STAT232B
Importance and Sequential Importance Sampling

Gianfranco Doretto

Andrea Vedaldi

June 7, 2004

1 Monte Carlo Integration

- **Goal:** computing the following integral

$$\mu = \int_{\chi} h(x)\pi(x)\, dx$$

- **Standard numerical methods:** discretize the domain χ by regular grid, evaluate $h(x)\pi(x)$, and then use the Riemann sum as approximation

- **Monte Carlo integration:** consider $\mu = E_{\pi}[h(X)]$, $X \sim \pi$, draw m samples $x^{(1)}, \ldots, x^{(m)}$ from π, and compute the **Monte Carlo estimate**:

$$\hat{\hat{\mu}} = \frac{1}{m} \sum_{i=1}^{m} h(x^{(i)})$$

2 Monte Carlo Integration: properties of the estimator

- **Rate of convergence:** by the law of large numbers we have

$$\hat{\hat{\mu}} \longrightarrow \mu \text{ in } O(m^{-1/2})$$

- **Unbiasedness:**

$$E[\hat{\hat{\mu}}] = E[\frac{1}{m}\sum_{i=1}^{m} h(X^{(i)})] = E_{\pi}[h(X)]$$

- **Variance:**

$$\begin{aligned} \text{var}(\hat{\hat{\mu}}) &= \text{var}(\frac{1}{m}\sum_{i=1}^{m} h(X^{(i)})) = \frac{1}{m}\text{var}(h(X)) \\ &= \frac{1}{m}\int_{\chi}(h(x)-\mu)^2\pi(x)dx \end{aligned}$$

3 Monte Carlo Integration: can we do better?

- **Rate of convergence:** sorry, this is the best we can do!
- **Unbiasedness:** hey, you cannot ask more than this!
- **Variance:** sure there is something we can do: **reducing it**!
  - is sampling from π the best thing we can do?
  - what if I do not know how to sample from π?

USE IMPORTANCE SAMPLING!

4 Importance sampling

- **Idea:** consider $X \sim g$, such that $\int_\chi g(x)dx = 1$, and $g(x) \neq 0 \; \forall x \in \chi$, so that

$$\mu = \int_\chi h(x)\pi(x)dx = \int_\chi \frac{h(x)\pi(x)}{g(x)} g(x)dx = E_g[\frac{h(x)\pi(x)}{g(x)}]$$

and the new Monte Carlo estimate is

$$\tilde{\mu} = \frac{1}{m}\sum_{i=1}^{m} h(x^{(i)})\frac{\pi(x^{(i)})}{g(x^{(i)})} = \frac{1}{m}\sum_{i=1}^{m} w^{(i)} h(x^{(i)})$$

where $w^{(i)} = \pi(x^{(i)})/g(x^{(i)})$ are the **importance weights**.

5 Importance Sampling: properties of the estimator

- **Rate of convergence:**

$$\tilde{\mu} \longrightarrow \mu \text{ in } O(m^{-1/2})$$

- **Unbiasedness:**

$$E[\tilde{\mu}] = E[\frac{1}{m}\sum_{i=1}^{m}\frac{\pi(X^{(i)})h(X^{(i)})}{g(X^{(i)})}] = E_g[\frac{\pi(X)h(X)}{g(X)}] = E_\pi[h(X)]$$

- **Variance:**

$$\text{var}(\tilde{\mu}) = \frac{1}{m}\int_\chi (\frac{h(x)\pi(x)}{g(x)} - \mu)^2 g(x)dx$$

Note: **minimized** if $g(x) \propto |h(x)\pi(x)|$. In particular, if $\alpha g(x) = h(x)\pi(x)$, then $\tilde{\mu} = \alpha$, and $\text{var}(\tilde{\mu}) = 0$!

6 Importance Sampling: biased estimator

- Note that:

$$\bar{w} = \frac{1}{m}\sum_{i=1}^{m} w^{(i)} \qquad E[\bar{w}] = E[\frac{1}{m}\sum_{i=1}^{m}\frac{\pi(X^{(i)})}{g(X^{(i)})}] = 1$$

  and one can use also the following estimator:

$$\hat{\mu} = \frac{w^{(1)}h(x^{(1)}) + \cdots + w^{(m)}h(x^{(m)})}{w^{(1)} + \cdots + w^{(m)}}$$

- **Biased:**

$$\begin{aligned} E[\hat{\mu}] &= E[\frac{\sum_{i=1}^{m} w^{(i)}h(X^{(i)})}{\sum_{i=1}^{m} w^{(i)}}] = \sum_{i=1}^{m} E[\frac{\pi(X^{(i)})h(X^{(i)})}{g(X^{(i)})\sum_{i=1}^{m} w^{(i)}}] \\ &\neq E_{\pi}[h(X)] \end{aligned}$$

7 Advantages of the biased estimator

- $\hat{\mu}$ may have **smaller mean square error** than $\tilde{\mu}$
- Need to know π only **up to a constant factor** c. Therefore we can use the weights

$$w^{(i)} = \frac{c\pi(x^{(i)})}{g(x^{(i)})}$$

8 Importance Sampling: an estimator independent of h

- **Goal:** computing $E_\pi[h(X)]$ for some arbitrary h, when sampling from π is difficult
- **Solution:** design g and **use importance sampling**
- **Recap:**
  - Ideal case:

$$(x^{(1)}, ..., x^{(m)}) \sim \pi \ \rightarrow \ \hat{\hat{\mu}} = \frac{1}{m} \sum_{i=1}^{m} h(x^{(i)})$$

  - Importance sampling solution:

$$(x^{(1)}, ..., x^{(m)}) \sim g \rightarrow \ \hat{\mu} = \frac{\sum_{i=1}^{m} w^{(i)} h(x^{(i)})}{\sum_{i=1}^{m} w^{(i)}}$$

9 Efficiency of Importance Sampling

- Define the **coefficient of variation** as

$$\mathrm{cv}^2(w) = \frac{\sum_{j=1}^{m}(w^{(j)} - \bar{w})^2}{(m-1)\bar{w}^2}$$

