10: Introduction to Bayesian Computation

Taylor

University of Virginia

Introduction

This chapter gives an overview for how to approximate intractable quantities such as posterior expectations and predictions.

Definitions

Numerical integration methods approximate integrals.

These methods can be loosely categorized as either **stochastic** or **deterministic** (e.g. quadrature methods).

Deterministic Methods

Deterministic methods don't draw samples, and instead approximate the integral by summing up approximate volumes:

$$E[h(\theta) \mid y] = \int h(\theta) p(\theta \mid y) \mathrm{d}\theta \approx \frac{1}{S} \sum_{s=1}^{S} w_s h(\theta^s) p(\theta^s \mid y)$$

Stochastic Methods

Stochastic methods involve sample averages of simulated draws from some distribution. There are many ways to do this, but generally

$$E[h(\theta) \mid y] = \int h(\theta)p(\theta \mid y)\mathrm{d}\theta \approx \frac{1}{S}\sum_{s=1}^{S} h(\theta^s)$$

with $\theta^s \sim p(\theta \mid y)$, or

$$E[h(\tilde{y}) \mid y] = \int h(\tilde{y})p(\tilde{y} \mid y)\mathrm{d}\tilde{y} \approx \frac{1}{S}\sum_{s=1}^{S} h(\tilde{y}^s)$$

with $\tilde{y} \sim p(\tilde{y} \mid y)$

Stochastic Methods

Drawing $\tilde{y}$ samples can be done in a two-stage way:

1. draw $\theta^s \sim p(\theta \mid y)$
2. draw $\tilde{y}^s \sim p(\tilde{y} \mid \theta^s)$

Stochastic Methods

Drawing $\tilde{y}$ samples can be done in a two-stage way:

1. draw $\theta^s \sim p(\theta \mid y)$
2. draw $\tilde{y}^s \sim p(\tilde{y} \mid \theta^s)$

If you can derive $E\left(h(\tilde{y}) \mid \theta\right)$, you should probably use a Rao-Blackwellized procedure:

$$E[h(\tilde{y}) \mid y] = E[E\left(h(\tilde{y}) \mid \theta\right) \mid y] \approx \frac{1}{S}\sum_{s=1}^{S} E\left(h(\tilde{y}) \mid \theta^s\right)$$

with $\theta^s \sim p(\theta \mid y)$

Approximating the posterior on a grid

We used this method earlier for the rat tumor example!

Given that we can evaluate the unnormalized target $p(\theta, y) = p(y \mid \theta)p(\theta)$, we choose a nonrandom grid of points (think `seq`) $\theta_1, \ldots, \theta_S$, and then we approximate the the continuous posterior with a discrete random variable with pmf equal to

$$\tilde{p}(\theta_j \mid y) = \frac{p(y \mid \theta_j)p(\theta_j)}{\sum_{s=1}^{S} p(y \mid \theta_s)p(\theta_s)}$$

for any $\theta_j \in \{\theta_1, \ldots, \theta_S\}$

The general setup

From now on we will write the posterior in terms of an unnormalized density $q(\theta \mid y)$. In other words:

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

Most (maybe all) of the sampling techniques will assume that we can't evaluate $p(\theta \mid y)$, but that we can evaluate $q(\theta \mid y)$

Rejection Sampling aka Accept-Reject sampling

Setup

1. $p(\theta \mid y)$ the target, posterior
2. $q(\theta \mid y) = p(y \mid \theta)p(\theta)$ the unnormalized target
3. $g(\theta)$ the "instrumental" or "proposal" distribution
4. need $q(\theta \mid y)/g(\theta) \leq M$ uniformly
5. need $g \gg q$ i.e. the proposal "dominates" your target (won't divide by 0)

We are free to choose our own $g(\theta)$. For the time being, we assume that $\int g(\theta) d\theta = 1$.

