complément scientifique École Doctorale MATISSE
IRISA et INRIA, salle Markov
jeudi 26 janvier 2012

Variantes Algorithmiques et Justifications Théoriques

François Le Gland
INRIA Rennes et IRMAR

http://www.irisa.fr/aspi/legland/ed-matisse/

. . . programme of the day

#1 more general models, from non–linear and non Gaussian systems to hidden Markov models and partially observed Markov chains, so as to handle e.g.

- regime / mode switching
- correlation between state noise and observation noise

#2 for each of these models (or just for most general model), representation of

$$\mathbb{P}[X_{0:n} \in dx_{0:n} \mid Y_{0:n}]$$

as a Gibbs–Boltzmann distribution, with recursive formulation, and idem for

$$\mathbb{P}[X_n \in dx_n \mid Y_{0:n}]$$

#3 particle approximation (SIS and SIR algorithms) from either representations

#4 asymptotic behaviour as sample size goes to infinity

#5 numerous algorithmic variants

some notations (continued)

if X is a random variable taking values in E, then mapping

$$\phi \longmapsto \mathbb{E}[\phi(X)] \qquad \text{or equivalently} \qquad A \longmapsto \mathbb{P}[X \in A]$$

defines a probability distribution μ on E, denoted as

$$\mu(dx) = \mathbb{P}[X \in dx]$$

and such that

$$\mathbb{E}[\phi(X)] = \int_E \phi(x)\, \mu(dx) = \langle \mu, \phi \rangle \qquad \text{or} \qquad \mathbb{P}[X \in A] = \mu(A)$$

caracterizes uncertainty about X

transition probability kernel $M(x, dx')$ on E

collection of probability distributions on E indexed by $x \in E$

acts on functions according to

$$M\,\phi(x) = \int_E M(x, dx')\;\phi(x')$$

and acts on probability distributions according to

$$\mu\,M(dx') = \int_E \mu(dx)\;M(x, dx')$$

seen as a mixture distribution caracterized by

$$\begin{aligned}\langle \mu\,M, \phi\rangle &= \int_E [\int_E \mu(dx)\;M(x, dx')]\;\phi(x') \\ &= \int_E \mu(dx)\;[\int_E M(x, dx')\;\phi(x')] \\ &= \langle \mu, M\,\phi\rangle\end{aligned}$$

Non–linear and non Gaussian systems, and beyond

- non–linear and non Gaussian systems
- hidden Markov models
- partially observed Markov chains
- likelihood–free models

non–linear and non Gaussian systems

prior model for hidden state taking values in E

$$X_k = f_k(X_{k-1}, W_k) \qquad \text{with} \qquad W_k \sim p_k^W(dw)$$

initial condition $X_0 \sim \eta_0(dx)$

observation taking values in $\mathbb{R}^d$ with additive noise *admitting a density*

$$Y_k = h_k(X_k) + V_k \qquad \text{with} \qquad V_k \sim q_k^V(v)\, dv$$

random variables X_0, $W_1, \cdots, W_k, \cdots$ and $V_0, V_1, \cdots, V_k, \cdots$ are

mutually independent but non necessarily Gaussian

only requirement (to be used later on) : *easy* to

- *simulate* a r.v. according to $\eta_0(dx)$ or to $p_k^W(dw)$
- *evaluate* function $q_k^V(v)$ for any $v \in \mathbb{R}^d$

Proposition hidden states $\{X_k\}$ form a Markov chain taking values in E, i.e.

$$\mathbb{P}[X_k \in dx \mid X_{0:k-1}] = \mathbb{P}[X_k \in dx \mid X_{k-1}]$$

characterization in terms of *transition kernel*

$$\mathbb{P}[X_k \in dx' \mid X_{k-1} = x] = Q_k(x, dx')$$

defined (implicitly) by its action on functions

$$\begin{aligned} Q_k\,\phi(x) &= \mathbb{E}[\phi(X_k) \mid X_{k-1} = x] \\ &= \mathbb{E}[\phi(f_k(X_{k-1}, W_k)) \mid X_{k-1} = x] \\ &= \int_{\mathbb{R}^m} \phi(f_k(x, w))\; p_k^W(w)\, dw \end{aligned}$$

Remark easy to simulate next state X_k given $X_{k-1} = x$, i.e. to simulate a r.v. according to $Q_k(x, dx')$ for a given $x \in E$

indeed, set $X_k = f_k(x, W_k)$ where W_k is simulated according to $p_k^W(dw)$

Remark in general, transition kernel $Q_k(x, dx')$ does not admit a density

indeed, conditionnally to $X_{k-1} = x$, r.v. X_k necessarily belongs to subset

$$\mathcal{M}(x) = \{x' \in \mathbb{R}^m \text{ : there exist } w \in \mathbb{R}^p \text{ such that } x' = f_k(x, w)\}$$

if $p < m$ and under some mild regularity assumptions, this subset of $\mathbb{R}^m$ has zero Lebesgue measure

therefore, conditionnally to $X_{k-1} = x$, probability distribution $Q_k(x, dx')$ of r.v. X_k cannot have a density w.r.t. Lebesgue measure on $\mathbb{R}^m$

Remark if $f_k(x,w) = b_k(x) + w$ and if probability distribution $p_k^W(dw)$ of r.v. W_k admits a density, still denoted $p_k^W(w)$, i.e. if

$$X_k = b_k(X_{k-1}) + W_k \qquad \text{with} \qquad W_k \sim p_k^W(w)\,dw$$

then a more explicit expression is available

$$Q_k(x, dx') = p_k^W(x' - b_k(x))\,dx'$$

i.e. transition kernel $Q_k(x, dx')$ admits an (easy to evaluate) density

indeed, change of variable $x' = b_k(x) + w$ yields

$$\begin{aligned} Q_k\,\phi(x) &= \int_{\mathbb{R}^m} \phi(b_k(x) + w)\; p_k^W(w)\,dw \\ &= \int_{\mathbb{R}^m} \phi(x')\; p_k^W(x' - b_k(x)))\,dx' \end{aligned}$$

Proposition observations $\{Y_k\}$ satisfy *memoryless channel* assumption, i.e.

$$\mathbb{P}[Y_{0:n} \in dy_{0:n} \mid X_{0:n}] = \prod_{k=0}^{n} \mathbb{P}[Y_k \in dy_k \mid X_k]$$

characterization in terms of *emission density*

$$\mathbb{P}[Y_k \in dy \mid X_k = x] = q_k^V(y - h_k(x))\, dy$$

define *likelihood function*

$$g_k(x) = q_k^V(Y_k - h_k(x))$$

a quantitative measure of consistency between possible hidden state $x \in E$ and actual observation Y_k

Remark easy to evaluate $g_k(x)$ for any $x \in E$

hidden Markov models

motivating example

hybrid continuous / discrete systems

$$X_k = f_k(s_{k-1}, X_{k-1}, W_k)$$

$$Y_k = h_k(X_k) + V_k$$

where regime / mode sequence $\{s_k\}$ forms a Markov chain with finite state space

does not fit into non–linear and non Gaussian systems, however

- hidden states and modes $\{(X_k, s_k)\}$ jointly form a Markov chain
- observations $\{Y_k\}$ satisfy memoryless channel assumption

i.e. fits into hidden Markov models

Remark easy to simulate next state (X_k, s_k) given $(X_{k-1}, s_{k-1}) = (x, s)$

more generally, hidden states $\{X_k\}$ could form a Markov chain taking values in a quite general space E, e.g.

- hybrid continuous / discrete
- differentiable manifold
- constrained
- graphical (collection de connected edges)

characterization in terms of *transition kernel*

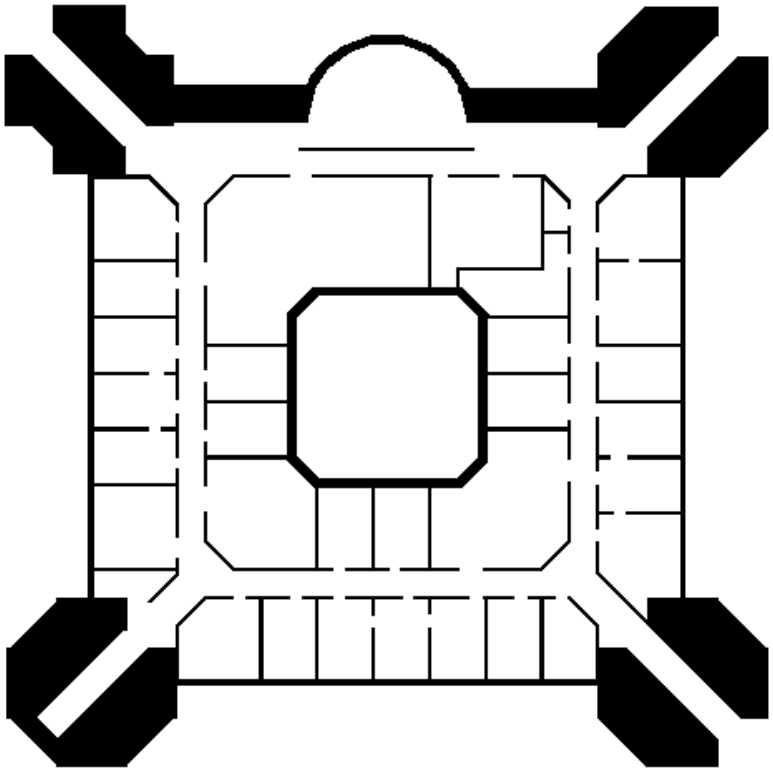
$$\mathbb{P}[X_k \in dx' \mid X_{k-1} = x] = Q_k(x, dx')$$

and initial distribution

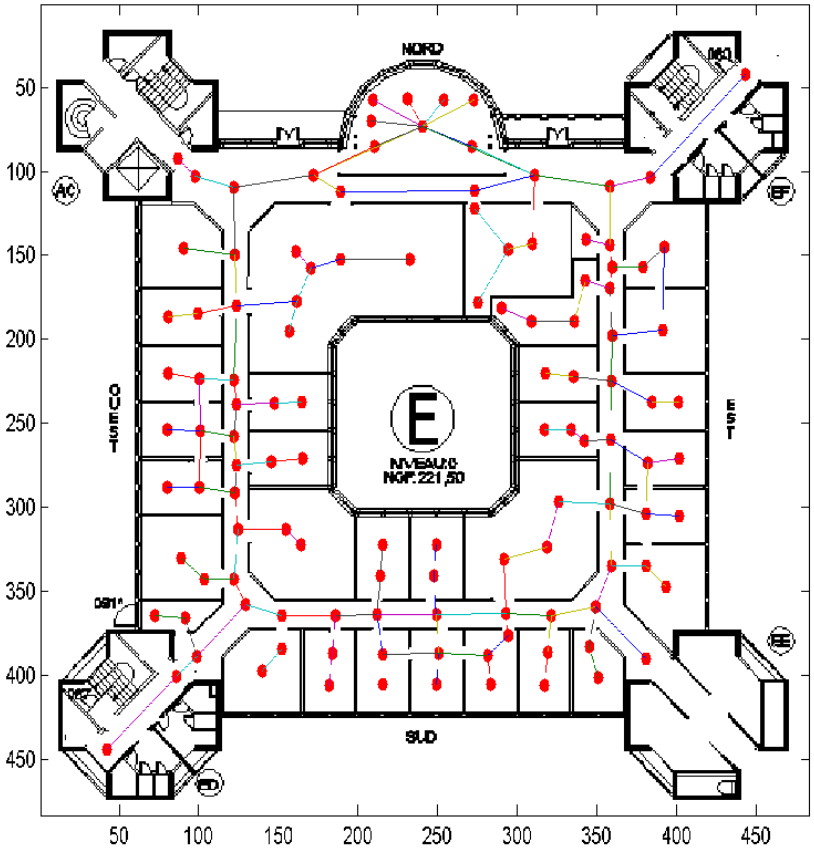
$$\mathbb{P}[X_0 \in dx] = \eta_0(dx)$$

joint probability distribution of hidden states $X_{0:n}$ verifies

$$\mathbb{P}[X_{0:n} \in dx_{0:n}] = \eta_0(dx_0) \prod_{k=1}^{n} Q_k(x_{k-1}, dx_k)$$



user should respect displacement constraints due to obstacles, as read on map



simplified model : user walks on a Voronoi graph, displacement constraints due to obstacles are taken automatically into account

observations $\{Y_k\}$ could verify *memoryless channel* assumption, i.e.