- Define the **effective sample size** as

$$\mathrm{ESS}(m) = \frac{m}{1 + \mathrm{cv}^2(w)}$$

- As a first order approximation

$$\frac{\mathrm{var}_\pi(\hat{\hat{\mu}})}{\mathrm{var}_g\,(\hat{\mu})} \approx \frac{1}{1 + \mathrm{cv}^2(w)} = \frac{\mathrm{ESS}(m)}{m},$$

10 Interpretation of the efficiency

- Since

$$\mathrm{var}_\pi(\hat{\hat{\mu}}) \propto \frac{1}{m}$$

  then

$$\mathrm{var}_g(\hat{\mu}) \propto \frac{1}{\mathrm{ESS}(m)}$$

- **m samples drawn from g are worth** ESS(m) **samples drawn from π**
- **If**, $g(x) \approx \pi(x) \Rightarrow$ weights are **similar** $\Rightarrow \mathrm{cv}^2(w)$ is **small** $\Rightarrow$ ESS(m) $\approx m$
- **Rule of thumb: keep** $\mathrm{cv}^2(w)$ **small**

11 Sequential Importance Sampling: dealing with high dimensional spaces

- **Goal**: computing $E_\pi[h(\mathbf{X})]$, with $\mathbf{X} = (X_1, ..., X_n) \sim \pi$ for $n \gg 1$.
- **Basic idea**: decompose $\pi(\mathbf{x})$

$$\pi(\mathbf{x}) = \pi(x_1)\pi(x_2|x_1)\cdots\pi(x_n|x_1, ..., x_{n-1})$$

and sample from the **conditionals**:

$$\left.\begin{array}{l} x_1^{(j)} \sim \pi(x_1), \\ x_2^{(j)} \sim \pi(x_2|x_1^{(j)}), \\ \quad\vdots \\ x_n^{(j)} \sim \pi(x_n|x_1^{(j)}, ..., x_{n-1}^{(j)}) \end{array}\right\} \longrightarrow (x_1^{(j)}, x_2^{(j)}, ..., x_n^{(j)}).$$

12 Sequential Importance Sampling: dealing with high dimensional spaces

- **Problem**: cannot sample from $\pi(x_t|x_1, ..., x_{t-1})$.
- **Idea**: generalize importance sampling: sample from a **sequence of trial distributions** $g_1(x_1), g_2(x_2|x_1), \ldots,$ $g_n(x_n|x_1, ..., x_{n-1})$, such that:

$$g(\mathbf{x}) = g_1(x_1)g_2(x_2|x_1)\cdots g_n(x_n|x_1, ..., x_{n-1})$$

and

$$w(\mathbf{x}) = \frac{\pi(x_1)\pi(x_2|x_1)\cdots\pi(x_n|x_1, ..., x_{n-1})}{g_1(x_1)g_2(x_2|x_1)\cdots g_n(x_n|x_1, ..., x_{n-1})}$$

If $\mathbf{x}_t = (x_1, \ldots, x_t)$ is the **partial sample**), then the **partial weight** is

$$w_t(\mathbf{x}_t) = w_{t-1}(\mathbf{x}_{t-1})\frac{\pi(x_t|\mathbf{x}_{t-1})}{g_t(x_t|\mathbf{x}_{t-1})} = w_{t-1}(\mathbf{x}_{t-1})\frac{\pi(\mathbf{x}_t)}{\pi(\mathbf{x}_{t-1})g_t(x_t|\mathbf{x}_{t-1})}$$

13 Sequential Importance Sampling

- **Problem**: cannot even **compute the marginal** $\pi(\mathbf{x}_t)$.
- **Idea**: introduce another layer of **auxiliary distributions** $\pi_1(\mathbf{x}_1), \pi_2(\mathbf{x}_2), \ldots, \pi_n(\mathbf{x}_n)$ such that

$$\pi_t(\mathbf{x}_t) \approx \pi(\mathbf{x}_t), \quad t = 1, \ldots, n-1 \quad \text{and} \quad \pi_n(\mathbf{x}_n) = \pi(\mathbf{x}_n).$$

14 SIS step

1: **for** $t = 2, ..., n$ **do**

2: draw $x_t \sim g_t(x_t|\mathbf{x}_{t-1})$

3: let $\mathbf{x}_t \leftarrow (x_t, \mathbf{x}_{t-1})$

4: compute the **incremental weight**

$$u_t \leftarrow \frac{\pi_t(\mathbf{x}_t)}{\pi_{t-1}(\mathbf{x}_{t-1})g_t(x_t|\mathbf{x}_{t-1})} = \frac{\pi_t(\mathbf{x}_{t-1})}{\pi_{t-1}(\mathbf{x}_{t-1})} \frac{\pi_t(\mathbf{x}_{t-1})}{g_t(x_t|\mathbf{x}_{t-1})}$$

5: compute the **partial weight** $w_t \leftarrow w_{t-1}u_t$.

6: **end for**

where

distribution	notation	approximates
trial	$g_t(x_t\|x_1, ..., x_{t-1})$	$\pi(x_t\|x_1, ..., x_{t-1})$
auxiliary	$\pi_t(x_1, ..., x_t)$	$\pi(x_1, ..., x_t)$

15 The choice of the trial distribution

- **1-step-look-ahead:**

$$g_t(x_t|\mathbf{x}_{t-1}) \propto \pi_t(x_t|\mathbf{x}_{t-1})$$

simpler incremental weight $u_t = \pi_t(\mathbf{x}_{t-1})/\pi_{t-1}(\mathbf{x}_{t-1})$, with no dependence on x_t, and no correction ratio

- $(s+1)$**-step-look-ahead:**

$$g_t(x_t|\mathbf{x}_{t-1}) \propto \int \pi_{t+s}(x_{t+s}, \ldots, x_t|\mathbf{x}_{t-1}) dx_{t+1} \cdots dx_{t+s}$$

tries to make use of as much future information as possible. It is computationally impractical for high s.