Rejection Sampling aka Accept-Reject sampling

To (potentially) produce one draw:

1. propose the draw $\theta^s \sim g(\theta)$
2. accept θ^s with probability $\frac{q(\theta^s|y)}{g(\theta^s)M}$

Rejection Sampling aka Accept-Reject sampling

To (potentially) produce one draw:

1. propose the draw $\theta^s \sim g(\theta)$
2. accept θ^s with probability $\frac{q(\theta^s|y)}{g(\theta^s)M}$

Note this is the same as

1. propose the draw $\theta^s \sim g(\theta)$
2. draw $U \sim \text{Uniform}(0, 1]$
3. accept θ^s if $U < q(\theta^s \mid y)/\{g(\theta^s)M\}$

Proof

$$
\begin{aligned}
P\left(\theta \le t \,\middle|\, U \le \frac{q(\theta \mid y)}{Mg(\theta)}\right) &= \frac{P\left(\theta \le t, U \le \frac{q(\theta|y)}{Mg(\theta)}\right)}{P\left(U \le \frac{q(\theta|y)}{Mg(\theta)}\right)} \\
&= \frac{\int_{-\infty}^{t} \int_{0}^{\frac{q(\theta|y)}{Mg(\theta)}} g(\theta) 1 \mathrm{d}u \mathrm{d}\theta}{\int_{-\infty}^{\infty} \int_{0}^{\frac{q(\theta|y)}{Mg(\theta)}} g(\theta) 1 \mathrm{d}u \mathrm{d}\theta} \\
&= \frac{\int_{-\infty}^{t} g(\theta) \frac{q(\theta|y)}{Mg(\theta)} \mathrm{d}\theta}{\int_{-\infty}^{\infty} g(\theta) \frac{q(\theta|y)}{Mg(\theta)} \mathrm{d}\theta} \\
&= \frac{\int_{-\infty}^{t} q(\theta \mid y) \mathrm{d}\theta}{\int_{-\infty}^{\infty} q(\theta \mid y) \mathrm{d}\theta} \\
&= P(\theta \le t \mid y).
\end{aligned}
$$

Example 1

Assume $y \sim \text{Normal}(\theta, 1)$, and $p(\theta) = \frac{1}{\pi(1+\theta^2)}$. Our goal is to draw from

$$
\begin{aligned}
p(\theta \mid y) &\propto q(\theta \mid y) \\
&= p(y \mid \theta) p(\theta) \\
&= \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(y-\theta)^2\right] \frac{1}{\pi(1+\theta^2)} \\
&\propto \exp\left[-\frac{(\theta-y)^2}{2} - \log(1+\theta^2)\right],
\end{aligned}
$$

Example 1

Let's assume that we want to use our prior distribution as a proposal: $g(\theta) = p(\theta)$. Then we have to find M:

$$\begin{aligned}
\frac{q(\theta \mid y)}{g(\theta)} &= \frac{p(y \mid \theta)p(\theta)}{p(\theta)} \\
&= p(y \mid \theta) \\
&= \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(y-\theta)^2\right] \\
&\leq \frac{1}{\sqrt{2\pi}} \overset{\text{def}}{=} M
\end{aligned}$$

Our acceptance probability for draw θ^s is then

$$q(\theta^s \mid y)/\{g(\theta^s)M\} = p(y \mid \theta^s)/M = \exp\left[-\frac{1}{2}(y-\theta^s)^2\right]$$

Rejection Sampling aka Accept-Reject sampling

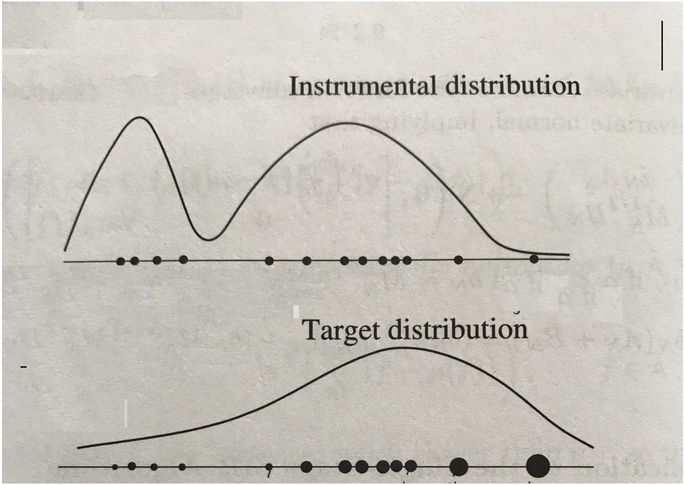
Generally a good strategy is work with logarithms, and then exponentiate as late as possible.

```
y <- 2 # fake data
num_trials <- 1000
theta_proposals <- rt(num_trials, 1)
us <- runif(num_trials, min = 0, max = 1)
log_accept_prob <- function(theta){
  -.5*(y - theta)^2
}
probs <- exp(log_accept_prob(theta_proposals))
accepts <- ifelse(us < probs, TRUE, FALSE)
hist(theta_proposals[accepts]) # only the accepted draws
#hist(theta_proposals) # all draws!
```

Importance Sampling

importance sampling also involves ratios like the previous algorithm. However, instead of using those ratios to either accept or discard samples, it uses the ratios to weight samples.