$$\mathbb{P}[Y_{0:n} \in dy_{0:n} \mid X_{0:n}] = \prod_{k=0}^{n} \mathbb{P}[Y_k \in dy_k \mid X_k]$$

characterization in terms of *emission density*

$$\mathbb{P}[Y_k \in dy \mid X_k = x] = g_k(x, y)\, \lambda_k^F(dy)$$

where nonnegative measure $\lambda_k^F(dy)$ defined on F does not depend on $x \in E$

define (abuse of notation) *likelihood function* as

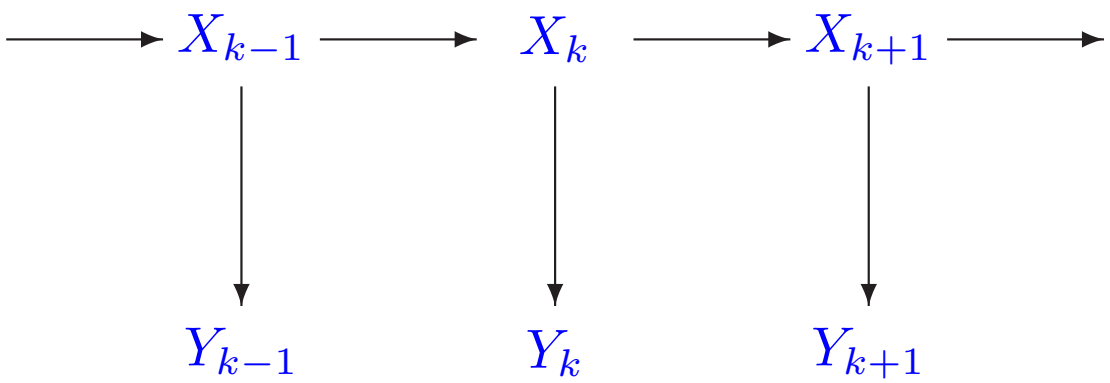
$$g_k(x) = g_k(x, Y_k)$$

a quantitative measure of consistency between $x \in E$ and observation Y_k

joint conditional distribution of observations $Y_{0:n}$ given hidden states $X_{0:n}$ verifies

$$\mathbb{P}[Y_{0:n} \in dy_{0:n} \mid X_{0:n} = x_{0:n}] = \prod_{k=0}^{n} g_k(x_k, y_k)\, \lambda_0^F(dy_0) \cdots \lambda_n^F(dy_n)$$

representation as



arrows represent dependency between random variables

only requirement (to be used later on) : *easy* to

- *simulate* for any $x \in E$, a r.v. according to transition kernel $Q_k(x, dx')$
- *evaluate* for any $x' \in E$, likelihood function $g_k(x')$

hidden Markov models : importance decomposition

motivation : from simulations seen last week, basic paradigm

- particles move according to prior model, described by its *transition kernel*
- new particles are weighted by evaluating *likelihood function*

hopefully, resulting weighted empirical distribution provides reasonable approximation to non–tractable Bayesian filter

concern / questions :

- is this safe ?
- could more information be used in mutation step ?

recall indoor navigation example : if user is detected by a beacon with known location a and with finite range R, then necessarily user position is within detection disk centered at a and with radius R

in other words, generating particles according to prior model alone could result in (a few, some, many, all) particles outside detection disk, i.e. useless particles, waste

why not generate explicitly all new particles within disk, and accomodate for wrong model by changing weights ?

more generally, why not (and how) use next observation to move particles ? ideal situation would be