16 The normalizing constant

- If we know $\pi(\mathbf{x}_t)$ only up to a normalizing constant, which means that we know $q_t(\mathbf{x}_t) = Z_t\pi(\mathbf{x}_t)$. Then, the **incremental weight** is

$$u_t = \frac{q_t(\mathbf{x}_t)}{q_{t-1}(\mathbf{x}_{t-1})g_t(x_t|\mathbf{x}_{t-1})} = \frac{Z_t\pi_t(\mathbf{x}_t)}{Z_{t-1}\pi_{t-1}(\mathbf{x}_{t-1})g_t(x_t|\mathbf{x}_{t-1})}$$

- The final weight become

$$w_n = \Pi_{t=1}^n u_t = \frac{Z_n}{Z_1}\frac{\pi_n(\mathbf{x}_n)}{g(x_1)\cdots g_n(x_n|\mathbf{x}_{n-1})}$$

- It is possible to **estimate the normalizing constant** by

$$E[w_n] = \frac{Z_n}{Z_1}$$

17 The parallel SIS framework

- **Goal:** draw m samples $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$, from π to estimate $E_\pi[h(\mathbf{X})]$.
- **Sequential approach:** repeat m sequential sampling processes to sample $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}$.
- **Parallel approach:** start m independent sequential sampling processes in parallel meaning that:

  for $t = 1, \ldots, n$ do: generate m i.i.d. samples $x_t^{(j)}$ from $g_t(\cdot|\mathbf{x}_{t-1}^{(j)})$, $j = 1, \ldots, m$ to produce the collection $\{(x_t^{(1)}, \mathbf{x}_{t-1}^{(1)}), \ldots, (x_t^{(m)}, \mathbf{x}_{t-1}^{(m)})\}$.

18 Speeding up the SIS: uses of the estimated weights

The estimates of the weights are used to "diagnose and repair" the collection of samples by one of the following techniques:

- **Rejection control**: handle a **specific sample** $\mathbf{x}_t^{(j)}$ by discarding it and starting from scratch if its weight is too small.
- **Resampling**: handle the **whole collection** $\{\mathbf{x}_t^{(1)}, \ldots, \mathbf{x}_t^{(m)}\}$ by replacing low weighted samples with copies of the high weighted ones.
- **Partial rejection control**: handle a **specific sample** $\mathbf{x}_t^{(j)}$ by backtracking and resampling it if its weight is too small.

19 Case study: self-avoiding random walk

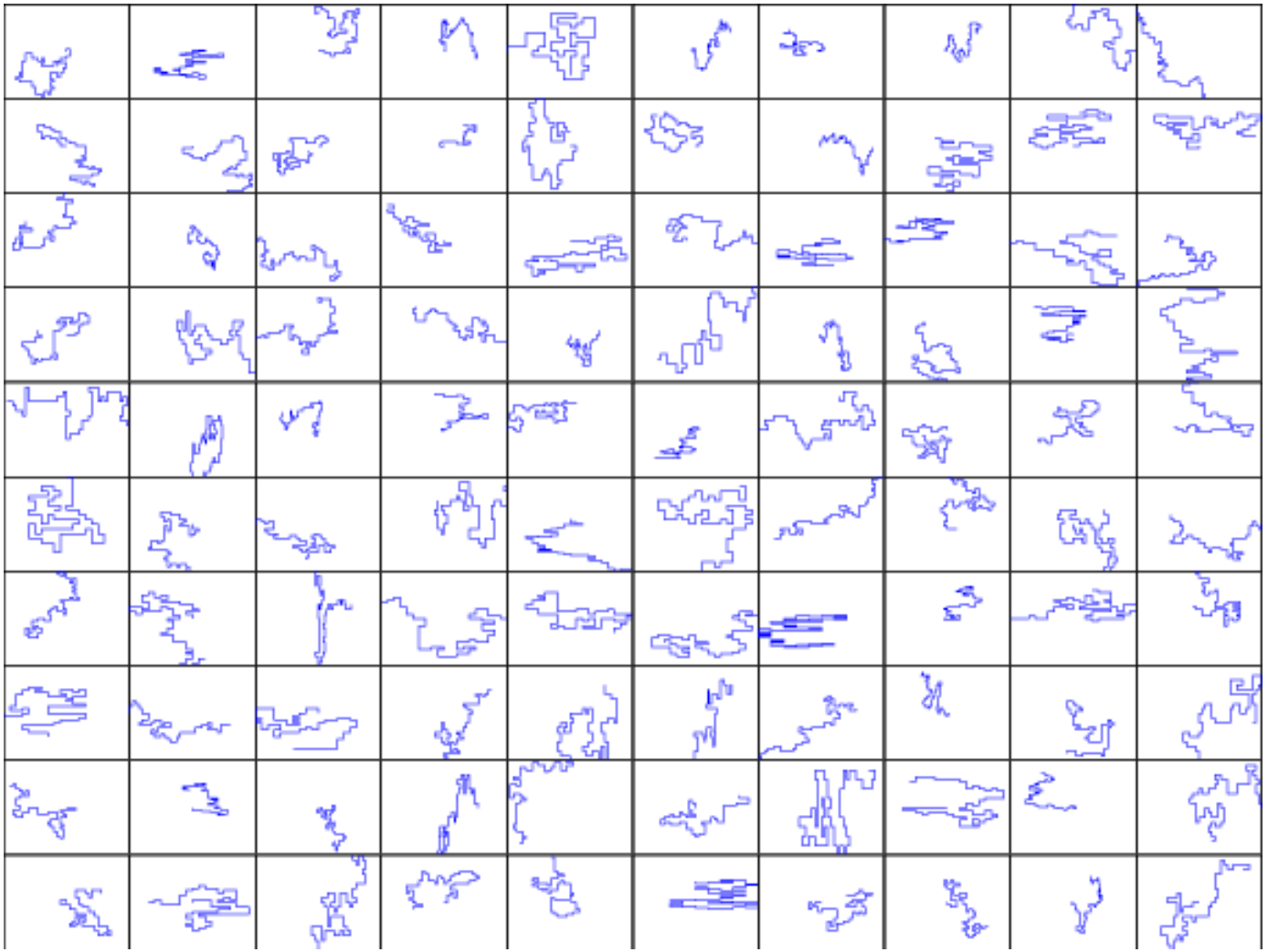
A **self-avoiding random walk (SAW)** of length n is fully characterized by the points $\mathbf{x} = (x_1, ..., x_n)$, such that $x_t \in \mathbb{Z}^2$, $\|x_{i+1} - x_i\| = 1$, and $x_i \neq x_k, \forall k < i$. Ω_n is the set of all SAWs of length n.