Importance Sampling



Original, unedited image is from
`https://www.springer.com/us/book/9780387402642`

Importance Sampling

Setup

1. $p(\theta \mid y)$ the target, posterior
2. $q(\theta \mid y) = p(y \mid \theta)p(\theta)$ the unnormalized target
3. $g(\theta)$ the "instrumental" or "proposal" distribution
4. $g \gg q$ i.e. the proposal dominates your target

We are free to choose our own $g(\theta)$. For the time being, we assume that $\int g(\theta)\mathrm{d}\theta = 1$.

Importance Sampling

Algorithm: for each iteration s

1. draw $\theta^s \sim g(\theta)$
2. calculate unnormalized weight $\tilde{w}(\theta^s) = \frac{q(\theta^s|y)}{g(\theta^s)}$
3. calculate normalized weights $w(\theta^s) = \tilde{w}(\theta^s) / \sum_r \tilde{w}(\theta^r)$

Final calculation:

$$E_q[h(\theta) \mid y] \approx \sum_s w(\theta^s) h(\theta^s)$$

Importance Sampling

Motivation:

$$
\begin{aligned}
E_q[h(\theta) \mid y] &= \int h(\theta) p(\theta \mid y) \mathrm{d}\theta \\
&= \frac{\int h(\theta) q(\theta \mid y) \mathrm{d}\theta}{\int q(\theta \mid y) \mathrm{d}\theta} \\
&= \frac{\int h(\theta) \frac{q(\theta|y)}{g(\theta)} g(\theta) \mathrm{d}\theta}{\int \frac{q(\theta|y)}{g(\theta)} g(\theta) \mathrm{d}\theta}
\end{aligned}
$$

Importance Sampling

Motivation:

$$
\begin{aligned}
E_q[h(\theta) \mid y] &= \int h(\theta) p(\theta \mid y) \mathrm{d}\theta \\
&= \frac{\int h(\theta) q(\theta \mid y) \mathrm{d}\theta}{\int q(\theta \mid y) \mathrm{d}\theta} \\
&= \frac{\int h(\theta) \frac{q(\theta \mid y)}{g(\theta)} g(\theta) \mathrm{d}\theta}{\int \frac{q(\theta \mid y)}{g(\theta)} g(\theta) \mathrm{d}\theta}
\end{aligned}
$$

So first:

$$
\frac{1}{S} \sum_{s=1}^{S} \frac{q(\theta^s \mid y)}{g(\theta^s)} \to E_g\left[\frac{q(\theta \mid y)}{g(\theta)}\right] = \int \frac{q(\theta \mid y)}{g(\theta)} g(\theta) \mathrm{d}\theta = \int q(\theta \mid y) \mathrm{d}\theta
$$

Importance Sampling

1. $E_q[h(\theta) \mid y] = \frac{\int h(\theta)q(\theta|y)\mathrm{d}\theta}{\int q(\theta|y)\mathrm{d}\theta}$
2. $\frac{1}{S}\sum_{s=1}^{S} \frac{q(\theta^s|y)}{g(\theta^s)} \to \int q(\theta \mid y)\mathrm{d}\theta$ (for the denominator)

And second:

$$\frac{1}{S}\sum_{s=1}^{S} h(\theta^s)\frac{q(\theta^s \mid y)}{g(\theta^s)} \to E_g\left[h(\theta)\frac{q(\theta \mid y)}{g(\theta)}\right] = \int h(\theta)q(\theta \mid y)\mathrm{d}\theta$$

which converges to the numerator

Importance Sampling

1. $E_q[h(\theta) \mid y] = \frac{\int h(\theta) q(\theta|y) \mathrm{d}\theta}{\int q(\theta|y) \mathrm{d}\theta}$
2. $\frac{1}{S} \sum_{s=1}^{S} \frac{q(\theta^s|y)}{g(\theta^s)} \to \int q(\theta \mid y) \mathrm{d}\theta$
3. $\frac{1}{S} \sum_{s=1}^{S} h(\theta^s) \frac{q(\theta^s|y)}{g(\theta^s)} \to \int h(\theta) q(\theta \mid y) \mathrm{d}\theta$

