Optimal Importance Sampling Density Approximation Methods

Josef Leydold

joint work with

Kemal Dinçer Dingeç and Wolfgang Hörmann

Institute for Statistics and Mathematics, WU Wien, Austria
Altınbaş University, İstanbul, Turkey
Boğaziçi University, Istanbul, Turkey

13th MCQMC 2018

Rennes, July 5, 2018

MC Estimate for Expectation

Given:

- absolutely continuous univariate random variate X with density f and CDF F.
- function q.

Compute:

$$I = \mathbb{E}_f[q(X)] = \int_{\mathbb{R}} q(x) f(x) dx = \int_{\mathbb{R}} q(x) dF(x)$$

Naive **Monte Carlo** (MC) estimator:

$$I_n = \frac{1}{n} \sum_{i=1}^{n} q(X_i) f(X_i) \,, \quad X_i \sim f$$

MC Integration Error

Advantage of MC integration:

We have a **probabilistic error estimate**

$$|I_n - I| = z_{1-\alpha/2} \frac{\sigma_f(q)}{\sqrt{n}}$$

which can be estimated from random sample X_i together with I_n.

Issues:

- Convergence $\mathcal{O}\left(1/\sqrt{n}\right)$ is slow.
- Need random variate generator for f.

Importance Sampling

Replace density f by a so called **importance density** g.

$$I = \mathbb{E}_f[q(X)] = \mathbb{E}_g\left[q(X)\frac{f(X)}{g(X)}\right] = \int_{\mathbb{R}} q(x)\frac{f(x)}{g(x)}g(x)dx$$

with MC estimator

$$I_n = \frac{1}{n}\sum_{i=1}^{n} q(X_i)\frac{f(X_i)}{g(X_i)}\,, \qquad X_i \sim g\,,$$

which is unbiased if

$$g(x) > 0 \quad \text{whenever} \quad q(x)f(x) \neq 0\,.$$

Optimal IS Density

Hope:

- $\sigma_g\big(q(x)f(x)/g(x)\big) < \sigma_f(q)$.
- Sampling from g is cheaper than from f.
- If possible: both.

Variance of MC estimator becomes **minimal** for

$$g(x) = \big|q(x)\big|f(x)\ .$$

For $q(x) \geq$ we even get a zero-variance IS density.

Optimal IS Density

However:
Sampling from optimal density $\left|q(x)\right|f(x)$ can be **challenging**.

Problem:
We have to normalize $\left|q(x)\right|f(x)$ which requires to compute

$$C = \int_{\mathbb{R}} |q(x)|f(x)dx \ .$$

However: If $q \geq 0$, then $C = I$!

That is, we need the integral to compute the integral with optimal IS density.

(See, e.g., C. Lemieux 2009)

Approximation Problems

(P1) Find an IS density $\tilde{g}$ that approximates the optimal density $g(x) \propto |q(x)|f(x)$ but is much cheaper to generate.

(P2) We assume that (for some reasons) g is a "good" IS density. Unfortunately, it is too expensive to sample from g directly. So we need a good approximation $\tilde{g}$ for g.

See next talk for an example.

Example: Sum of Lognormals

Density in CMC of (just) two log-normal random variates (with two parameters γ and s):

$$f(x)q(x) = \phi(x)\Phi\left(\sqrt{2/s}\left(\log\left(\frac{\gamma}{2}\right) - \log\left(\frac{1}{2}\left(\exp\left(\frac{s}{\sqrt{2}}x\right) + \exp\left(-\frac{s}{\sqrt{2}}x\right)\right)\right)\right)$$

where

- ϕ ... PDF of standard normal distribution
- Φ ... CDF of standard normal distribution

Automatic Sampling Method: PINV

PINV: Polynomial Interpolation of Inverse CDF
(Derflinger etal, 2010)

- Numerical inversion method when only PDF is known.
- Domain is partitioned into subintervals.
- Gauss-Lobato integration to estimate $F(x_{i+1}) - F(x_i)$ between some points x_i.
- Approximation of inverse CDF in each subinterval by means of Newton interpolation.
- Subintervals are adaptively splitted to ensure small u-error

$$\varepsilon_u = \max_u \left| u - F(\tilde{F}^{-1}(u)) \right| .$$

Automatic Sampling Method: PINV

- Why u-error?
  - It can be estimated for each subinterval.
  - It corresponds to the metric of weak convergence of random variables (convergence in distribution).

- Applying Gauss-Lobato integration and interpolation on the same subintervals makes setup fast and improves error estimate.
  Accuracy of $\leq 10^{-12}$ is possible in most applications.