Problem: sample m SAWs from the following distribution

$$\pi(\mathbf{x}) = \frac{1}{Z_n}, \quad Z_n = |\Omega_n|,$$

and compute statistics such as $E[\|x_n - x_1\|^2]$, and Z_n.

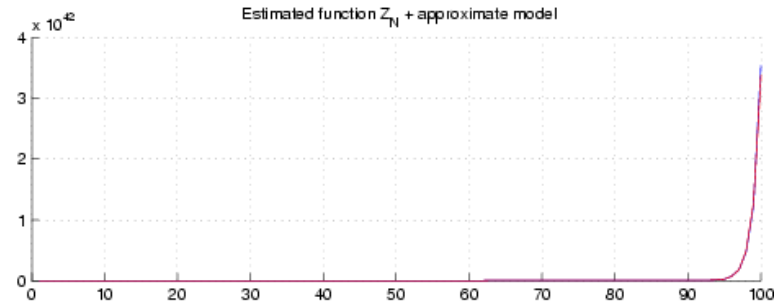
20 SAWs samples

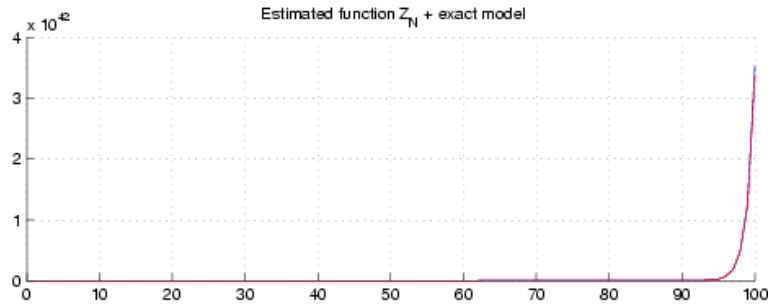


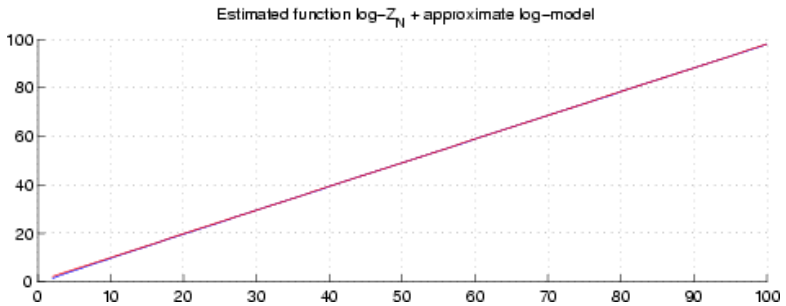
21 Estimating the partition function

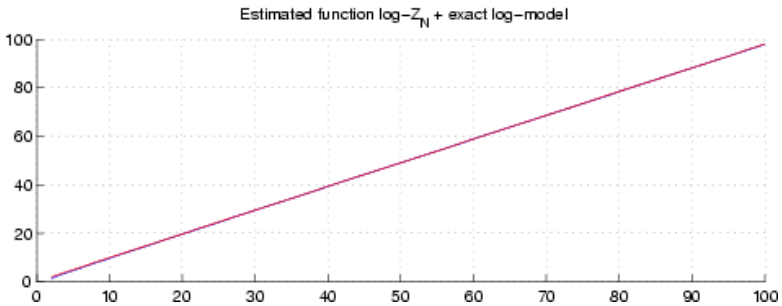
$$Z_t \approx cq^t, \quad c = 1.32\ (2.14), \quad q = 2.65\ (2.66),$$

$$Z_t = q_{\text{eff}}^t t^{\gamma-1}, \quad q_{\text{eff}} = 2.64\ (2.64), \quad \gamma = 1.3\ (1.38).$$





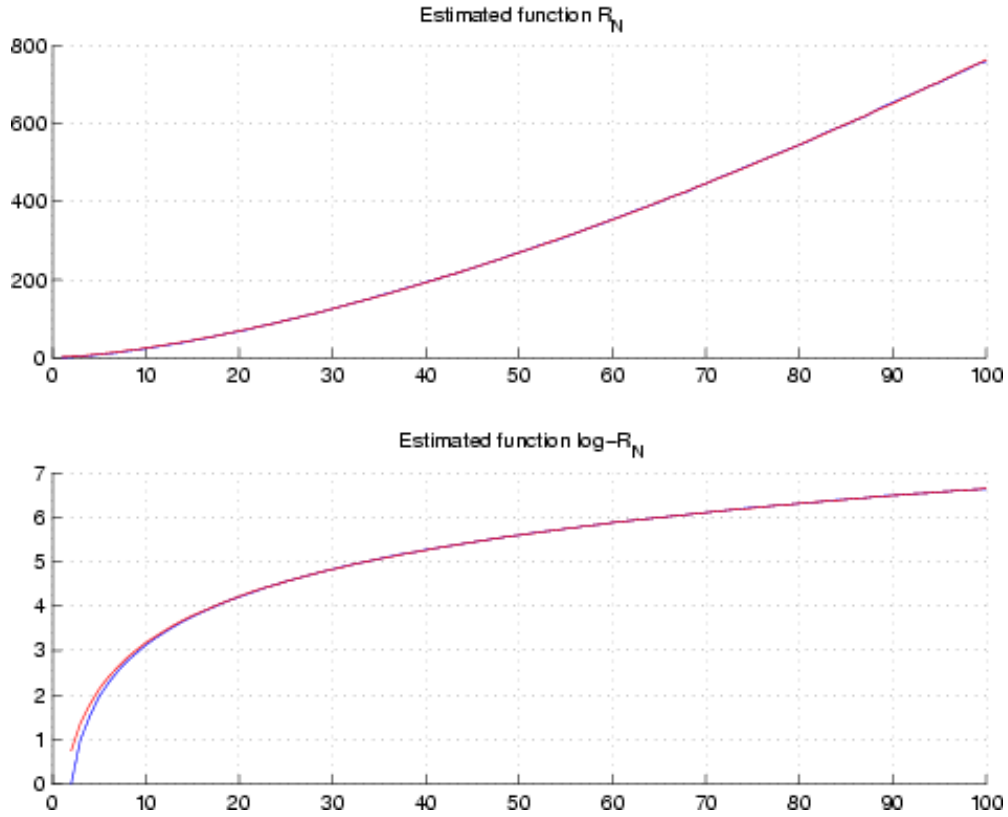




22 Estimating the squared extension

We used the samples to estimate $R_t = E_\pi[\|x_t - x_1\|^2]$.