So finally

$$\sum_{i=1}^{S} w(\theta^s) h(\theta^s) = \frac{\sum_{s=1}^{S} h(\theta^s) \frac{q(\theta^s|y)}{g(\theta^s)}}{\sum_{r=1}^{S} \frac{q(\theta^r|y)}{g(\theta^r)}} = \frac{\frac{1}{S} \sum_{s=1}^{S} h(\theta^s) \frac{q(\theta^s|y)}{g(\theta^s)}}{\frac{1}{S} \sum_{r=1}^{S} \frac{q(\theta^r|y)}{g(\theta^r)}} \to E[h(\theta) \mid y]$$

where $w(\theta^s) = \frac{q(\theta^s|y)}{g(\theta^s)} \Big/ \sum_{r=1}^{S} \frac{q(\theta^r|y)}{g(\theta^r)}$ are the self-normalized weights

Example 2

Assume $y \sim \text{Normal}(\theta, 1)$, and $p(\theta) = \frac{1}{\pi(1+\theta^2)}$. Approximate $E_q[\theta \mid y]$ using proposal $g(\theta) = p(\theta)$.

Example 2

Assume $y \sim \text{Normal}(\theta, 1)$, and $p(\theta) = \frac{1}{\pi(1+\theta^2)}$. Approximate $E_q[\theta \mid y]$ using proposal $g(\theta) = p(\theta)$.

If we sample from $g(\theta) = p(\theta) = \frac{1}{\pi(1+\theta^2)}$ then the unnormalized weights are

$$\begin{aligned}\tilde{w}(\theta^s) &= \frac{q(\theta^s \mid y)}{g(\theta^s)} \\ &= p(y \mid \theta^s) \\ &= \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(y - \theta^s)^2\right]\end{aligned}$$

Example 2

Assume $y \sim \text{Normal}(\theta, 1)$, and $p(\theta) = \frac{1}{\pi(1+\theta^2)}$. Approximate $E_q[\theta \mid y]$ using proposal $g(\theta) = p(\theta)$.

If we sample from $g(\theta) = p(\theta) = \frac{1}{\pi(1+\theta^2)}$ then the unnormalized weights are

$$\begin{aligned}\tilde{w}(\theta^s) &= \frac{q(\theta^s \mid y)}{g(\theta^s)} \\ &= p(y \mid \theta^s) \\ &= \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2}(y - \theta^s)^2\right]\end{aligned}$$

then normalize these...

Example 2

```
y <- 2 # fake data
num_samples <- 1000000
theta_draws <- rt(num_samples , 1)
log_unnorm_weight <- function(theta){
  # sqrt(2pi) because it will cancel out
  -.5*(y - theta)^2
}
lunws <- log_unnorm_weight(theta_draws)
norm_weights <- exp(lunws)/sum(exp(lunws))
sum(norm_weights * theta_draws)
hist(norm_weights)
```

Example 2

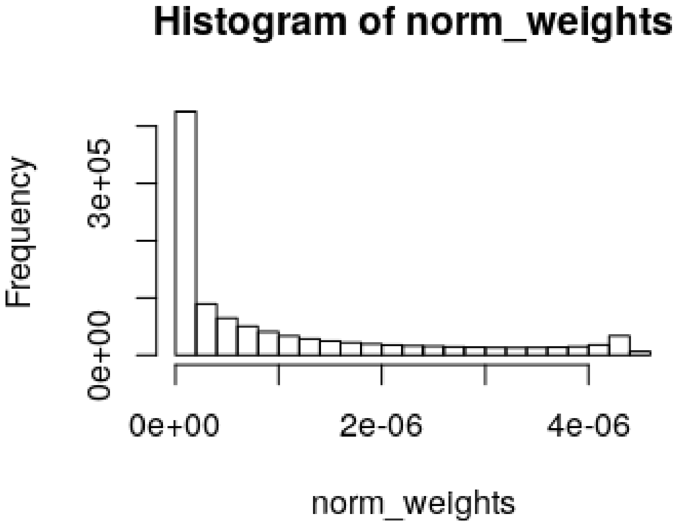
The choice of proposal is very important:

Using the Delta method:

$$\mathrm{Var}_g\left(\sum_{s=1}^{S} w(\theta^s)h(\theta^s)\right) \approx \frac{1}{S}E_g\left[\underbrace{\left(\frac{\tilde{w}(\theta)}{E_g[\tilde{w}(\theta)]}\right)^2}_{!}(h(\theta) - E_q[h(\theta)])^2\right]$$

Details:
`https://stats.stackexchange.com/questions/250934/var-self-normalised-sampling-estimator/250972#250972`

Example 2


Histogram of norm_weights
Frequency
0e+00
3e+05
0e+00
2e-06
4e-06
norm_weights

Example 2

Beware of bad proposal distributions!

A sample estimator of this approximate variance is

$$\sum_{s=1}^{S} w(\theta^s) \left(h(\theta^s) - \hat{E}[h(\theta)] \right)^2$$

where $\hat{E}[h(\theta)] = \sum_s w(\theta^s) h(\theta^s)$.

Note the weights aren't uniform like a "standard" estimation of the sample variance.

Example 2

```
y <- 2 # fake data
log_unnorm_weight <- function(theta){
  # sqrt(2pi) because it will cancel out
  -.5*(y - theta)^2 }
getISEstimator <- function(num_samples){
  theta_draws <- rt(num_samples , 1)
  lunws <- log_unnorm_weight(theta_draws)
  norm_weights <- exp(lunws)/sum(exp(lunws))
  estimator <- sum(norm_weights * theta_draws)
  list("estimate" = estimator,
       "approx_var" = sum( norm_weights*(theta_draws - estima
# two ways to calculate standard errors
num_samps_per_estimate <- 10
sqrt(getISEstimator(num_samps_per_estimate)$approx_var)
sd(replicate(1000,
    getISEstimator(num_samps_per_estimate)$estimate))
```

Effective Sample Size

$$\sum_{s=1}^{S} w(\theta^s)\left(h(\theta^s) - \hat{E}[h(\theta)]\right)^2 \stackrel{\text{set}}{=} \frac{1}{S_{\text{eff}}} \text{Var}\left[h(\theta^i)\right]$$

Solving for S_{eff} yields
TODO

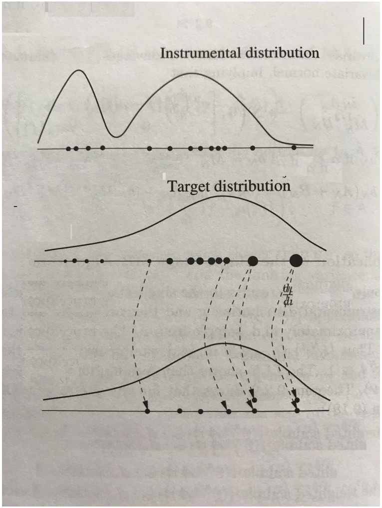
Adding Resampling

Importance Sampling gives you weighted draws $(\theta^1, w(\theta^1)), (\theta^2, w(\theta^2)), \ldots$

You can draw from these, with replacement. At the expense of more variance, it will give you unweighted draws from your target distribution: $\tilde{\theta}^1, \tilde{\theta}^2, \ldots$

This is known as **factored sampling** or **importance sampling with resampling** or **sampling importance resampling** (SIR).

Adding Resampling



Original, unedited image is from
`https://www.springer.com/us/book/9780387402642`

SIR

Stage 1: do importance sampling to get $\{\theta^i, w(\theta^i)\}_{i=1}^S$.

Stage 2: for $i = 1, \ldots, S$, select

$$P[\tilde{\theta}^i = \theta^j \mid \theta^1, w(\theta^1), \ldots, \theta^S, w(\theta^S)] = \frac{w(\theta^j)}{\sum_k w(\theta^k)}.$$

SIR

Another way to write it:

Stage 1: do importance sampling to get $\{\theta^i, w(\theta^i)\}_{i=1}^S$.