- particles move according to posterior model

$\longrightarrow$ warning

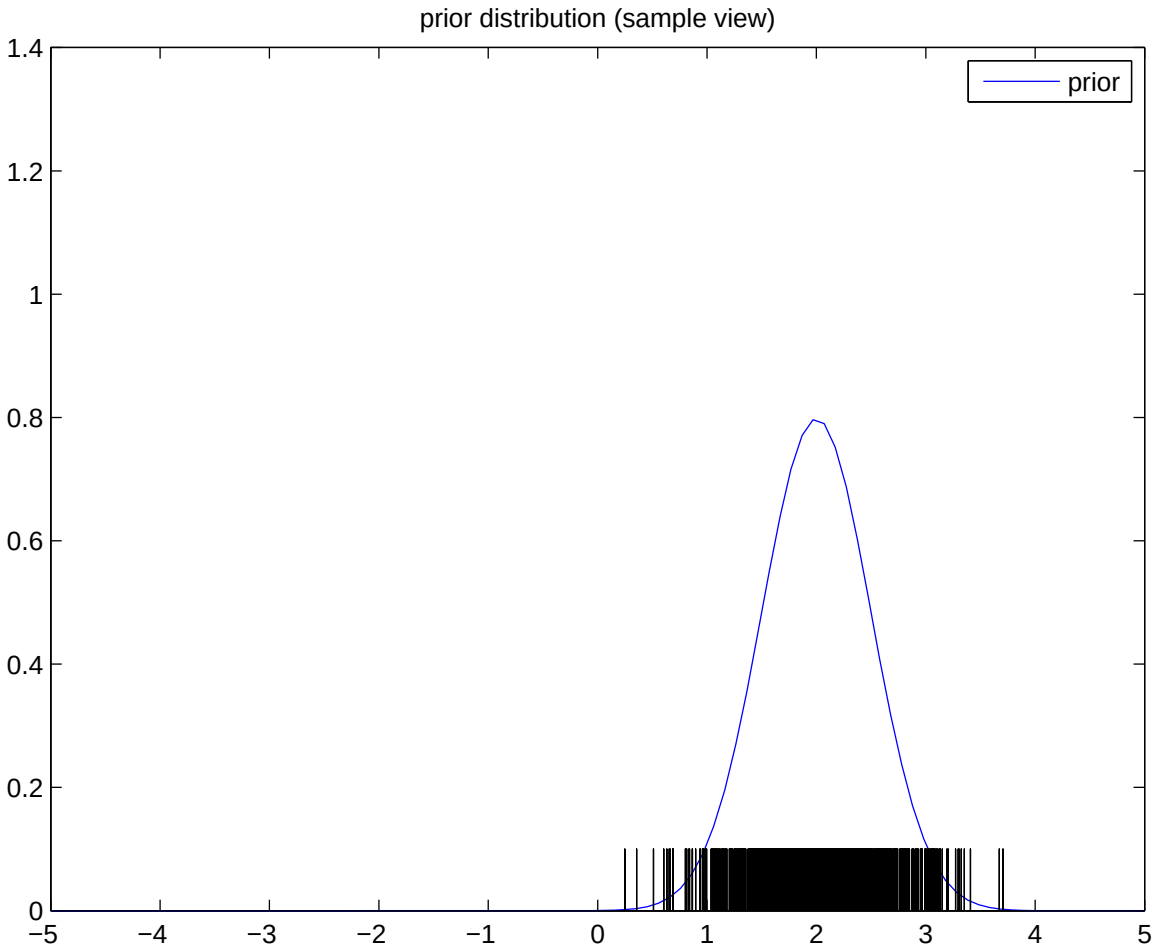


Figure 1: Prior density and generated sample

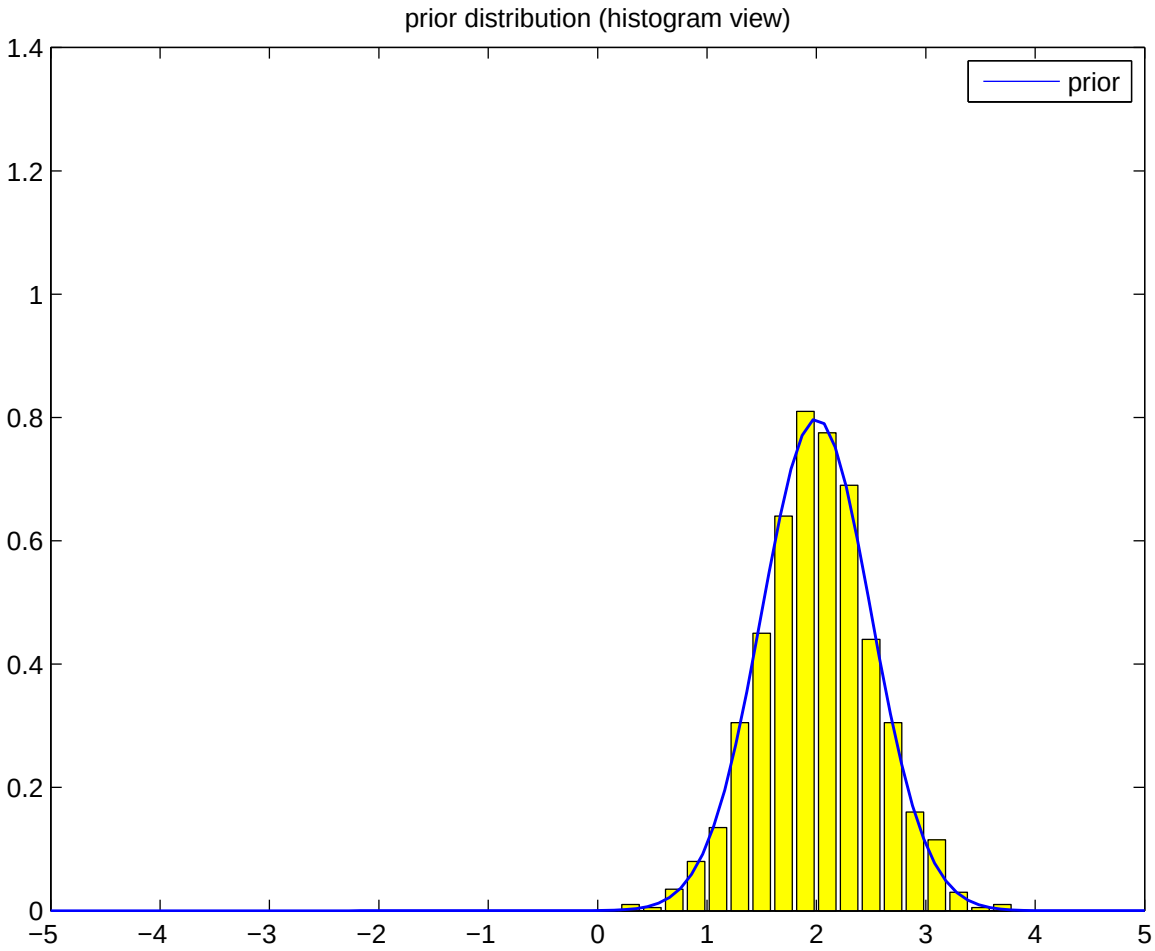


Figure 2: Prior density and histogramme associated with generated sample

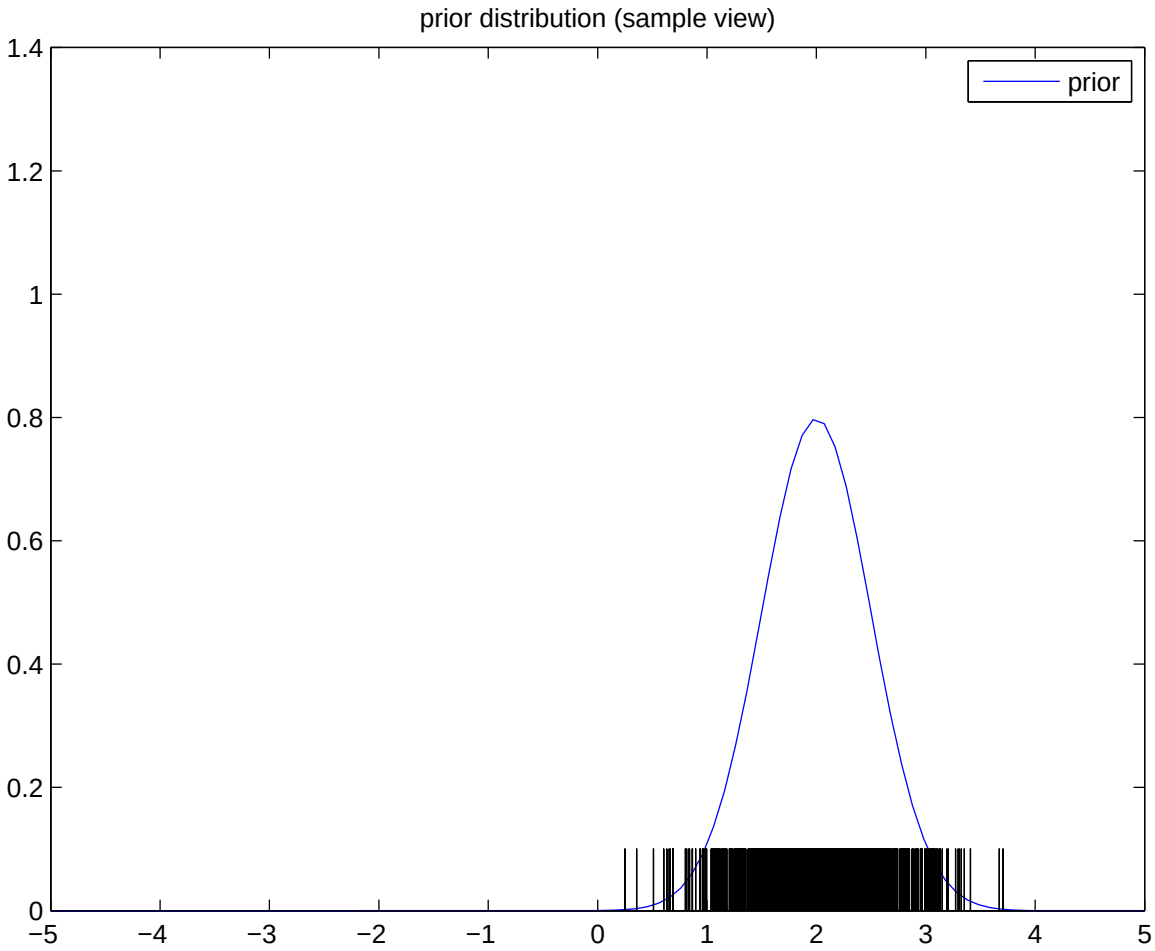


Figure 1: Prior density and generated sample

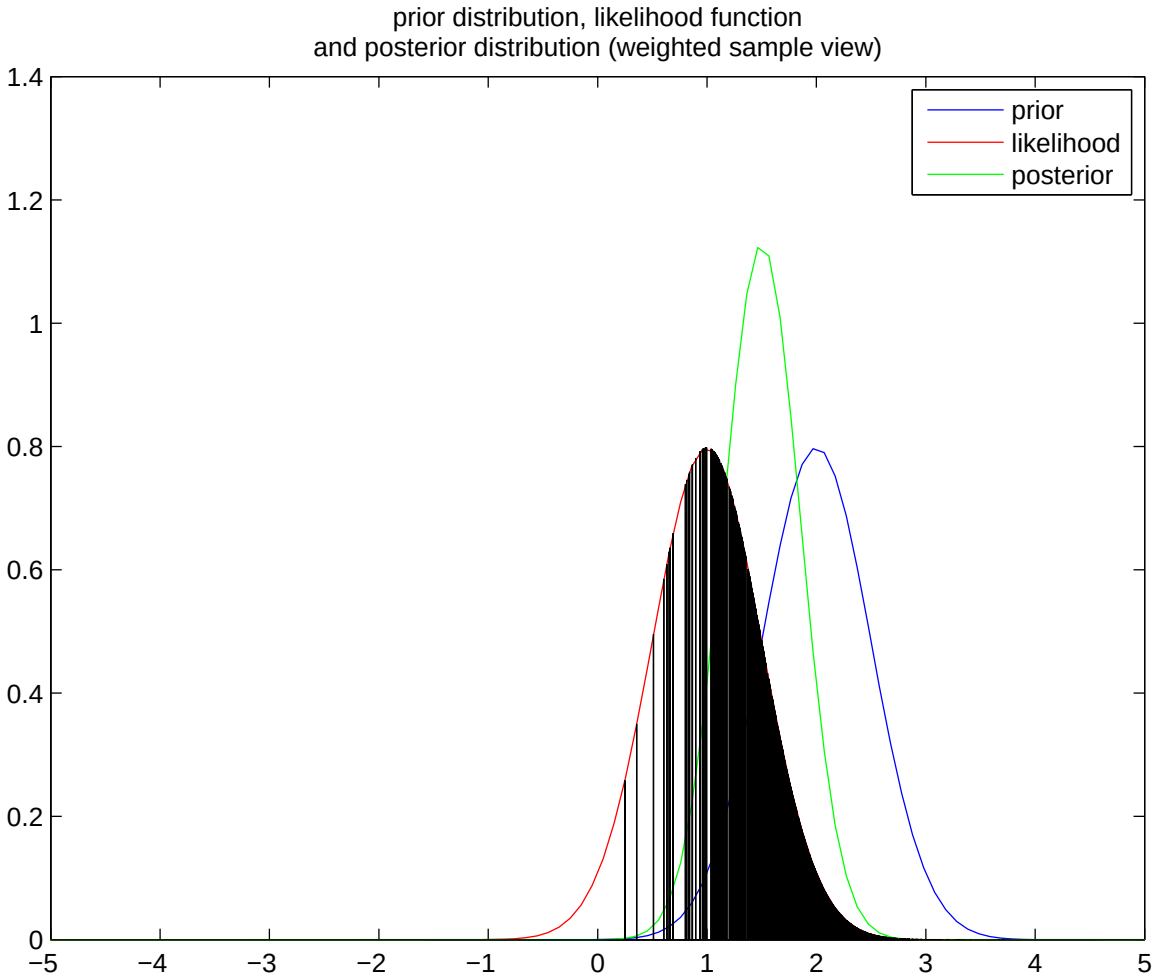


Figure 3a: Prior density, likelihood function, posterior density and weighted sample

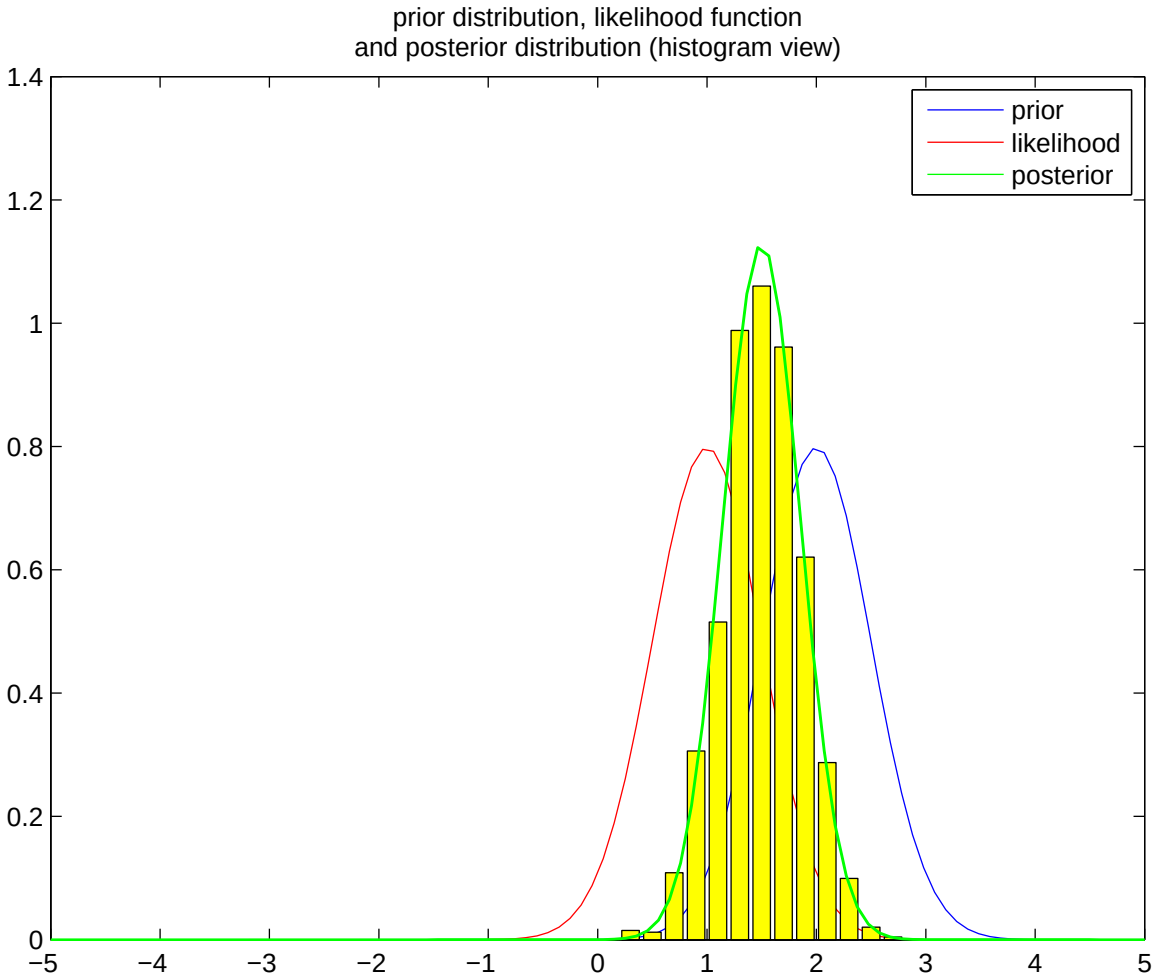


Figure 4a: Prior density, likelihood function, posterior density and histogramme associated with weighted sample

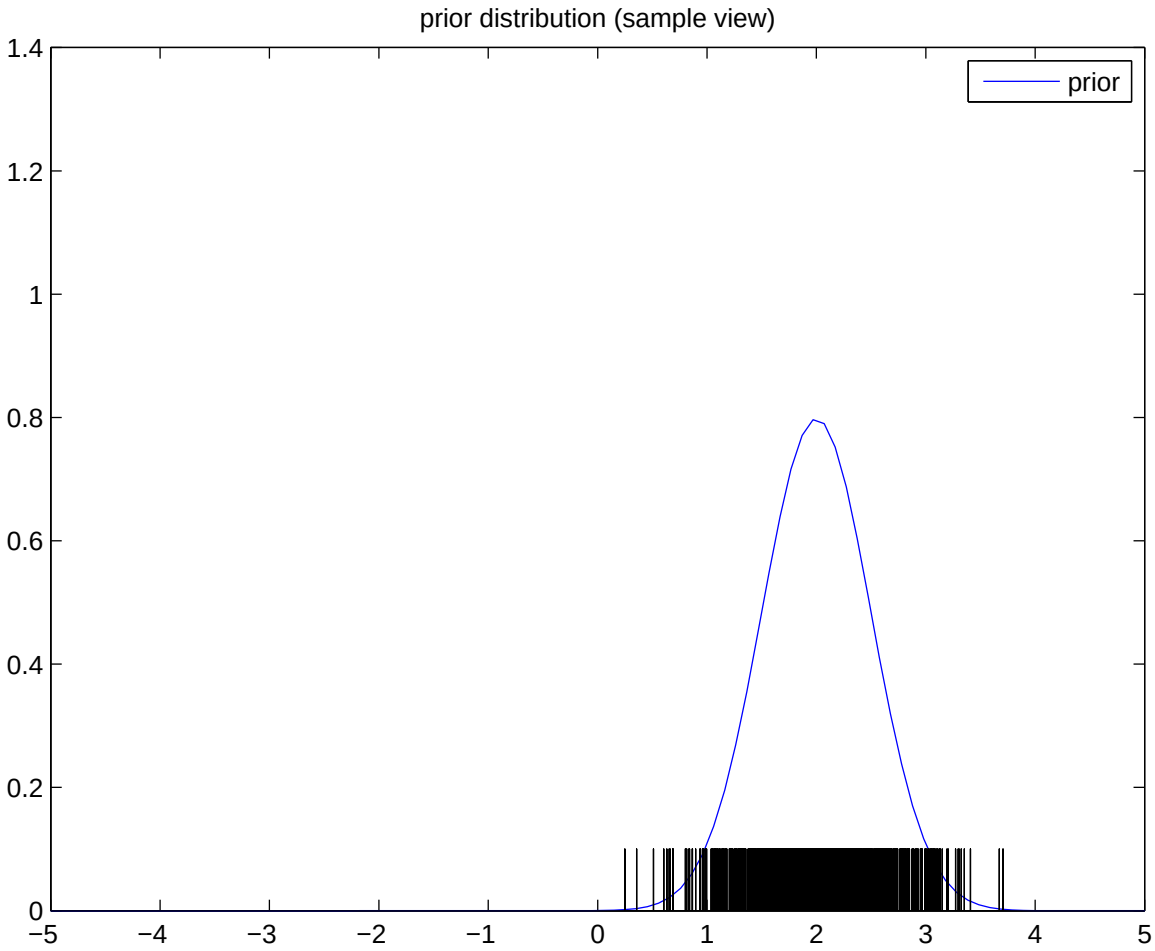


Figure 1: Prior density and generated sample

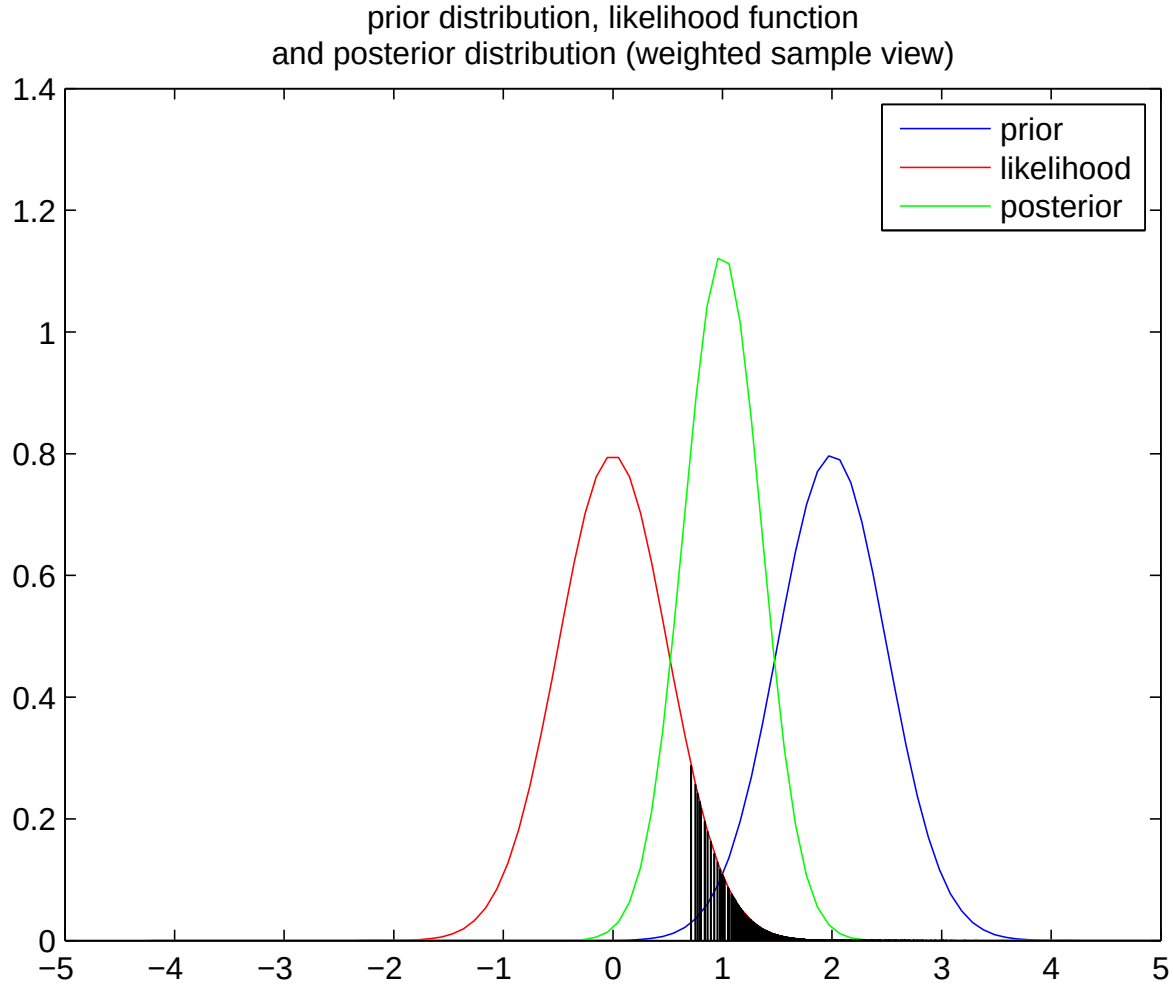


Figure 3b: Prior density, likelihood function, posterior density and weighted sample (more difficult)