$$R_t \approx at^b, \quad a = 1.0\ (0.917), \quad b = 1.44\ (1.45).$$

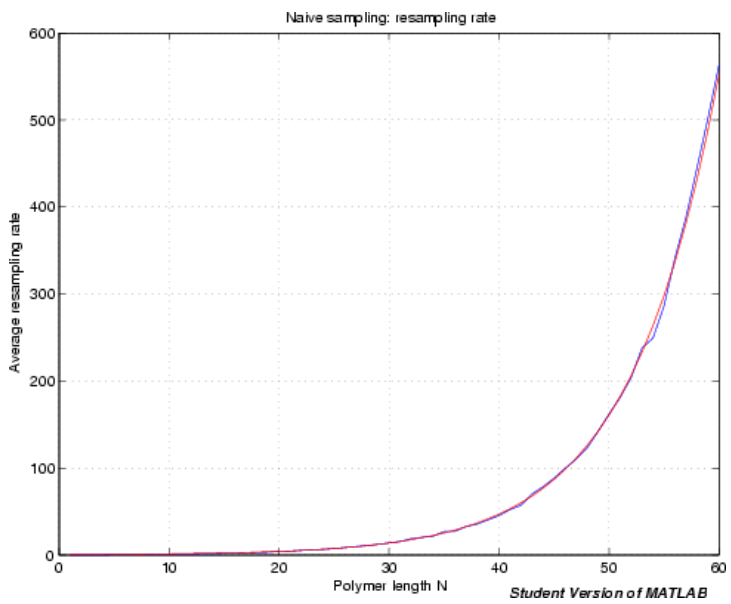


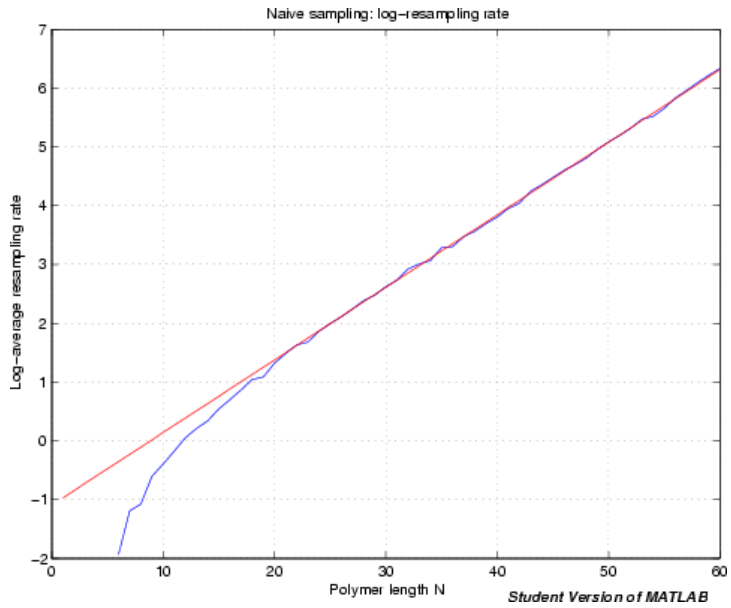
23 A naive sampler

1. Start the SAW at $(0, 0)$.
2. For $t = 2, \ldots, n$ do
   - the walker cannot go back to x_{t-1};
   - sample, with equal probability, one of the three allowed neighboring positions;
   - if the positions has already been visited go to Step 1.

24 A naive sampler

The graphs show how many times one has to restart the sampler before a valid SAW can be drawn, as function of n. Asymptotically, the average number of resampling is $O(1.13^n)$.





Need a more efficient sampler!

25 Setting up the SIS framework

To setup a SIS sampler, one has to

1. choose the **trial distributions** $g_t(\mathbf{x}_t),\ t = 1, \ldots, n$ and
2. choose the **auxiliary distributions** $\pi_t(\mathbf{x}_t),\ t = 1, \ldots, n$.

26 Auxiliary distribution

Different choices for $\pi_t(\mathbf{x}_t)$ are possible:

type	auxiliary distribution
1-look-ahead	$\pi_t(\mathbf{x}_t) = \pi^t(\mathbf{x}_t)$
2-look-ahead	$\pi_t(\mathbf{x}_t) = \sum_{x_{t+1}} \pi^{t+1}(x_{t+1}, \mathbf{x}_t)$
$\vdots$	$\vdots$
q-look-ahead	$\pi_t(\mathbf{x}_t) = \sum_{x_{t+1},\ldots,x_{t+q-1}} \pi^{t+q-1}(x_{t+q-1}, \ldots, x_{t+1}, \mathbf{x}_t)$

In general:

$$\pi_t(\mathbf{x}_t) = \pi^{t+q-1}(\mathbf{x}_t) = \frac{n^{t+q-1}(\mathbf{x}_t)}{Z_{t+q-1}}, \quad q \geq 1.$$

where $n^{t+l}(\mathbf{x}_t) =$ is the number of SAWs of length $t+l$ that start with $\mathbf{x}_t = (x_1, ..., x_t)$.