Stage 2: for $i = 1, \ldots, S$, select indexes

$$P[I^i = j \mid \theta^1, w(\theta^1), \ldots, \theta^S, w(\theta^S)] = \frac{w(\theta^j)}{\sum_k w(\theta^k)}$$

and then set

$$\tilde{\theta}^i = \theta^{I^i}.$$

Example 2 revisited

```
y <- 2 # fake data
log_unnorm_weight <- function(theta){
  # sqrt(2pi) because it will cancel out
  -.5*(y - theta)^2 }
num_samples <- 10000
theta_draws <- rt(num_samples , 1)
lunws <- log_unnorm_weight(theta_draws)
# note: prob arg automatically normalizes
random_indexes <- sample(x = num_samples,
                         size = num_samples,
                         replace = T,
                         prob = exp(lunws))
sort(random_indexes) # repeats!
resampled_draws <- theta_draws[random_indexes]
hist(resampled_draws) # can't do this unless we resample
```

Going sequential

Resampling adds variance, so why do it?

It throws away bad samples, and duplicates promising ones. When you're looking at a sequence of distribution targets, this can have a good effect on future samples.

Going sequential

Resampling adds variance, so why do it?

It throws away bad samples, and duplicates promising ones. When you're looking at a sequence of distribution targets, this can have a good effect on future samples.

sequential monte carlo or **particle filtering** methods are basically doing SIR over and over again.

Examples of sequences of distributions

Data annealing[1]

$$p(\theta), p(\theta \mid y_1), p(\theta \mid y_{1:2}), \ldots, p(\theta \mid y_{1:n}),$$

Temperature annealing[2]

$$p(y \mid \theta)^{a_0} p(\theta), p(y \mid \theta)^{a_1} p(\theta), \ldots p(y \mid \theta)^{a_n} p(\theta)$$

with $0 = a_0 < a_1 < \cdots < a_n = 1$.

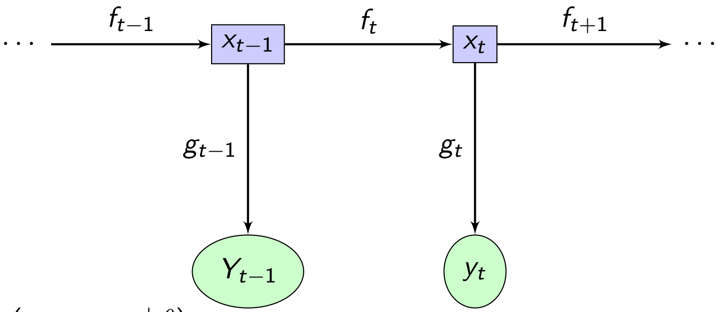
filtering and smoothing in state space models

$$p(x_1 \mid y_1, \theta), \ldots, p(x_n \mid y_{1:n}, \theta)$$

[1] Chopin, *A Sequential Particle Filter Method for Static Models*.

[2] Neal, "Annealed Importance Sampling".

state space models



$$
\begin{aligned}
&p(x_{1:T}, y_{1:T} \mid \theta) \\
&= g(y_1 \mid x_1, \theta) f(x_1 \mid \theta) \prod_{t=2}^{T} g(y_t \mid x_t, \theta) f(x_t \mid x_{t-1}, \theta)
\end{aligned}
$$

Example: filtering in state space models

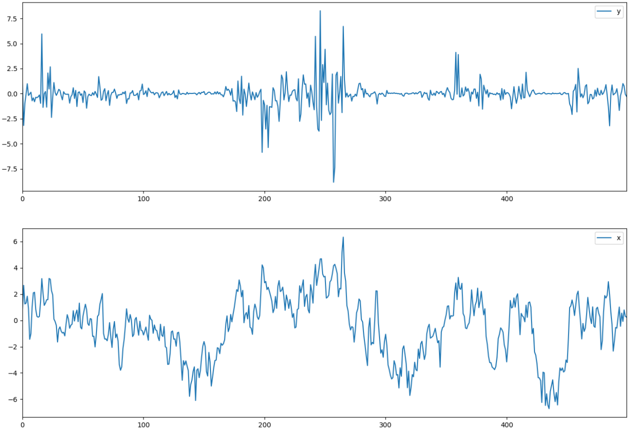
Here's an example of a state space model. y_t is a univariate time series, and x_t is a hidden/unobserved/latent time series.

$$y_t = \exp(x_t/2)\epsilon_t \tag{1}$$
$$x_t = c + \phi x_{t-1} + v_t \tag{2}$$

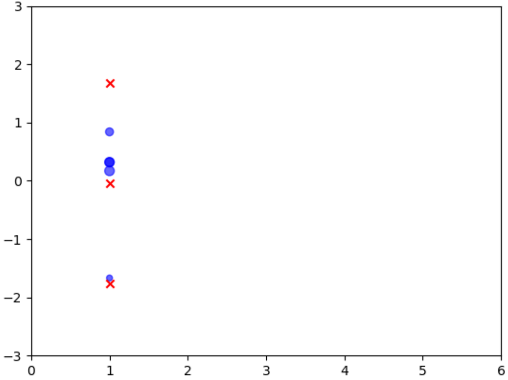
We sometimes refer to (1) as $g(y_t \mid x_t)$ or the observation equation, and (2) as the state transition equation or $f(x_t \mid x_{t-1})$.