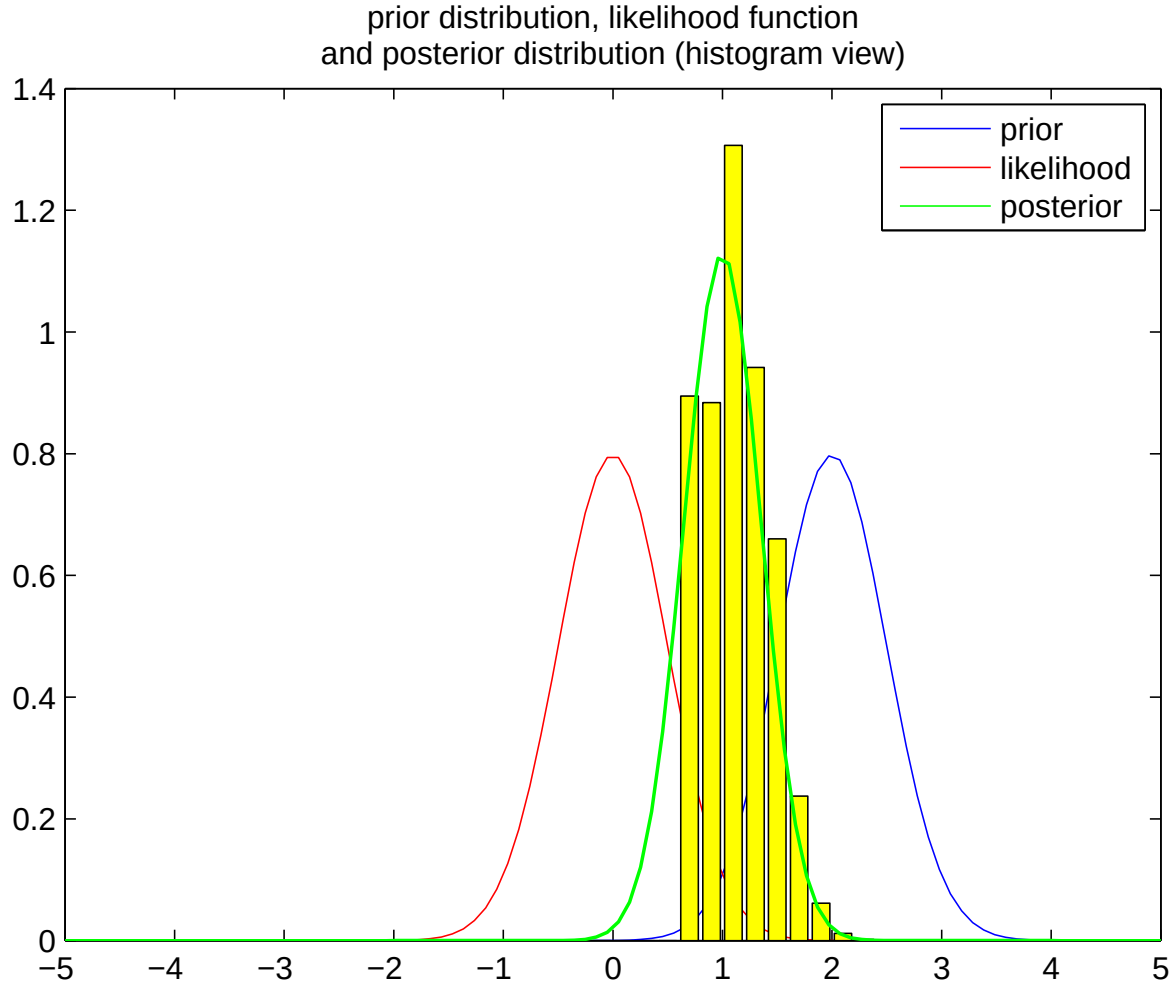


Figure 4b: Prior density, likelihood function, posterior density and histogramme associated with weighted sample (more difficult)

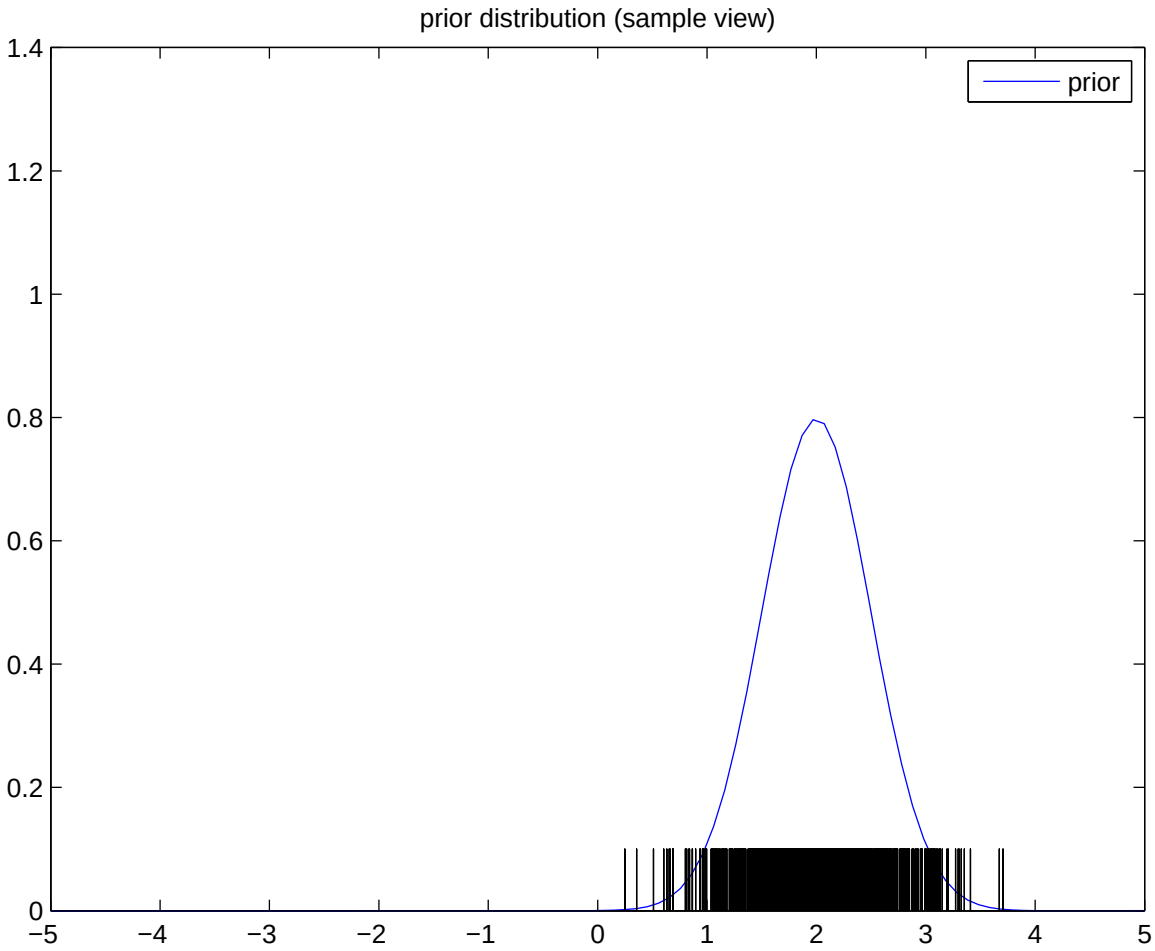


Figure 1: Prior density and generated sample

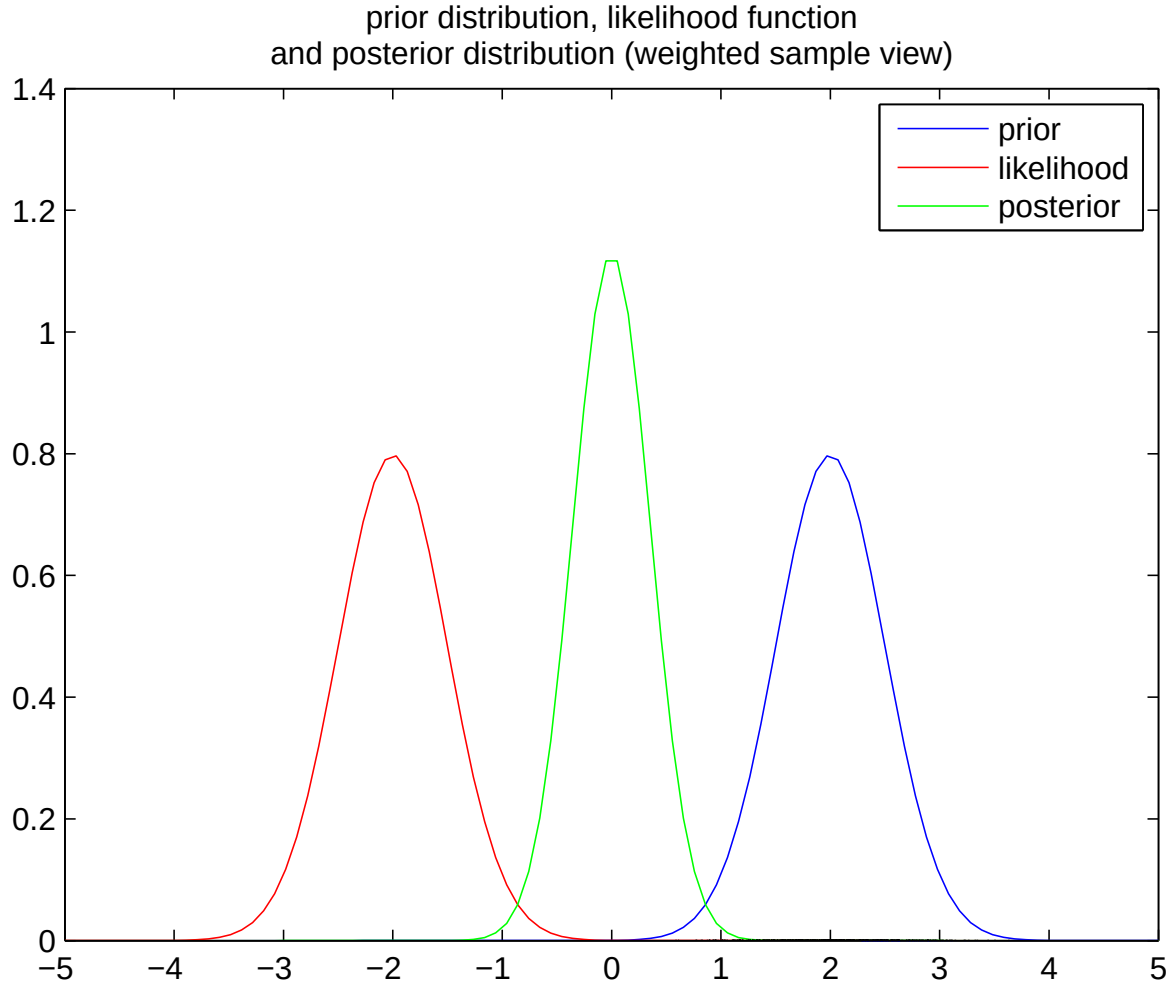


Figure 3c: Prior density, likelihood function, posterior density and weighted sample (just impossible)

possible (non unique) decomposition

$$\gamma_0(dx) = g_0(x)\,\eta_0(dx) = g_0^{\mathrm{imp}}(x)\,\eta_0^{\mathrm{imp}}(dx)$$

and

$$R_k(x, dx') = Q_k(x, dx')\,g_k(x') = g_k^{\mathrm{imp}}(x, x')\,Q_k^{\mathrm{imp}}(x, dx')$$

as product of

- a nonnegative *weight function* $g_0^{\mathrm{imp}}(x)$ or $g_k^{\mathrm{imp}}(x, x')$
- a probability distribution $\eta_0^{\mathrm{imp}}(dx)$ or a *transition kernel* $Q_k^{\mathrm{imp}}(x, dx')$

respectively, only requirement about proposed decomposition : *easy* to

- *simulate* a r.v. according to $\eta_0^{\mathrm{imp}}(dx)$
- *simulate* for any $x \in E$, a r.v. according to $Q_k^{\mathrm{imp}}(x, dx')$
- *evaluate* for any $x, x' \in E$, weighting function $g_k^{\mathrm{imp}}(x, x')$

attention : evaluating weighting function $g_k^{\mathrm{imp}}(x, x')$ requires some knowledge about transition kernels $Q_k^{\mathrm{imp}}(x, dx')$ and $Q_k(x, dx')$ (was not required originally)

popular (optimal) importance decomposition : *blind* vs. *guided* mutation

$$\mathbb{P}[X_k \in dx', Y_k \in dy' \mid X_{k-1} = x]$$

$$= \underbrace{\mathbb{P}[Y_k \in dy' \mid X_k = x', X_{k-1} = x]}_{g_k(x',y')\ \lambda_k(dy')} \underbrace{\mathbb{P}[X_k \in dx' \mid X_{k-1} = x]}_{Q_k(x,dx')}$$

and alternatively

$$\mathbb{P}[X_k \in dx', Y_k \in dy' \mid X_{k-1} = x]$$

$$= \underbrace{\mathbb{P}[X_k \in dx' \mid Y_k = y', X_{k-1} = x]}_{\widehat{Q}_k(x,y',dx')} \underbrace{\mathbb{P}[Y_k \in dy' \mid X_{k-1} = x]}_{\widehat{g}_k(x,y')\ \lambda_k(dy')}$$

i.e.