27 Trial distribution

- Since the auxiliary distributions are very simple, the trial distributions can be obtained directly from the these:

$$g_t(x_t|\mathbf{x}_{t-1}) = \pi_t(x_t|\mathbf{x}_{t-1}) = \frac{n^{t+q-1}(x_t, \mathbf{x}_{t-1})}{n^{t+q-1}(\mathbf{x}_{t-1})}.$$

- The incremental weights are (up to a unknown constant factor)

$$u_t^{(j)} \propto \frac{1}{g_t(x_t^{(j)}|x_1^{(j)}, ..., x_{t-1}^{j)})}.$$

28 Our criterion for efficiency

- **Efficiency**. Let

	SIS sampler	ideal sampler
# sampling op.	T	T^*
# equivalent samples obtained	m	m^*

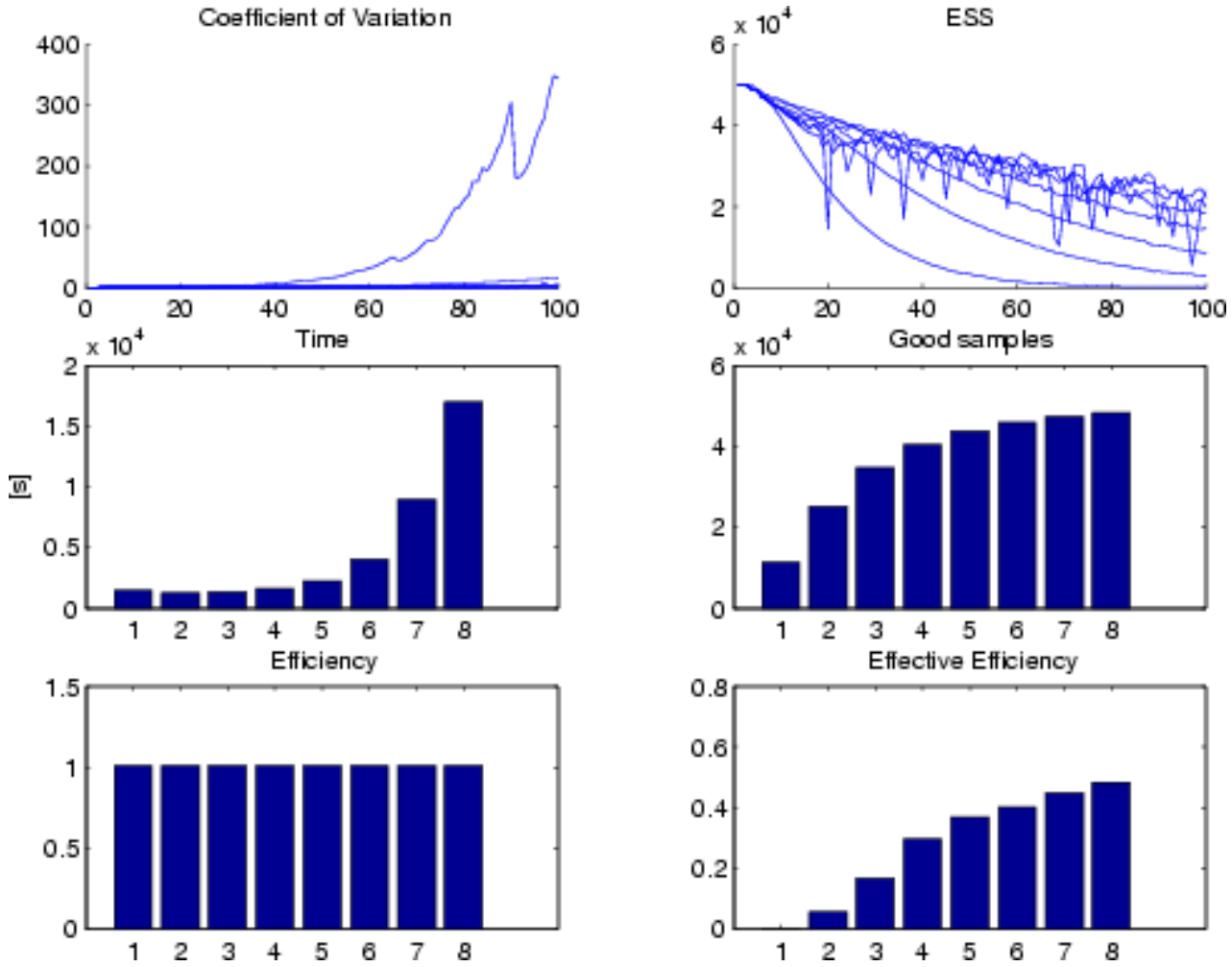
The efficiency of the SIS sampler is

$$E = \frac{m/T}{m^*/T^*}$$

- **Effective efficiency**. If we set $m = \mathrm{ESS}(m^*)$, then

$$E_{\mathrm{eff}} = \frac{T^*}{T}\frac{\mathrm{ESS}(m^*)}{m^*}.$$

29 Comparing the trial distributions



30 Speeding up the sampler

We can speed up the sampler by using the following techniques:

- rejection control,
- resampling,
- partial rejection control.

31 Rejection Control (RC)

- **Goal**: avoid to carry on **low-weighted** (therefore useless) samples.
- **How**: use the partial weights to detect and kill as soon as possible low-weighted samples.

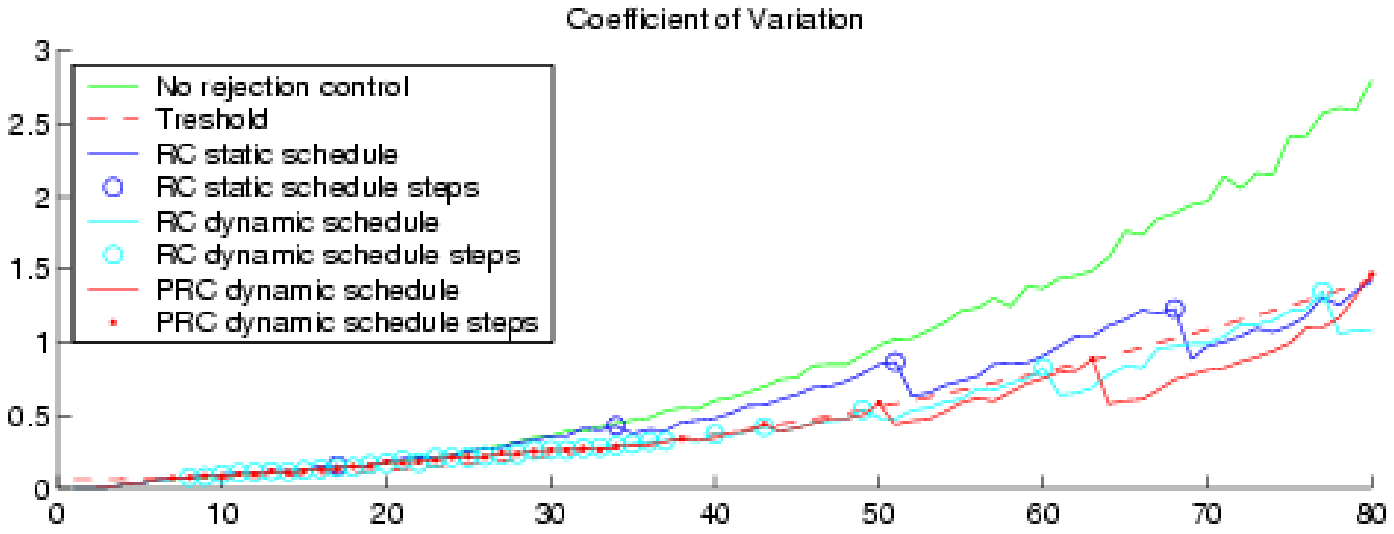
We perform a RC step with **time schedule** $\{t_1, t_2, ..., t_k, ..., t_l\}$ and **thresholds** $\{c_1, c_2, ..., c_k, ..., c_l\}$. At time t_k, each sample is accepted with probability