Example: filtering in state space models

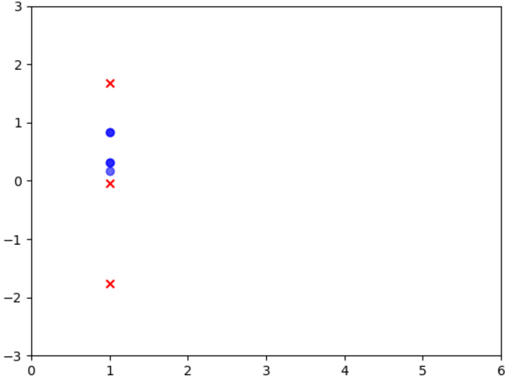
$y_{1:t}$ observed, $x_{1:t}$ hidden. Goal: $p(x_t \mid y_{1:t})$ in real-time.



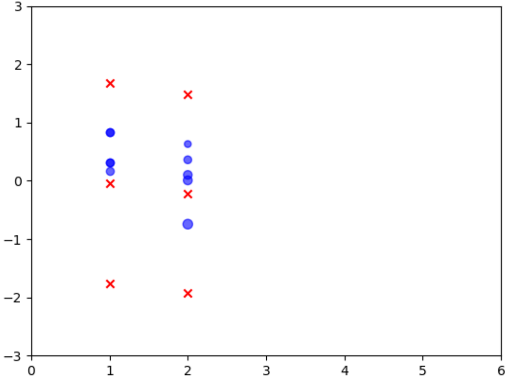
Example: filtering in state space models