$$R_k(x, dx') = g_k(x')\ Q_k(x, dx') = \widehat{g}_k(x)\ \widehat{Q}_k(x, dx')$$

with (abuse of notation)

$$\widehat{g}_k(x) = \widehat{g}_k(x, Y_k) \qquad \text{and} \qquad \widehat{Q}_k(x, dx') = \widehat{Q}_k(x, Y_k, dx')$$

remaining question : how *easy* is it to

- *simulate* for any $x \in E$, a r.v. according to $\widehat{Q}_k(x, dx')$?
- *evaluate* for any $x \in E$, weighting function $\widehat{g}_k(x)$?

positive answer in special case : linear observations and additive Gaussian noise

$$X_k = f_k(X_{k-1}) + \sigma_k(X_{k-1})\, W_k$$

$$Y_k = H_k\, X_k + V_k$$

indeed (for simplicity, assume $\sigma_k(x) = I$)

$$Y_k = H_k\, [f_k(X_{k-1}) + W_k] + V_k = H_k\, f_k(X_{k-1}) + (H_k\, W_k + V_k)$$

conditionally on $X_{k-1} = x$, r.v. (X_k, Y_k) is jointly Gaussian, with mean and covariance matrix

$$\begin{pmatrix} f_k(x) \\ H_k\, f_k(x) \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} Q_k^W & Q_k^W\, H_k^* \\ H_k\, Q_k^W & H_k\, Q_k^W\, H_k^* + Q_k^V \end{pmatrix}$$

partially observed Markov chains

motivating example : assume (unsynchronized) sensors take noisy observations of different components of hidden state at different time instants, e.g. $X_k = (X_k^1, X_k^2)$ and for simplicity

$$X_k = f(X_{k-1}) + W_k$$

$$Y_k = \begin{cases} h^1(X_k^1) + V_k^1 & \text{at odd time instants} \\ H^2\ X_k^2 + V_k^2 & \text{at even time instants} \end{cases}$$

observing all components of hidden state is fine, but processing partial observations at each time instant can be risky, since likelihood functions will be flat along some directions : ideally, try to collect and process simultaneously two successive observations, so that likelihood functions are more peaky

down–sampling : set

$$\bar{X}_k = X_{2k+1} \qquad \text{and} \qquad \bar{Y}_k = \begin{pmatrix} Y_{2k+1} \\ Y_{2k+2} \end{pmatrix}$$

state equation

$$\begin{aligned} \bar{X}_k = X_{2k+1} &= f(X_{2k}) + W_{2k+1} \\ &= f(f(X_{2k-1}) + W_{2k}) + W_{2k+1} \end{aligned}$$

i.e.

$$\bar{X}_k = \bar{f}(\bar{X}_{k-1}, \bar{W}_k)$$

with

$$\bar{W}_k = \begin{pmatrix} W_{2k} \\ W_{2k+1} \end{pmatrix}$$

observation equation : introducing projections π_1 and π_2 on 1st and 2nd components of state vector, yields

$$\bar{Y}_k = \begin{pmatrix} Y_{2k+1} \\ Y_{2k+2} \end{pmatrix} = \begin{pmatrix} h^1(X^1_{2k+1}) + V^1_{2k+1} \\ H^2\ X^2_{2k+2} + V^2_{2k+2} \end{pmatrix}$$

$$= \begin{pmatrix} h^1(\pi_1(X_{2k+1})) + V^1_{2k+1} \\ H^2\,\pi_2\left(f(X_{2k+1}) + W_{2k+2}\right) + V^2_{2k+2} \end{pmatrix}$$

i.e.

$$\bar{Y}_k = \bar{h}(\bar{X}_k) + \bar{V}_k$$

with

$$\bar{V}_k = \begin{pmatrix} V^1_{2k+1} \\ H^2\,\pi_2(W_{2k+2}) + V^2_{2k+2} \end{pmatrix}$$

resulting system

$$\bar{X}_k = \bar{f}(\bar{X}_{k-1}, \bar{W}_k)$$

$$\bar{Y}_k = \bar{h}(\bar{X}_k) + \bar{V}_k$$

with

$$\bar{W}_k = \begin{pmatrix} W_{2k} \\ W_{2k+1} \end{pmatrix} \quad \text{and} \quad \bar{V}_k = \begin{pmatrix} V^1_{2k+1} \\ H^2\,\pi_2(W_{2k+2}) + V^2_{2k+2} \end{pmatrix}$$

clearly $\bar{W}_k$ and $\bar{V}_{k-1}$ share W_{2k} in common and are *correlated*, hence *dependent*, and memoryless channel assumption cannot hold

trick : decompose

$$\bar{W}_k = M\ \bar{V}_{k-1} + \bar{B}_k$$

where $\bar{B}_k$ and $\bar{V}_{k-1}$ are now independent, substitute in state equation and import

$$\bar{V}_{k-1} = \bar{Y}_{k-1} - \bar{h}(\bar{X}_{k-1})$$

from observation equation, yielding

$$\begin{aligned} \bar{X}_k &= \bar{f}(\bar{X}_{k-1}, M\ (\bar{Y}_{k-1} - \bar{h}(\bar{X}_{k-1})) + \bar{B}_k) \\ \bar{Y}_k &= \bar{h}(\bar{X}_k) + \bar{V}_k \end{aligned}$$

does not fit into hidden Markov model, hidden state alone does not form a Markov chain

however, hidden states and observations $\{(\bar{X}_k, \bar{Y}_k)\}$ jointly form a Markov chain, the second component of which only is observed

even more generally, with previous motivating example in mind, hidden states and observations $\{(X_k, Y_k)\}$ could jointly form a Markov chain taking values in product space $E \times F$

characterization in terms of *transition kernel*

$$\mathbb{P}[X_k \in dx', Y_k \in dy' \mid X_{k-1} = x, Y_{k-1} = y] = R_k(x, y, y', dx') \; \lambda_k^F(y, dy')$$

and initial distribution

$$\mathbb{P}[X_0 \in dx, Y_0 \in dy] = \gamma_0(y, dx) \; \lambda_0^F(dy)$$

attention : hidden states $\{X_k\}$ alone need not form a Markov chain

joint probability distribution of hidden states and observations $(X_{0:n}, Y_{0:n})$

$$\mathbb{P}[X_{0:n} \in dx_{0:n}, Y_{0:n} \in dy_{0:n}]$$

$$= \gamma_0(y_0, dx_0) \prod_{k=1}^{n} R_k(x_{k-1}, y_{k-1}, y_k, dx_k) \; \lambda_0^F(dy_0) \; \lambda_k^F(y_{k-1}, dy_k)$$

partially observed Markov chains : importance decomposition

required (non unique) decomposition

$$\gamma_0(dx) = g_0^{\mathrm{imp}}(x)\, \eta_0^{\mathrm{imp}}(dx)$$

and

$$R_k(x, dx') = g_k^{\mathrm{imp}}(x, x')\, Q_k^{\mathrm{imp}}(x, dx')$$

as product of

- a nonnegative *weight function* $g_0^{\mathrm{imp}}(x)$ or $g_k^{\mathrm{imp}}(x, x')$
- a probability distribution $\eta_0^{\mathrm{imp}}(dx)$ or a *transition kernel* $Q_k^{\mathrm{imp}}(x, dx')$

respectively, only requirement about proposed decomposition : *easy* to

- *simulate* a r.v. according to $\eta_0^{\mathrm{imp}}(dx)$
- *simulate* for any $x \in E$, a r.v. according to $Q_k^{\mathrm{imp}}(x, dx')$
- *evaluate* for any $x, x' \in E$, weighting function $g_k^{\mathrm{imp}}(x, x')$

likelihood–free models

so far, at least implicitly, additive observation noise has been assumed

$$Y_k = h(X_k) + V_k \qquad \text{with} \qquad V_k \sim q_k^V(v)\, dv$$

with known and explicit form for probability density $q_k^V(v)$

this was key assumption in deriving expression of density emission

$$\mathbb{P}[Y_k \in dy \mid X_k = x] = g_k(x, y)\, \lambda_k(dy)$$

hence explicit expression of likelihood function

questions : could anything be said in more general cases where

- no explicit expression is available for a density, or it does not even exist
- non–additive observation noise, with dimension smaller than observation, i.e.

$$Y_k = h(X_k, V_k)$$

- perfect observations, i.e. observation noise is simply not present

$$Y_k = h(X_k)$$

trick, a form of ABC (approximate Bayesian computation) : pretend that observations are produced by slightly perturbed but regular model, i.e.

$$Y_k = h(X_k) + V_k + \varepsilon\, U_k$$

or

$$Y_k = h(X_k, V_k) + \varepsilon\, U_k$$

or

$$Y_k = h(X_k) + \varepsilon\, U_k$$

depending on the case under consideration, with $U_k \sim q_k^U(u)\, du$ and set (X_k, V_k) as new hidden state

new requirement : *easy* to

- *simulate* (X_k, V_k) jointly
- *evaluate* density $q_k^U(u)$

Bayesian filter

- hidden Markov models
  - representation as Gibbs–Boltzmann distribution
  - recursive formulation
- partially observed Markov chains + given importance decomposition
  - representation as Gibbs–Boltzmann distribution

Bayesian filter : hidden Markov models : representation

Theorem joint conditional distribution of hidden state sequence $X_{0:n}$ given observations $Y_{0:n}$ as a Gibbs–Boltzmann distribution

$$\mathbb{P}[X_{0:n} \in dx_{0:n} \mid Y_{0:n}] \propto \underbrace{\prod_{k=0}^{n} g_k(x_k)}_{g_{0:n}(x_{0:n})} \underbrace{\eta_0(dx_0) \prod_{k=1}^{n} Q_k(x_{k-1}, dx_k)}_{\eta_{0:n}(dx_{0:n})}$$

with *likelihood functions* defined (abuse of notation) as

$$g_k(x) = g_k(x, Y_k)$$

and with joint probability distribution of hidden state sequence $X_{0:n}$

$$\eta_{0:n}(dx_{0:n}) = \mathbb{P}[X_{0:n} \in dx_{0:n}] = \eta_0(dx_0) \prod_{k=1}^{n} Q_k(x_{k-1}, dx_k)$$

general principle :

$$\boxed{p_{X|Y=y}(x) = \frac{p_{X,Y}(x,y)}{p_Y(y)} \Longrightarrow p_{X|Y}(x) \propto p_{X,Y}(x,Y)}$$

Proof Bayes rule + Markov property + memoryless channel assumption, yield joint probability distribution of hidden states and observations $(X_{0:n}, Y_{0:n})$

$$\mathbb{P}[X_{0:n} \in dx_{0:n}, Y_{0:n} \in dy_{0:n}]$$

$$= \mathbb{P}[Y_{0:n} \in dy_{0:n} \mid X_{0:n} = x_{0:n}] \, \mathbb{P}[X_{0:n} \in dx_{0:n}]$$

$$= \eta_0(dx_0) \prod_{k=1}^{n} Q_k(x_{k-1}, dx_k) \prod_{k=0}^{n} g_k(x_k, y_k) \, \lambda_0^F(dy_0) \cdots \lambda_n^F(dy_n)$$

hence

$$\mathbb{P}[X_{0:n} \in dx_{0:n} \mid Y_{0:n}] \propto \eta_0(dx_0) \prod_{k=1}^{n} Q_k(x_{k-1}, dx_k) \prod_{k=0}^{n} g_k(x_k) \qquad \square$$

Remark for any function f depending on whole trajectory

$$\mathbb{E}[f(X_{0:n}) \mid Y_{0:n}] \propto \int_E \cdots \int_E f(x_{0:n})\, g_{0:n}(x_{0:n})\, \eta_{0:n}(dx_{0:n})$$

$$\propto \mathbb{E}[f(X_{0:n}) \prod_{k=0}^{n} g_k(X_k)]$$

expectation w.r.t. hidden state sequence $X_{0:n}$, while observations $Y_{0:n}$ are fixed implicit parameters in likelihood functions : recall (abuse of notation)

$$g_k(x) = g_k(x, Y_k)$$

if $f = \phi \circ \pi$ depends only upon last state, then

$$\langle \mu_n, \phi \rangle = \mathbb{E}[\phi(X_n) \mid Y_{0:n}] \propto \mathbb{E}[\phi(X_n) \prod_{k=0}^{n} g_k(X_k)] = \langle \gamma_n, \phi \rangle$$

which defines unnormalized distribution $\gamma_n(dx)$ implicitly, through its action on arbitrary functions

for a given importance decomposition

$$\mathbb{P}[X_{0:n} \in dx_{0:n} \mid Y_{0:n}] \propto \eta_0(dx_0) \prod_{k=1}^{n} Q_k(x_{k-1}, dx_k) \prod_{k=0}^{n} g_k(x_k)$$

$$\propto \underbrace{\eta_0^{\mathrm{imp}}(dx_0) \prod_{k=1}^{n} Q_k^{\mathrm{imp}}(x_{k-1}, dx_k)}_{\eta_{0:n}^{\mathrm{imp}}(dx_{0:n})} \underbrace{\prod_{k=0}^{n} g_k^{\mathrm{imp}}(x_k)}_{g_{0:n}^{\mathrm{imp}}(x_{0:n})}$$

