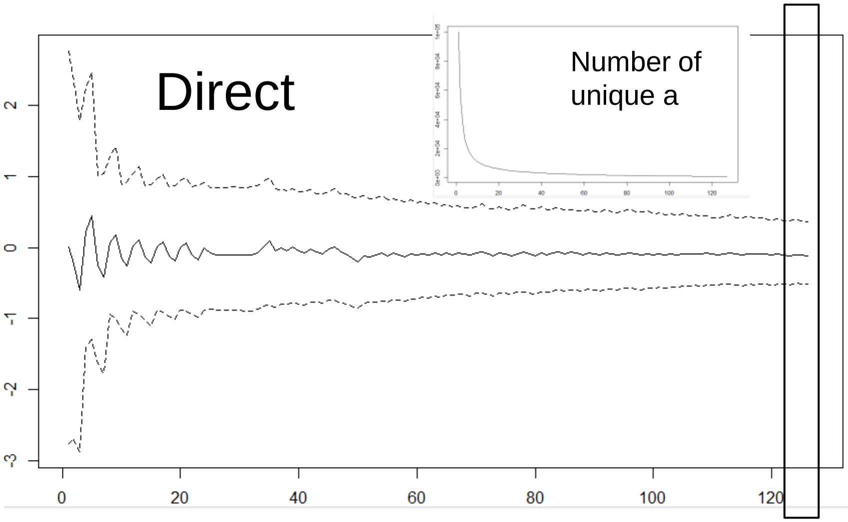
Algorithm Storvik filter

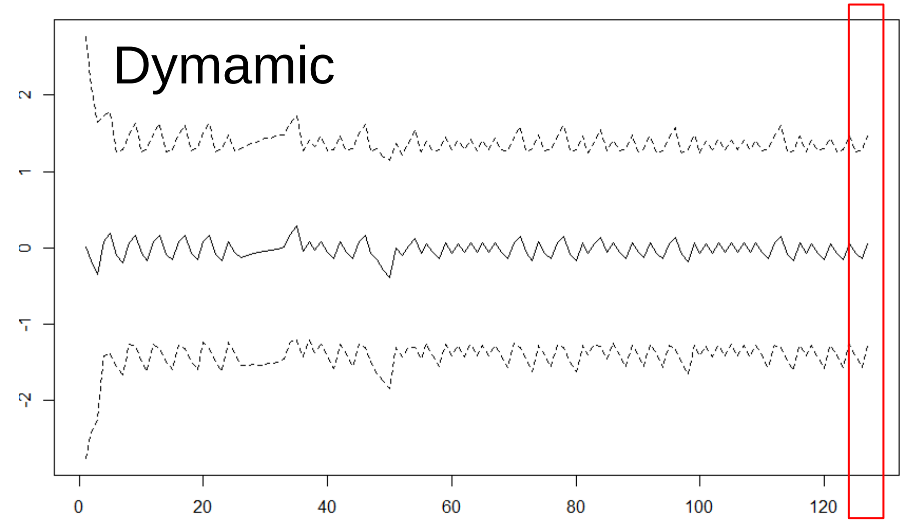
Algorithm 3 SMC with parameter updating

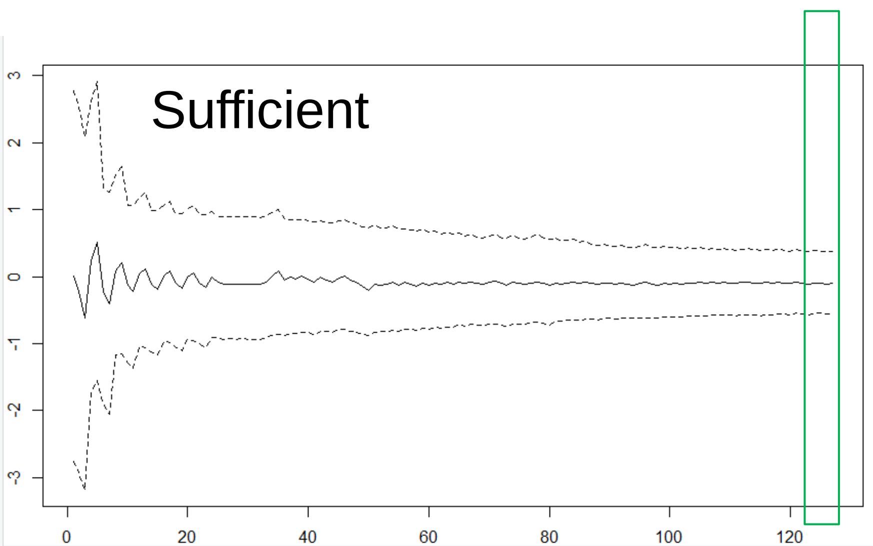
Simulate $\theta^i \sim p(\theta)$ for $i = 1, \ldots, N$. ▷ Initialization
Simulate $x_1^i \sim p(x_1|\theta^i)$ for $i = 1, \ldots, N$.
Put weights $w_1^i = p(y_1|x_1^i)$.
Put $S_1^i = 0$ for $i = 1, \ldots, N$.
for $t = 2, 3, \ldots$ **do** ▷ Sequential Monte Carlo
    Simulate $\theta^i \sim p(\theta|S_{t-1}^i)$ for $i = 1, \ldots, N$.
    Simulate $x_t^i \sim p(x_t|x_{t-1}^i, \theta^i)$ for $i = 1, \ldots, N$.
    Put weights $w_t^i = w_{t-1}^i p(y_t|x_t^i)$.
    Put $S_t^i = h(S_{t-1}^i, x_{t-1}^i, x_t^i)$.
    if $\hat{N}_{eff}$ is small **then** ▷ Resampling
        Resample (x_t^i, S_t^i) with probabilities proportional to w_t^i.
        Put $w_t^i = 1/N$.
    end if
end for