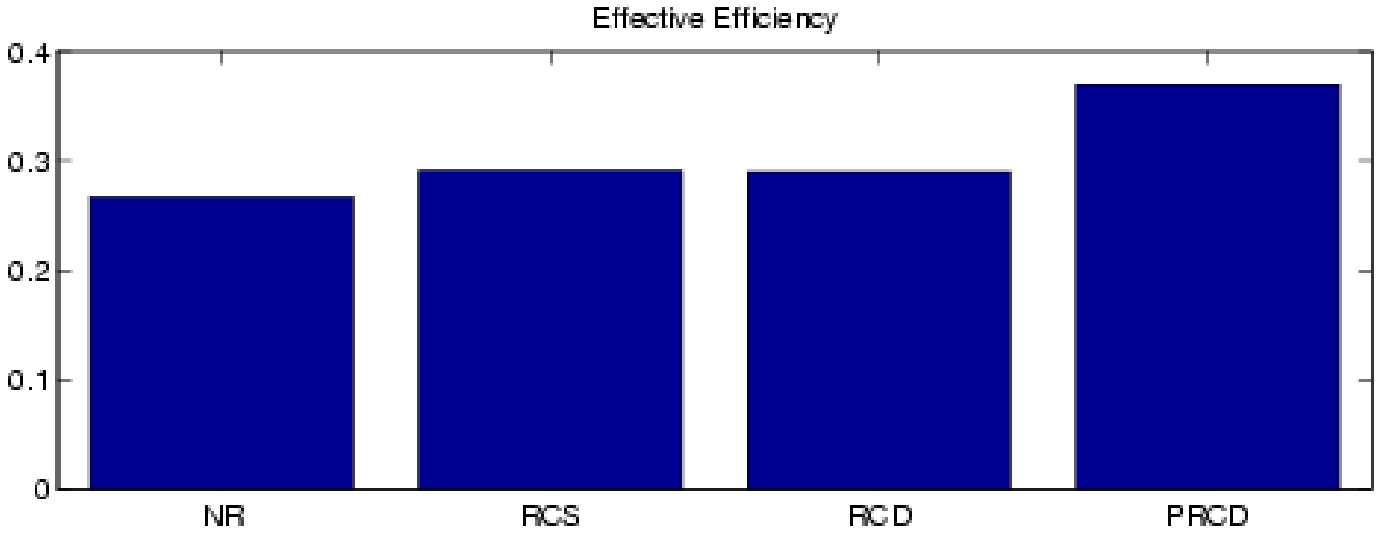
$$\min\left\{1, \frac{w^{(j)}}{c_k}\right\}$$

and the weights are updated so that

$$w^{(*j)} = p_c \max\left\{c_t, w^{(*j)}\right\}.$$

- The schedule can be either
  - **static**: $t_k = kT$,
  - **dynamic**: we do a RC step if $\mathrm{cv}^2 > c_{\mathrm{sched},t}$; typically, $c_{\mathrm{sched},t} = a_{\mathrm{sched}} + b_{\mathrm{sched}} t^{\alpha_{\mathrm{sched}}}$.
- The rejection threshold can be set in a variety of ways:
  - **epsilon**: $c_t = \epsilon$ very small: discards only zero weighted samples.
  - **average**: $c_t = \alpha \min w_t^{(j)} + \beta \overline{w}_t + \gamma \max w_t^{(j)}$.
  - **percentile** $c_t = \mathrm{percentile}(\{w_t^{(1)}, ..., w_t^{(m)}\}, p)$.
  - **polynomial** $c_t = a + bt^{\alpha}$.


Coefficient of Variation
No rejection control
Treshold
RC static schedule
RC static schedule steps
RC dynamic schedule
RC dynamic schedule steps
PRC dynamic schedule
PRC dynamic schedule steps
0
0.5
1
1.5
2
2.5
3
0
10
20
30
40
50
60
70
80