Bayesian filter : hidden Markov models : recursive formulation

Theorem Bayesian filter $\mu_k(dx) = \mathbb{P}[X_k \in dx \mid Y_{0:k}]$ satisfies

$$\mu_{k-1} \xrightarrow{\text{prediction}} \eta_k = \mu_{k-1} Q_k \xrightarrow{\text{correction}} \mu_k = g_k \cdot \eta_k$$

with initial condition $\eta_0(dx) = \mathbb{P}[X_0 \in dx]$

Remark in Theorem statement

$$\mu_{k-1} Q_k(dx') = \int_E \mu_{k-1}(dx)\, Q_k(x, dx')$$

denotes mixture distribution resulting from transition kernel $Q_k(x, dx')$ acting on probability distribution $\mu_{k-1}(dx)$, and

$$g_k \cdot \eta_k = \frac{g_k\ \eta_k}{\langle \eta_k, g_k \rangle}$$

denotes (projective) product of prior probability distribution $\eta_k(dx')$ with likelihood function $g_k(x')$

Proof recall representation for joint conditional probability distribution of hidden state sequence $X_{0:k}$ given observations $Y_{0:k}$

$$\mathbb{P}[X_{0:k} \in dx_{0:k} \mid Y_{0:k}] \propto \eta_0(dx_0) \prod_{p=1}^{k} Q_p(x_{p-1}, dx_p) \prod_{p=0}^{k} g_p(x_p)$$

$$\propto g_k(x_k)\, Q_k(x_{k-1}, dx_k)\, \mathbb{P}[X_{0:k-1} \in dx_{0:k-1} \mid Y_{0:k-1}]$$

integration w.r.t. variables $x_{0:k-1}$ (and in RHS, first w.r.t. variables $x_{0:k-2}$ and next w.r.t. variable x_{k-1}), provides conditional distribution of current hidden state X_k given observations $Y_{0:k}$, i.e. Bayesian filter, as

$$\mu_k(dx_k) = \mathbb{P}[X_k \in dx_k \mid Y_{0:k}]$$

$$\propto g_k(x_k) \int_E Q_k(x_{k-1}, dx_k)\, \mathbb{P}[X_{k-1} \in dx_{k-1} \mid Y_{0:k-1}]$$

$$\propto g_k(x_k) \underbrace{\int_E \mu_{k-1}(dx_{k-1})\, Q_k(x_{k-1}, dx_k)}_{\eta_k(dx_k)} \qquad \square$$

Remark unnormalized version satisfies recurrent relation

$$\gamma_k(dx') = g_k(x') \int_E \gamma_{k-1}(dx)\, Q_k(x, dx') \qquad \text{and} \qquad \mu_k = \frac{\gamma_k}{\langle \gamma_k, 1 \rangle}$$

or equivalently

$$\gamma_k(dx') = \int_E \gamma_{k-1}(dx)\, R_k(x, dx')$$

introducing nonnegative kernel $R_k(x, dx') = Q_k(x, dx')\, g_k(x')$

Bayesian filter : partially observed Markov chains : representation

Theorem joint conditional distribution of hidden state sequence $X_{0:n}$ given observations $Y_{0:n}$

$$\mathbb{P}[X_{0:n} \in dx_{0:n} \mid Y_{0:n}] \propto \gamma_0(dx_0) \prod_{k=1}^{n} R_k(x_{k-1}, dx_k)$$

with nonnegative distribution defined (abuse of notation) as

$$\gamma_0(dx) = \gamma_0(Y_0, dx)$$

and with nonnegative kernel defined (abuse of notation) as

$$R_k(x_{k-1}, dx_k) = R_k(x_{k-1}, Y_{k-1}, Y_k, dx_k)$$

general principle :

$$\boxed{p_{X|Y=y}(x) = \frac{p_{X,Y}(x,y)}{p_Y(y)} \Longrightarrow p_{X|Y}(x) \propto p_{X,Y}(x,Y)}$$

Proof by definition joint probability distribution of hidden states and observations $(X_{0:n}, Y_{0:n})$

$$\mathbb{P}[X_{0:n} \in dx_{0:n}, Y_{0:n} \in dy_{0:n}]$$

$$= \gamma_0(y_0, dx_0) \prod_{k=1}^{n} R_k(x_{k-1}, y_{k-1}, y_k, dx_k) \; \lambda_0^F(dy_0) \; \lambda_k^F(y_{k-1}, dy_k)$$

hence

$$\mathbb{P}[X_{0:n} \in dx_{0:n} \mid Y_{0:n}] \propto \gamma_0(dx_0) \prod_{k=1}^{n} R_k(x_{k-1}, dx_k) \qquad \square$$

for a given importance decomposition

$$\mathbb{P}[X_{0:n} \in dx_{0:n} \mid Y_{0:n}] \propto \gamma_0(dx_0) \prod_{k=1}^{n} R_k(x_{k-1}, dx_k)$$

$$\propto \underbrace{\eta_0^{\mathrm{imp}}(dx_0) \prod_{k=1}^{n} Q_k^{\mathrm{imp}}(x_{k-1}, dx_k)}_{\eta_{0:n}^{\mathrm{imp}}(dx_{0:n})} \underbrace{\prod_{k=0}^{n} g_k^{\mathrm{imp}}(x_k)}_{g_{0:n}^{\mathrm{imp}}(x_{0:n})}$$

Bayesian filter : partially observed Markov chains : recursive formulation

Theorem Bayesian filter $\mu_k(dx) = \mathbb{P}[X_k \in dx \mid Y_{0:k}]$ satisfies

$$\mu_k(dx') \propto \int_E \mu_{k-1}(dx)\ R_k(x, dx')$$

with initial condition $\mu_0(dx) \propto \gamma_0(dx)$

Remark unnormalized version satisfies recurrent relation

$$\gamma_k(dx') = \int_E \gamma_{k-1}(dx)\ R_k(x, dx') \qquad \text{and} \qquad \mu_k = \frac{\gamma_k}{\langle \gamma_k, 1 \rangle}$$

Proof recall representation for joint conditional probability distribution of hidden state sequence $X_{0:k}$ given observations $Y_{0:k}$

$$\mathbb{P}[X_{0:k} \in dx_{0:k} \mid Y_{0:k}] \propto \gamma_0(dx_0) \prod_{p=1}^{k} R_p(x_{p-1}, dx_p)$$

$$\propto R_k(x_{k-1}, dx_k)\, \mathbb{P}[X_{0:k-1} \in dx_{0:k-1} \mid Y_{0:k-1}]$$

integration w.r.t. variables $x_{0:k-1}$ (and in RHS, first w.r.t. variables $x_{0:k-2}$ and next w.r.t. variable x_{k-1}), provides conditional distribution of current hidden state X_k given observations $Y_{0:k}$, i.e. Bayesian filter, as

$$\mu_k(dx_k) = \mathbb{P}[X_k \in dx_k \mid Y_{0:k}]$$

$$\propto \int_E R_k(x_{k-1}, dx_k)\, \mathbb{P}[X_{k-1} \in dx_{k-1} \mid Y_{0:k-1}]$$

$$\propto \int_E \mu_{k-1}(dx_{k-1})\, R_k(x_{k-1}, dx_k) \qquad \square$$

Monte Carlo approximation : particle filters

- Monte Carlo methods : importance sampling
- importance sampling $\longrightarrow$ SIS algorithm
  - derived from Bayesian filter representation
  - recursive formulation
- redistribution $\longrightarrow$ SIR algorithm, adaptive redistribution
  - derived directly from Bayesian filter recursive formulation
- estimation error, CLT

Monte Carlo methods

if *computing* an integral (or a mathematical expectation)

$$\langle \mu, \phi \rangle = \int_E \phi(x)\, \mu(dx) = \mathbb{E}[\phi(X)] \qquad \text{with} \qquad X \sim \mu(dx)$$

is difficult, but *simulating* a r.v. according to distribution μ is easy, then introduce empirical probability distribution

$$S^N(\mu) = \frac{1}{N} \sum_{i=1}^{N} \delta_{\xi_i}$$

where $(\xi_1, \cdots, \xi_N)$ is an N–sample distributed according to μ, and approximation

$$\langle \mu, \phi \rangle \approx \langle S^N(\mu), \phi \rangle = \frac{1}{N} \sum_{i=1}^{N} \phi(\xi_i)$$

by *law of large numbers*

$$\langle S^N(\mu), \phi \rangle \longrightarrow \langle \mu, \phi \rangle$$

in probability as $N \uparrow \infty$, with speed $1/\sqrt{N}$

indeed

$$\langle S^N(\mu) - \mu, \phi \rangle = \frac{1}{N} \sum_{i=1}^{N} (\phi(\xi_i) - \langle \mu, \phi \rangle)$$

hence (non–asymptotical) *mean square* error

$$\mathbb{E} | \langle S^N(\mu) - \mu, \phi \rangle |^2 = \frac{1}{N} \operatorname{var}(\phi, \mu)$$

since

$$\frac{1}{N^2} \sum_{i,j=1}^{N} \mathbb{E}[(\phi(\xi_i) - \langle \mu, \phi \rangle) (\phi(\xi_j) - \langle \mu, \phi \rangle)] = \frac{1}{N^2} \sum_{i=1}^{N} \underbrace{\mathbb{E} | \phi(\xi_i) - \langle \mu, \phi \rangle |^2}_{\operatorname{var}(\phi, \mu)}$$

and *central limit theorem* holds

$$\sqrt{N} \langle S^N(\mu) - \mu, \phi \rangle = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} (\phi(\xi_i) - \langle \mu, \phi \rangle) \Longrightarrow \mathcal{N}(0, \operatorname{var}(\phi, \mu))$$

in distribution as $N \uparrow \infty$

important special case : Gibbs–Boltzmann distribution

$$\mu = g \cdot \eta = \frac{g\,\eta}{\langle \eta, g \rangle} \qquad \text{i.e.} \qquad \langle \mu, \phi \rangle = \frac{\langle \eta, g\,\phi \rangle}{\langle \eta, g \rangle}$$

with (non unique) decomposition in terms of

- a probability distribution η
- a nonnegative function g

introduce unnormalized distribution defined by

$$\langle \gamma, \phi \rangle = \langle \eta, g\,\phi \rangle = \mathbb{E}[g(\Xi)\,\phi(\Xi)] \qquad \text{hence} \qquad \langle \mu, \phi \rangle = \frac{\langle \eta, g\,\phi \rangle}{\langle \eta, g \rangle} = \frac{\langle \gamma, \phi \rangle}{\langle \gamma, 1 \rangle} \quad (\star)$$

where r.v. Ξ is distributed according to η

motivation : Bayes rule

"posterior distribution" $\propto$ "likelihood function" $\times$ "prior distribution"

if simulating a r.v. according to μ is difficult, but

- *simulating* a r.v. according to η
- and *evaluating* nonnegative function $g(x)$ for any x

is easy, then it is possible to

- approximate μ by a weighted empirical probability distribution associated with a sample distributed according to η and weighted with nonnegative function $g(x)$

even though normalizing constant $\langle \eta, g \rangle$ might be unknown

importance sampling

idea : approximate numerator and denominator in $(\star)$ with a unique sample distributed according to η : introduce approximation

$$\langle \gamma, \phi \rangle = \langle \eta, g\,\phi \rangle \approx \langle S^N(\eta), g\,\phi \rangle = \frac{1}{N} \sum_{i=1}^{N} g(\xi^i)\,\phi(\xi^i)$$

hence

$$\langle \mu, \phi \rangle = \langle g \cdot \eta, \phi \rangle \approx \langle g \cdot S^N(\eta), \phi \rangle = \frac{\displaystyle\sum_{i=1}^{N} g(\xi^i)\,\phi(\xi^i)}{\displaystyle\sum_{i=1}^{N} g(\xi^i)}$$

where $(\xi^1, \cdots, \xi^N)$ is an N–sample with common probability distribution η

in other words

$$\gamma \approx \gamma^N = g\, S^N(\eta) = \frac{1}{N} \sum_{i=1}^{N} g(\xi^i)\, \delta_{\xi^i}$$

and

$$\mu \approx \mu^N = g \cdot S^N(\eta) = \sum_{i=1}^{N} \frac{g(\xi^i)}{\sum_{j=1}^{N} g(\xi^j)}\, \delta_{\xi^i} = \sum_{i=1}^{N} w^i\, \delta_{\xi^i}$$

where nonnegative normalized weights $(w^1, \cdots, w^N)$ are defined for any $i = 1 \cdots N$ by

$$w^i = \frac{g(\xi^i)}{\sum_{j=1}^{N} g(\xi^j)}$$

importance sampling ⟶ SIS algorithm

recall Bayesian filter representation as a Gibbs–Boltzmann distribution

$$\mu_{0:n} = g_{0:n} \cdot \eta_{0:n} = \frac{g_{0:n}\,\eta_{0:n}}{\langle \eta_{0:n}, g_{0:n} \rangle} \qquad \text{i.e.} \qquad \langle \mu_{0:n}, f \rangle = \frac{\langle \eta_{0:n}, g_{0:n}\, f \rangle}{\langle \eta_{0:n}, g_{0:n} \rangle} = \frac{\langle \gamma_{0:n}, f \rangle}{\langle \gamma_{0:n}, 1 \rangle}$$

with

$$g_{0:n}(x_{0:n}) = \prod_{k=0}^{n} g_k(x_{k-1}, x_k)$$

and with joint probability distribution of hidden states $X_{0:n}$

$$\eta_{0:n}(dx_{0:n}) = \mathbb{P}[X_{0:n} \in dx_{0:n}] = \eta_0(dx_0) \prod_{k=1}^{n} Q_k(x_{k-1}, dx_k)$$