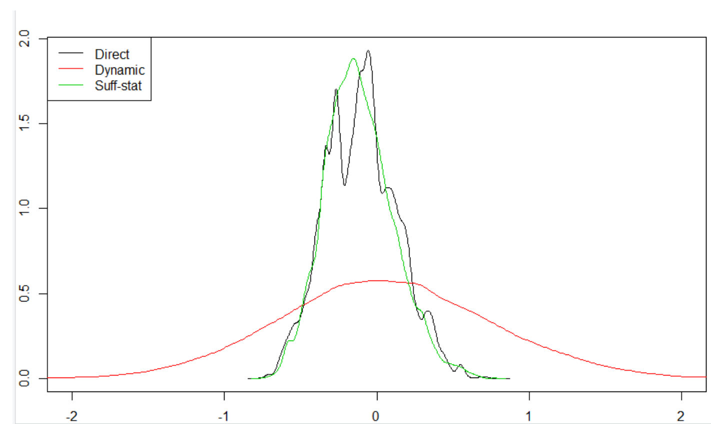
`SMC_lin_bin_parest_suff.R`

Example Lemmings evolving parameter: a









References

P. Fearnhead. Markov chain Monte Carlo, sufficient statistics, and particle filters. *Journal of Computational and Graphical Statistics*, 11(4):848–862, 2002.

J. Liu and M. West. Combined parameter and state estimation in simulation-based filtering. In *Sequential Monte Carlo methods in practice*, pages 197–223. Springer, 2001.

G. Poyiadjis, A. Doucet, and S. S. Singh. Particle approximations of the score and observed information matrix in state space models with application to parameter estimation. *Biometrika*, 98(1):65–80, 2011. doi: 10.1093/biomet/asq062. URL +http://dx.doi.org/10.1093/biomet/asq062.

G. Storvik. Particle filters for state-space models with the presence of unknown static parameters. *IEEE Transactions on signal Processing*, 50(2):281–289, 2002.