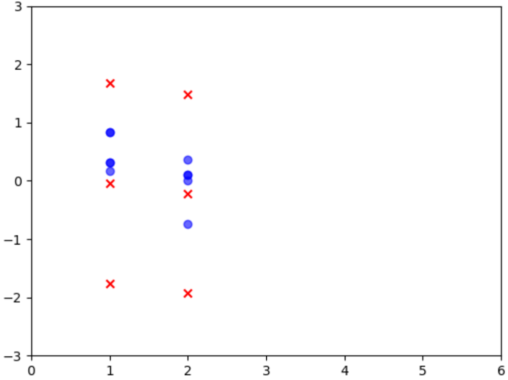
Going sequential



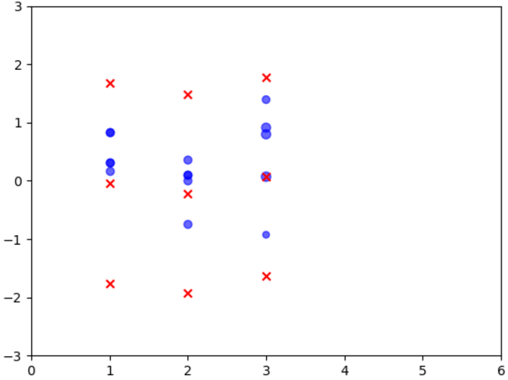
Going sequential



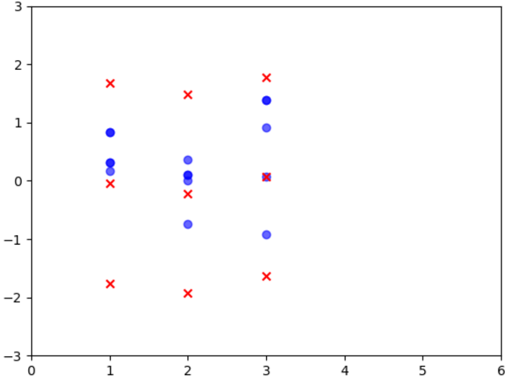
Going sequential



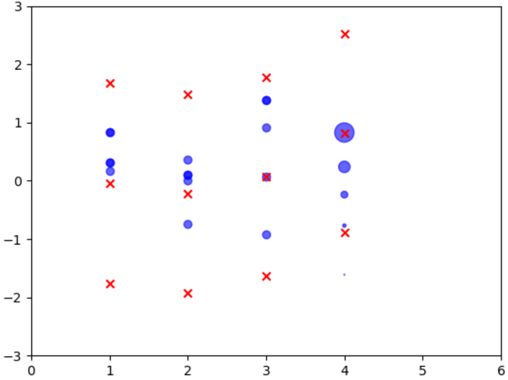
Going sequential



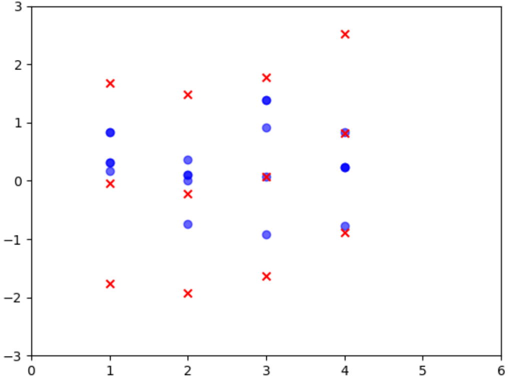
Going sequential



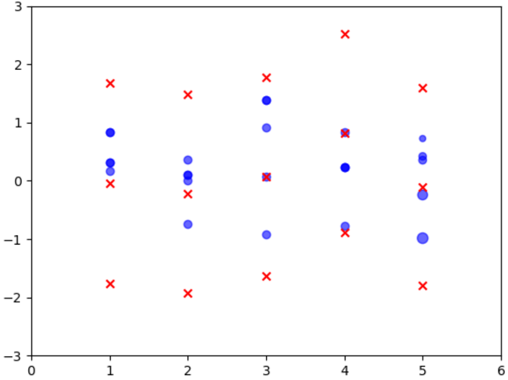
Going sequential



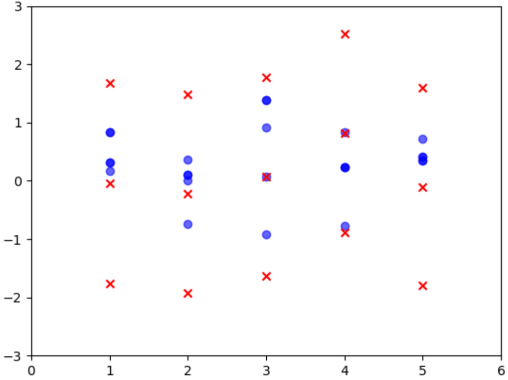
Going sequential



Going sequential



Going sequential



Filtering recursions

$$
\begin{aligned}
p(x_{1:t}|y_{1:t}) &= C_t^{-1} p(x_t, y_t \mid x_{t-1}) p(x_{1:t-1} \mid y_{1:t-1}) \\
&= C_t^{-1} \frac{p(x_t, y_t \mid x_{t-1})}{q_t(x_t \mid y_t, x_{t-1})} \times \\
&\qquad q_t(x_t \mid y_t, x_{t-1}) p(x_{1:t-1} \mid y_{1:t-1}) \\
&= C_t^{-1} \frac{g(y_t|x_t) f(x_t|x_{t-1})}{q_t(x_t|x_{t-1}, y_t)} \times \\
&\qquad q_t(x_t \mid x_{t-1}, y_t) p(x_{1:t-1} \mid y_{1:t-1})
\end{aligned}
$$

Repeat through time:

1. start with samples from $p(x_{1:t-1} \mid y_{1:t-1})$
2. mutate/propogate/extend using $q_t(x_t \mid x_{t-1}, y_t)$
3. adjust weights by multiplying by $\frac{g(y_t|x_t) f(x_t|x_{t-1})}{q_t(x_t|x_{t-1}, y_t)}$
4. resample, giving you particles distributed as $p(x_{1:t} \mid y_{1:t})$

Chopin, Nicolas. *A Sequential Particle Filter Method for Static Models*. 2000.

Neal, Radford M. "Annealed Importance Sampling". In: *Statistics and Computing* 11.2 (Apr. 2001), pp. 125–139. ISSN: 0960-3174. DOI: 10.1023/A:1008923215028. URL: https://doi.org/10.1023/A:1008923215028.