unnormalized version defined as

$$\langle \gamma_{0:n}, f \rangle = \langle \eta_{0:n}, g_{0:n}\, f \rangle = \mathbb{E}[g_{0:n}(X_{0:n})\, f(X_{0:n})]$$

and if $f = \phi \circ \pi$ depends only upon last state and not on whole trajectory, then

$$\langle \gamma_{0:n}, \phi \circ \pi \rangle = \mathbb{E}[g_{0:n}(X_{0:n})\ \phi \circ \pi(X_{0:n})\,] = \mathbb{E}[\phi(X_n) \prod_{k=0}^{n} g_k(X_{k-1}, X_k)\,] = \langle \gamma_n, \phi \rangle$$

importance sampling : approximation

$$\langle \gamma_{0:n}, f \rangle = \langle \eta_{0:n}, g_{0:n}\, f \rangle \approx \langle S^N(\eta_{0:n}), g_{0:n}\, f \rangle = \frac{1}{N} \sum_{i=1}^{N} g_{0:n}(\xi^i_{0:n})\, f(\xi^i_{0:n})$$

and

$$\langle \mu_{0:n}, f \rangle = \langle g_{0:n} \cdot \eta_{0:n}, f \rangle \approx \langle g_{0:n} \cdot S^N(\eta_{0:n}), f \rangle = \frac{\displaystyle\sum_{i=1}^{N} g_{0:n}(\xi^i_{0:n})\, f(\xi^i_{0:n})}{\displaystyle\sum_{i=1}^{N} g_{0:n}(\xi^i_{0:n})}$$

for any function f depending on whole trajectory, where $(\xi^1_{0:n}, \cdots, \xi^N_{0:n})$ is an N–sample with common probability distribution $\eta_{0:n}$

in particular if $f = \phi \circ \pi$ depends only upon last state and not on whole sequence, then

$$\langle \gamma_n, \phi \rangle = \langle \gamma_{0:n}, \phi \circ \pi \rangle \approx \frac{1}{N} \sum_{i=1}^{N} g_{0:n}(\xi_{0:n}^i)\, \phi(\xi_n^i)$$

and

$$\langle \mu_n, \phi \rangle = \langle \mu_{0:n}, \phi \circ \pi \rangle \approx \frac{\displaystyle\sum_{i=1}^{N} g_{0:n}(\xi_{0:n}^i)\, \phi(\xi_n^i)}{\displaystyle\sum_{i=1}^{N} g_{0:n}(\xi_{0:n}^i)}$$

for any function ϕ, where $(\xi_{0:n}^1, \cdots, \xi_{0:n}^N)$ is an N–sample with common probability distribution $\eta_{0:n}$, and for $i = 1 \cdots N$

$\xi_n^i = \pi(\xi_{0:n}^i)$ denotes last state of sequence $\xi_{0:n}^i = (\xi_0^i, \cdots, \xi_n^i)$

in other words

$$\gamma_n \approx \gamma_n^N = \frac{1}{N} \sum_{i=1}^{N} g_{0:n}(\xi_{0:n}^i)\ \delta_{\xi_n^i}$$

and

$$\mu_n \approx \mu_n^N = \sum_{i=1}^{N} \frac{g_{0:n}(\xi_{0:n}^i)}{\sum_{j=1}^{N} g_{0:n}(\xi_{0:n}^j)}\ \delta_{\xi_n^i} = \sum_{i=1}^{N} w_n^i\ \delta_{\xi_n^i}$$

where nonnegative normalized weights $(w_n^1, \cdots, w_n^N)$ are defined for any $i = 1 \cdots N$ by

$$w_n^i = \frac{g_{0:n}(\xi_{0:n}^i)}{\sum_{j=1}^{N} g_{0:n}(\xi_{0:n}^j)}$$

SIS algorithm

importance sampling approximation : non–recursive *depth–first* implementation

simulate an N–sample of hidden state sequences $(\xi_{0:n}^1, \cdots, \xi_{0:n}^N)$:
independently for any $i = 1 \cdots N$, simulate a sequence $\xi_{0:n}^i = (\xi_0^i, \cdots, \xi_n^i)$, i.e.

- simulate a r.v. ξ_0^i according to $\eta_0(dx)$
- for any $k = 1 \cdots n$
  simulate a r.v. ξ_k^i according to $Q_k(\xi_{k-1}^i, dx')$

and define for any $i = 1 \cdots N$

$$g_{0:n}(\xi_{0:n}^i) = \prod_{k=0}^{n} g_k(\xi_{k-1}^i, \xi_k^i) \qquad \text{and} \qquad w_n^i = \frac{g_{0:n}(\xi_{0:n}^i)}{\sum_{j=1}^{N} g_{0:n}(\xi_{0:n}^i)}$$

importance sampling approximation : non–recursive implementation for nonlinear and non Gaussian systems

simulate an N–sample of hidden state sequences $(\xi_{0:n}^1, \cdots, \xi_{0:n}^N)$:
independently for any $i = 1 \cdots N$, simulate a sequence $\xi_{0:n}^i = (\xi_0^i, \cdots, \xi_n^i)$, i.e.

- simulate a r.v. ξ_0^i according to $\eta_0(dx)$
- for any $k = 1 \cdots n$
  simulate a r.v. W_k^i according to $p_k^W(dw)$ and set $\xi_k^i = f_k(\xi_{k-1}^i, W_k^i)$

and define for any $i = 1 \cdots N$

$$g_{0:n}(\xi_{0:n}^i) = \prod_{k=0}^{n} q_k^V(Y_k - h_k(\xi_k^i)) \qquad \text{and} \qquad w_n^i = \frac{g_{0:n}(\xi_{0:n}^i)}{\sum_{j=1}^{N} g_{0:n}(\xi_{0:n}^i)}$$

recursive formulation of weights updating

for any $k = 1 \cdots n$ and for any $i = 1 \cdots N$

$$w_k^i = \frac{g_{0:k}(\xi_{0:k}^i)}{\sum_{j=1}^{N} g_{0:k}(\xi_{0:k}^j)} = \frac{g_{0:k-1}(\xi_{0:k-1}^i)\; g_k(\xi_{k-1}^i, \xi_k^i)}{\sum_{j=1}^{N} g_{0:k-1}(\xi_{0:k-1}^j)\; g_k(\xi_{k-1}^j, \xi_k^j)} = \frac{w_{k-1}^i\; g_k(\xi_{k-1}^i, \xi_k^i)}{\sum_{j=1}^{N} w_{k-1}^j\; g_k(\xi_{k-1}^j, \xi_k^j)}$$

benefit : allows *breadth–first* implementation

SIS algorithm (*sequential importance sampling*) : recursive implementation

- for $k = 0$, independently for any $i = 1 \cdots N$
  simulate a r.v. ξ_i^0 according to $\eta_0(dx)$, and define

$$w_0^i = \frac{g_0(\xi_0^i)}{\sum_{j=1}^{N} g_0(\xi_0^j)}$$

- for any $k = 1 \cdots n$, independently for any $i = 1 \cdots N$
  simulate a r.v. ξ_k^i according to $Q_k(\xi_{k-1}^i, dx')$, and update weight as

$$w_k^i = \frac{w_{k-1}^i \, g_k(\xi_{k-1}^i, \xi_k^i)}{\sum_{j=1}^{N} w_{k-1}^j \, g_k(\xi_{k-1}^j, \xi_k^j)}$$

SIS algorithm (*sequential importance sampling*) : recursive implementation for nonlinear and non Gaussian systems

- for $k = 0$, independently for any $i = 1 \cdots N$
  simulate a r.v. ξ_i^0 according to $\eta_0(dx)$, and define

$$w_0^i = \frac{q_0^V(Y_0 - h_0(\xi_0^i))}{\sum_{j=1}^{N} q_0^V(Y_0 - h_0(\xi_0^j))}$$

- for any $k = 1 \cdots n$, independently for any $i = 1 \cdots N$
  simulate a r.v. W_k^i according to $p_k^W(dw)$ and set $\xi_k^i = f_k(\xi_{k-1}^i, W_k^i)$, and update weight as

$$w_k^i = \frac{w_{k-1}^i \, q_k^V(Y_k - h_k(\xi_k^i))}{\sum_{j=1}^{N} w_{k-1}^j \, q_k^V(Y_k - h_k(\xi_k^j))}$$

pros : higher weights are allocated to simulated sequences that are often consistent with observations

cons : weights are evaluated afterwards, and do not have impact on how sequences are simulated (blind simulation strategy) + along a given sequence, weights are accumulated in a multiplicative way

- *weights degeneracy* : in practice, one single sequence receives a much larger weight than all other sequences, whose contributions are therefore negligible
- *memory effect* : a sequence cannot be consistent with all observations

  a sequence that is consistent (resp. inconsistent) with current observation, but inconsistent (resp. consistent) with earlier observations, will receive a small (resp. a large) weight

proposed solutions

- use observations to guide how sequences are simulated
- from time to time, replicate / terminate sequences according to their respective weights

SIR algorithm

approximate Bayesian filter

$$\mu_n(dx) = \mathbb{P}[X_n \in dx \mid Y_{0:n}]$$

using recursive formulation

$$\mu_{k-1} \xrightarrow{\text{prediction}} \eta_k = \mu_{k-1} Q_k \xrightarrow{\text{correction}} \mu_k = g_k \cdot \eta_k$$

with initial condition $\mu_0 = g_0 \cdot \eta_0$

idea : look for approximations in the form of (possibly weighted) empirical probability distributions

$$\eta_k \approx \eta_k^N = \sum_{i=1}^N v_k^i \, \delta_{\xi_k^i} \qquad \text{et} \qquad \mu_k \approx \mu_k^N = \sum_{i=1}^N w_k^i \, \delta_{\xi_k^i}$$

associated with population of N particles characterized by

- positions $(\xi_k^1, \cdots, \xi_k^N)$ in E
- nonnegative normalized weights $(v_k^1, \cdots, v_k^N)$ and $(w_k^1, \cdots, w_k^N)$

initial approximation : using importance sampling

$$\mu_0 = g_0 \cdot \eta_0 \approx g_0 \cdot S^N(\eta_0) = \sum_{i=1}^{N} \frac{g_0(\xi_0^i)}{\sum_{j=1}^{N} g_0(\xi_0^j)} \, \delta_{\xi_0^i} = \sum_{i=1}^{N} w_0^i \, \delta_{\xi_0^i}$$

where variables $(\xi_0^1, \cdots, \xi_0^N)$ are i.i.d. with common probability distribution η_0

correction step : clearly, from definition

$$\mu_k^N = g_k \cdot \eta_k^N = \sum_{i=1}^{N} \frac{v_k^i \, g_k(\xi_k^i)}{\sum_{j=1}^{N} v_k^j \, g_k(\xi_k^j)} \, \delta_{\xi_k^i} = \sum_{i=1}^{N} w_k^i \, \delta_{\xi_k^i}$$

which automatically has desired form

prediction step : from definition

$$\langle \mu_{k-1}^N Q_k, \phi \rangle = \int \mu_{k-1}^N(dx) \int Q_k(x, dx')\, \phi(x')$$

$$= \sum_{i=1}^N w_{k-1}^i \int Q_k(\xi_{k-1}^i, dx')\, \phi(x')$$

$$= \int [\sum_{i=1}^N w_{k-1}^i\, Q_k(\xi_{k-1}^i, dx')\,]\, \phi(x')$$

for any function ϕ, hence

$$\mu_{k-1}^N Q_k = \sum_{i=1}^N w_{k-1}^i\, m_k^i$$

in form of a finite mixture, with

$$m_k^i(dx') = Q_k(\xi_{k-1}^i, dx') \qquad \text{for any } i = 1 \cdots N$$

requires further approximation (several sampling schemes available)

▶ multinomial resampling

simulate an N–sample $(\xi_k^1, \cdots, \xi_k^N)$ according to $\mu_{k-1}^N Q_k$, and set

$$\mu_{k-1}^N Q_k \approx \eta_k^N = S^N(\mu_{k-1}^N Q_k) = \frac{1}{N} \sum_{i=1}^N \delta_{\xi_k^i} = \sum_{i=1}^N v_k^i \, \delta_{\xi_k^i}$$

with $v_k^i = 1/N$ for any $i = 1 \cdots N$

weights are used to select (without replacement) mixture components with higher weights, with expected consequence that

- components with higher weights are selected several times
- conversely, components with lower weights are possibly discarded and will not further contribute to approximation

if R_i denotes how many times i–th mixture component has been selected, or equivalently how many samples in new approximation originate from i–th mixture component, for any $i = 1 \cdots N$, then

r.v. $(R_1, \cdots, R_N)$ has a multinomial distribution

intuitively, if all mixture weights are equal (or close) to $1/N$, i.e. if distribution of mixture weights is close to equidistribution, then selecting mixture components could be counter–productive

▶ weigths preservation

simulate one individual exactly from each mixture component and preserve its weight, i.e. independently for any $i = 1 \cdots N$ simulate ξ_k^i according to $m_k^i(dx') = Q_k(\xi_{k-1}^i, dx')$ and set

$$\mu_{k-1}^N Q_k \approx \eta_k^N = \sum_{i=1}^N w_{k-1}^i \, \delta_{\xi_k^i} = \sum_{i=1}^N v_k^i \, \delta_{\xi_k^i}$$

with $v_k^i = w_{k-1}^i$ for any $i = 1 \cdots N$

intuitively, this approach is appropriate if distribution of mixture weights is close to equidistribution, and less appropriate in extreme case where most weights are zero, except a few components with positive weights