Effective Efficiency
0
0.1
0.2
0.3
0.4
NR
RCS
RCD
PRCD

32 Resampling (R)

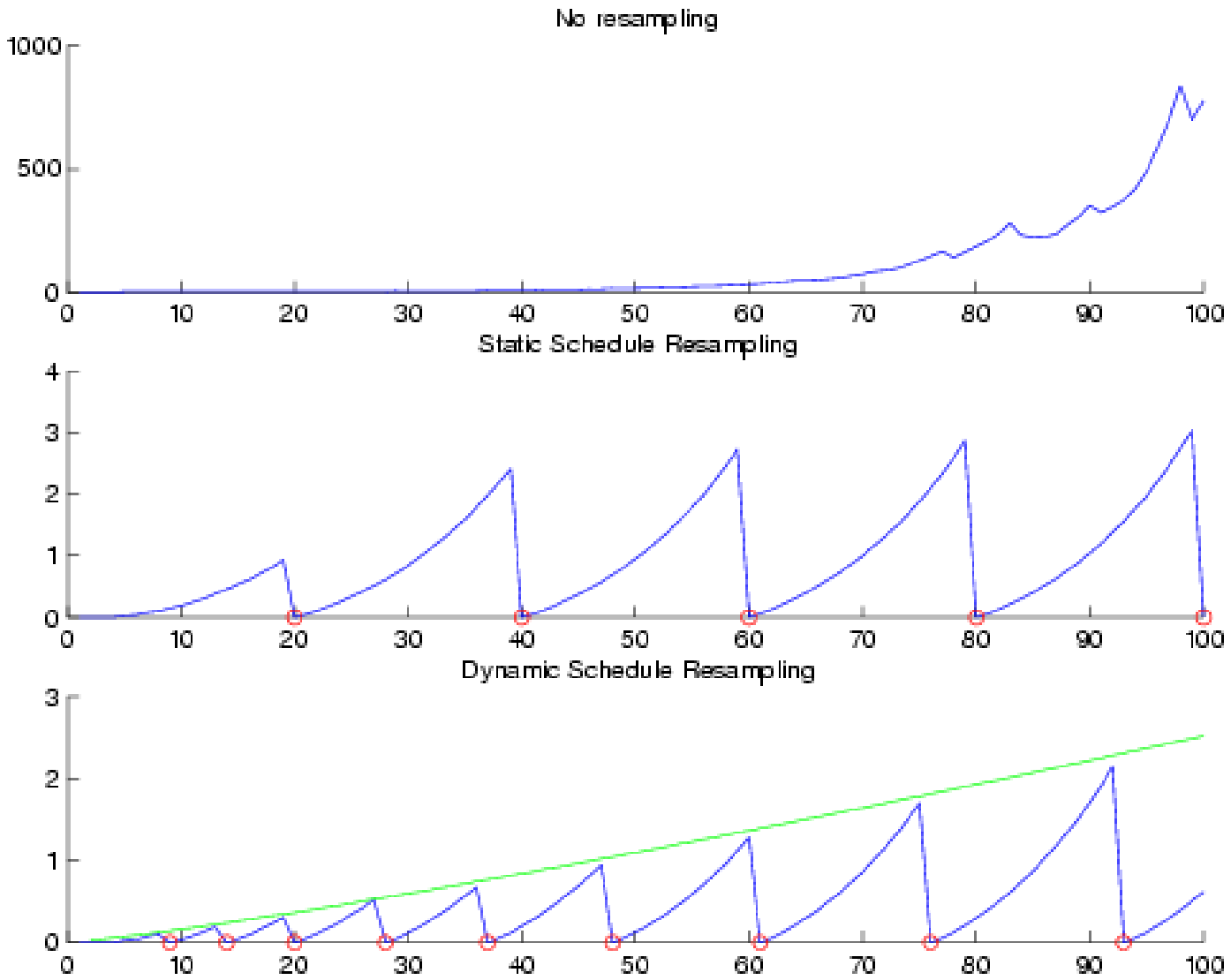
- **Goal**: keep a set of **high-weighted** (aka useful) samples.
- **How**: substitute low weighted samples with **copies** of the high weighted ones.

We perform a R step with **time schedule** $\{t_1, t_2, ..., t_k, ..., t_l\}$. Given the samples $\{x_{t_k}^{(1)}, ..., x_{t_k}^{(m)}\}$, we create a new set of samples $\{x_{t_k}^{\prime(1)}, ..., x_{t_k}^{\prime(m)}\}$ by taking copies of the high-weighted ones:

- **Simple scheme**: $x_{t_k}^{\prime(j')} = x_{t_k}^{(j)}, \quad j \sim w_{t_k}^{(j)} / \overline{w}_{t_k}$.
- **Residual scheme**: take $\lfloor w_{t_k}^{(j)} / \overline{w}_{t_k} \rfloor$ copies of the sample $x_{t_k}^{(j)}$ and fill the gaps using the simple scheme.

Then, we update the weights:

$$w_{t_k}^{(j)} \leftarrow \overline{w}_{t_k}, \quad j = 1, ..., m.$$


No resampling
1000
500
0
0
10
20
30
40
50
60
70
80
90
100
Static Schedule Resampling
4
3
2
1
0
0
10
20
30
40
50
60
70
80
90
100
Dynamic Schedule Resampling
3
2
1
0
0
10
20
30
40
50
60
70
80
90
100

33 Partial Rejection Control (PRC)

- **Goal**: keep a set of **high-weighted** samples.
- **How**: substituted low weighted samples with **computationally cheap** high-weighted ones.

We perform a R step with **time schedule** $\{t_1, t_2, ..., t_k, ..., t_l\}$ and **thresholds** $\{c_1, c_2, ..., c_k, ..., c_l\}$. At time t_k

- do a RC step, rejecting L samples;
- we go back at time t_{k-1} and we resample L new samples;
- we grow the new samples to time t_k.

A Appendix: some mathematical details

A.1 Theoretical justification of Rejection Control

Given that a sample is accepted, its distribution is

$$x_t^{(j)} \sim g_t^*(x_t|x_1^{(j)},...,x_{t-1}^{(j)}) = \frac{\min\left\{1, \frac{w_t^{(j)}(x_t)}{c_t}\right\}}{p_c} g_t(x_t|x_1^{(j)},...,x_{t-1}^{(j)}).$$

The distribution g^* is closer than g (in χ^2 norm) to the true marginal.

This is also the reason why the weights need to be updated:

$$w^{(*j)} \propto \frac{\pi_t}{g_t^*} = \frac{p_c}{\min\left\{1, \frac{w_t^{(j)}(x_t)}{c_t}\right\}} \frac{\pi_t}{g_t} = p_c \max\left\{c_t, w_t^{(j)}\right\}$$

A.2 Justification of Resampling

Consider $\{x^{(1)}, ..., x^{(m)} : x^{(j)} \sim g(x)\}$, m big. Then,

$$\#\{x^{(j)} : x^{(j)} = z\} \approx mg(z).$$

Any sample $x^{(j)} = z$ has probability $\approx w(z)/m$ to be resampled. Therefore

$$P(x^{*(j)} = z) \approx mg(z)\frac{w(z)}{m} = \pi(z)$$

which explains why samples weights are constant after resampling.

A.3 Evaluating the real efficiency of resampling is difficult

Resampling introduce **correlation** among samples: the efficiency formulas are not valid anymore. In particular, just after the **first** resampling step one has

$$E\left[\left(\sum_{i=1}^{n}\frac{x_i}{n}\right)^2\right] \approx \sigma_x^2\left(\frac{1}{\mathrm{ESS}(n)}+\frac{1}{n}\right).$$

so the variance of the estimator get **worst**!