SIR algorithm (*sampling with importance resampling*) : recursive implementation

- for $k = 0$, independently for any $i = 1 \cdots N$
  simulate a r.v. ξ_0^i according to $\eta_0(dx)$, and define

$$w_0^i = \frac{g_0(\xi_0^i)}{\sum_{j=1}^{N} g_0(\xi_0^j)}$$

- for any $k = 1 \cdots n$, independently for any $i = 1 \cdots N$
  - select an individual $\widehat{\xi}_{k-1}^i$ among population $(\xi_{k-1}^1, \cdots, \xi_{k-1}^N)$ and according to weights $(w_{k-1}^1, \cdots, w_{k-1}^N)$
  - simulate a r.v. ξ_k^i according to $Q_k(\widehat{\xi}_{k-1}^i, dx')$

  and define

$$w_k^i = \frac{g_k(\xi_{k-1}^i, \xi_k^i)}{\sum_{j=1}^{N} g_k(\xi_{k-1}^j, \xi_k^j)}$$

SIR algorithm (*sampling with importance resampling*) : recursive formulation for nonlinear and non Gaussian systems

- for $k = 0$, independently for any $i = 1 \cdots N$
  simulate a r.v. ξ_0^i according to $\eta_0(dx)$, and define

$$w_0^i = \frac{q_0^V(Y_0 - h_0(\xi_0^i))}{\sum_{j=1}^{N} q_0^V(Y_0 - h_0(\xi_0^j))}$$

- for any $k = 1 \cdots n$, independently for any $i = 1 \cdots N$
  - select and individual $\widehat{\xi}_{k-1}^i$ among population $(\xi_{k-1}^1, \cdots, \xi_{k-1}^N)$ and according to weights $(w_{k-1}^1, \cdots, w_{k-1}^N)$
  - simulate a r.v. W_k^i according to $p_k^W(dw)$ and set $\xi_k^i = f_k(\widehat{\xi}_{k-1}^i, W_k^i)$

  and define

$$w_k^i = \frac{q_k^V(Y_k - h_k(\xi_k^i))}{\sum_{j=1}^{N} q_k^V(Y_k - h_k(\xi_k^j))}$$

to summarize, particles $(\xi_{k-1}^1, \cdots, \xi_{k-1}^N)$

- *are selected* according to their respective weights $(w_{k-1}^1, \cdots, w_{k-1}^N)$ [*selection* step]
- *evolve* according to transition probabilities $Q_k(x, dx')$ [*mutation* step]
- and *are weighted* by evaluating likelihood function g_k [*weighting* step]

pros : weights do not accumulate along each sequence, but are used to select (or resample) particles

particles with larger (resp. smaller) weights are replicated (resp. are terminated)

by keeping only most probable particles at each time instant, expected benefit is to concentrate available computing power within regions of interest

cons : introduces additional randomness, in resampling (selection) step

proposed solutions

- alternate resampling strategies, that allocate an (almost) deterministic number of offsprings to each selected particle
- adaptive resampling, only when weights $(w_k^1, \cdots, w_k^N)$ are too much unbalanced (far from equidistribution)

cons : because of replication, fewer truly distinct positions are available (sample impoverishment)

- *positions degeneracy* : in practice, implicitly rely on mutation step to bring diversity again

proposed solution

- after resampling (selection) step, add some random move to each selected particle, or apply some artificial Markovian dynamics (Metropolis–Hastings, Gibbs sampling, etc.)

adaptive SIR algorithm

given a finite mixture

$$m = \sum_{i=1}^{N} w_i \, m_i$$

selecting mixture components is interesting only if weights $(w_1, \cdots, w_N)$ are far from equidistribution

several heuristic criteria have been proposed to quantify departure from equidistribution, and to decide wether particles should be resampled or not, e.g.

- effective sample size
- entropy

▶ χ^2 distance and effective sample size χ^2 distance between two probability vectors $p = (p_1, \cdots, p_N)$ and $q = (q_1, \cdots, q_N)$ is defined as

$$\chi^2(p, q) = \sum_{i=1}^{N} q_i \, (\frac{p_i}{q_i} - 1)^2$$

in particular for $p = w = (w_1, \cdots, w_N)$ and $q = (1/N, \cdots, 1/N)$, it holds

$$0 \leq \frac{1}{N} \sum_{i=1}^{N} (N\, w_i - 1)^2 = \frac{1}{N} \sum_{i=1}^{N} (N\, w_i)^2 - 1 = N \sum_{i=1}^{N} w_i^2 - 1$$

hence

$$1 \leq N_{\text{eff}} = 1 \;/\; [\sum_{i=1}^{N} w_i^2\,] \leq N$$

where equality is attained at *equidistribution*, which suggests to resample if

$$H(w_1, \cdots, w_N) = N \sum_{i=1}^{N} w_i^2 - 1 = \frac{N}{N_{\text{eff}}} - 1 \geq H_{\text{red}}$$

for some threshold $H_{\text{red}} > 0$ still to be fixed

on the way to asymptotic results (in 3 slides)

recall linear evolution for unnormalized version of Bayesian filter

$$\gamma_k = \gamma_{k-1} R_k = g_k \, (\gamma_{k-1} Q_k) = g_k \, (\mu_{k-1} Q_k) \, \langle \gamma_{k-1}, 1 \rangle = g_k \, \eta_k \, \langle \gamma_{k-1}, 1 \rangle$$

with initial condition $\gamma_0 = g_0 \, \eta_0$

proposed particle approximation for unnormalized distribution

$$\gamma_k^N = g_k \, \eta_k^N \, \langle \gamma_{k-1}^N, 1 \rangle$$

with initial condition $\gamma_0^N = g_0 \, \eta_0^N$ and $\eta_0^N = S^N(\eta_0)$: clearly

$$\langle \gamma_k^N, 1 \rangle = \langle \eta_k^N, g_k \rangle \, \langle \gamma_{k-1}^N, 1 \rangle \qquad \text{and} \qquad \langle \gamma_0^N, 1 \rangle = \langle \eta_0^N, g_0 \rangle$$

and it follows

$$\frac{\gamma_k^N}{\langle \gamma_k^N, 1 \rangle} = g_k \cdot \eta_k^N = \mu_k^N \qquad \text{and} \qquad \frac{\gamma_0^N}{\langle \gamma_0^N, 1 \rangle} = g_0 \cdot \eta_0^N = \mu_0^N$$

$\longrightarrow$ normalized version of proposed particle approximation for γ_k^N coincides with SIR bootstrap approximation μ_k^N for Bayesian filter

Remark key for induction : for any $k = 1 \cdots n$ and by difference

$$\gamma_k^N - \gamma_k = g_k \, \eta_k^N \, \langle \gamma_{k-1}^N, 1 \rangle - g_k \, (\gamma_{k-1} \, Q_k)$$

$$= g_k \, (\gamma_{k-1}^N \, Q_k - \gamma_{k-1} \, Q_k) + g_k \, (\eta_k^N - \mu_{k-1}^N \, Q_k) \, \langle \gamma_{k-1}^N, 1 \rangle$$

hence

$$\langle \gamma_k^N - \gamma_k, \phi \rangle = \langle \gamma_{k-1}^N - \gamma_{k-1}, Q_k(g_k \, \phi) \rangle + \langle \eta_k^N - \mu_{k-1}^N \, Q_k, g_k \, \phi \rangle \, \langle \gamma_{k-1}^N, 1 \rangle$$

error at current generation, evaluated on function ϕ, is decomposed into

- error at previous generation, evaluated on function $R_k \, \phi = Q_k(g_k \, \phi)$
- local error resulting from Monte Carlo approximation

even though samples are actually dependent, because of resampling

at each generation, *conditionally* on previous generations, new samples are generated independently

with this conditioning argument, error estimates

$$\sup_{\phi\,:\,\|\phi\|=1} \mathbb{E}\Big|\frac{\langle \gamma_k^N - \gamma_k, \phi\rangle}{\langle \gamma_k, 1\rangle}\Big| \leq \frac{c_k}{\sqrt{N}}$$

and

$$\sup_{\phi\,:\,\|\phi\|=1} \mathbb{E}\big|\langle \mu_k^N - \mu_k, \phi\rangle\big| \leq 2\,\frac{c_k}{\sqrt{N}}$$

of order $1/\sqrt{N}$, and CLT

$$\sqrt{N}\,\frac{\langle \gamma_k^N - \gamma_k, \phi\rangle}{\langle \gamma_k, 1\rangle} \Longrightarrow \mathcal{N}(0, V_k(\phi))$$

and

$$\sqrt{N}\,\langle \mu_k^N - \mu_k, \phi\rangle \Longrightarrow \mathcal{N}(0, v_k(\phi)) \qquad \text{with} \qquad v_k(\phi) = V_k(\phi - \langle \mu_k, \phi\rangle)$$

can be obtained by induction

Some algorithmic variants

- regularization
- progressive weighting, MCMC iterations
- sample–size adaptation
- marginalization aka Rao–Blackwellization
  - interacting Kalman filters
  - interacting finite state (Baum) filters

conditionning as a variance reduction technique

$$\mathbb{E}[f(X^1, X^2)] = \mathbb{E}[\,\mathbb{E}[f(X^1, X^2) \mid X^2]\,] = \mathbb{E}[F(X^2)]$$

if

$$F(x_2) = \mathbb{E}[f(X^1, X^2) \mid X^2 = x_2] = \int_{E_1} f(x_1, x_2)\, \mathbb{P}[X^1 \in dx_1 \mid X^2 = x_2]$$

has an explicit expression, then Monte Carlo estimator

$$\frac{1}{N} \sum_{i=1}^{N} F(X_i^2) \approx \mathbb{E}[F(X^2)] = \mathbb{E}[f(X^1, X^2)]$$

where $(X_1^2, \cdots, X_N^2)$ is an N–sample with same common distribution as X^2, has smaller variance than Monte Carlo estimator

$$\frac{1}{N} \sum_{i=1}^{N} f(X_i^1, X_i^2) \approx \mathbb{E}[f(X^1, X^2)]$$

where $((X_1^1, X_1^2), \cdots, (X_N^1, X_N^2)$ is an N–sample with same common distribution as (X^1, X^2)

1st example : conditionnally linear Gaussian systems

$$X_k^{\mathrm{L}} = F_k^{\mathrm{L}}(X_{k-1}^{\mathrm{NL}})\, X_{k-1}^{\mathrm{L}} + f_k^{\mathrm{L}}(X_{k-1}^{\mathrm{NL}}) + W_k^{\mathrm{L}}$$

$$X_k^{\mathrm{NL}} = F_k^{\mathrm{NL}}(X_{k-1}^{\mathrm{NL}})\, X_{k-1}^{\mathrm{L}} + f_k^{\mathrm{NL}}(X_{k-1}^{\mathrm{NL}}) + W_k^{\mathrm{NL}}$$

$$Y_k = h_k(X_k^{\mathrm{NL}}) + V_k$$

clearly

$$\mathbb{E}[\phi(X_n^{\mathrm{L}}, X_n^{\mathrm{NL}}) \prod_{k=0}^{n} g_k(X_k^{\mathrm{NL}})] = \mathbb{E}[\,\mathbb{E}[\phi(X_n^{\mathrm{L}}, X_n^{\mathrm{NL}}) \mid X_{0:n}^{\mathrm{NL}}] \prod_{k=0}^{n} g_k(X_k^{\mathrm{NL}})]$$

and conditional distribution of said linear component X_n^{L} given said nonlinear component sequence $X_{0:n}^{\mathrm{NL}}$, is Gaussian, with mean $\widehat{X}_k^{\mathrm{L}|\mathrm{NL}}$ and covariance matrix $P_k^{\mathrm{L}|\mathrm{NL}}$ given explicitly, in recursive form, by Kalman filter equation

introduce new hidden state $\{(X_k^{\mathrm{NL}}, \widehat{X}_k^{\mathrm{L}|\mathrm{NL}}, P_k^{\mathrm{L}|\mathrm{NL}})\}$ instead of $\{(X_k^{\mathrm{L}}, X_k^{\mathrm{NL}})\}$

benefit : explore with particles subspace associated with nonlinear components, and associated with each particle, a Kalman filter estimates linear components

2nd example : non–linear systems with Markovian switching regimes / modes

$$X_k \;=\; f_k(s_{k-1}, X_{k-1}, W_k)$$

$$Y_k \;=\; h_k(X_k) + V_k$$

where regime / mode sequence $\{s_k\}$ forms a Markov chain with finite state space

clearly

$$\mathbb{E}[\phi(s_n, X_n) \prod_{k=0}^{n} g_k(X_k)\,] = \mathbb{E}[\,\mathbb{E}[\phi(s_n, X_n) \mid X_{0:n}] \prod_{k=0}^{n} g_k(X_k)\,]$$

and conditional distribution of regime / mode s_n given continuous components sequence $X_{0:n}$, is a finite dimensional probability vector defined by

$$p_n^i = \mathbb{P}[s_n = i \mid X_{0:n}] \qquad \text{for any } i \in I$$

given explicitly, in recursive form, by solving Baum forward equation

introduce new hidden state $\{(X_k, p_k)\}$ instead of $\{(s_k, X_k)\}$

benefit : avoid sampling finite state space

Conclusion

particle filtering provides an implementation of Bayesian approach that is

- intuitive, easy to understand and implement
- flexible, adapts to many models, many algorithmic variants available
- numerically efficient, through some selection mechanism
- amenable to mathematical analysis