Disadvantage:

- Many intervals required for high accuracy goal in case of rare events $\Rightarrow$ slow setup.
- Approximation errors may remain too large.

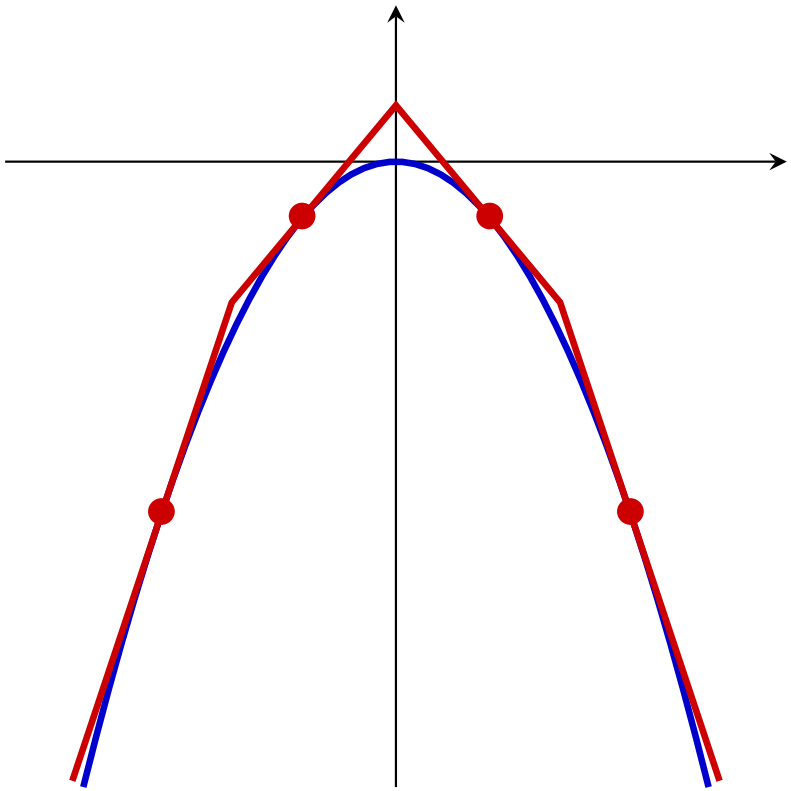
Automatic Sampling Method: TDR

TDR Transformed Density Rejection
(e.g., Hörmann etal, 2004)

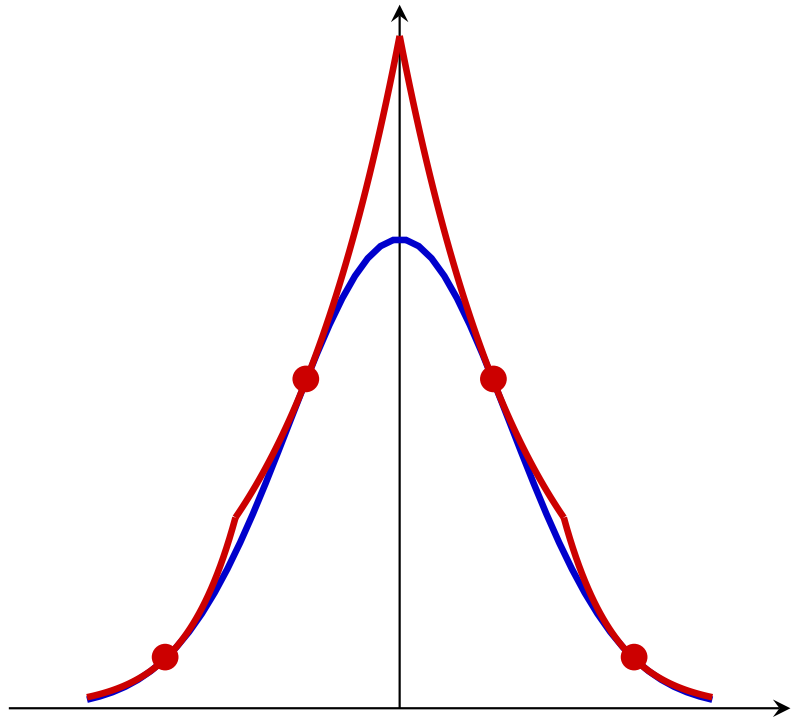
- Rejection method.
- Domain is partitioned into subintervals.
- Hat is constructed by means of tangents of the transformed density in subintervals.
- Subintervals are adaptively splitted to ensure close to 1 rejection constant.

Automatic Sampling Method: TDR

Example: PDF $\propto \exp(-x^2)$ and $T(x) = \log(x)$



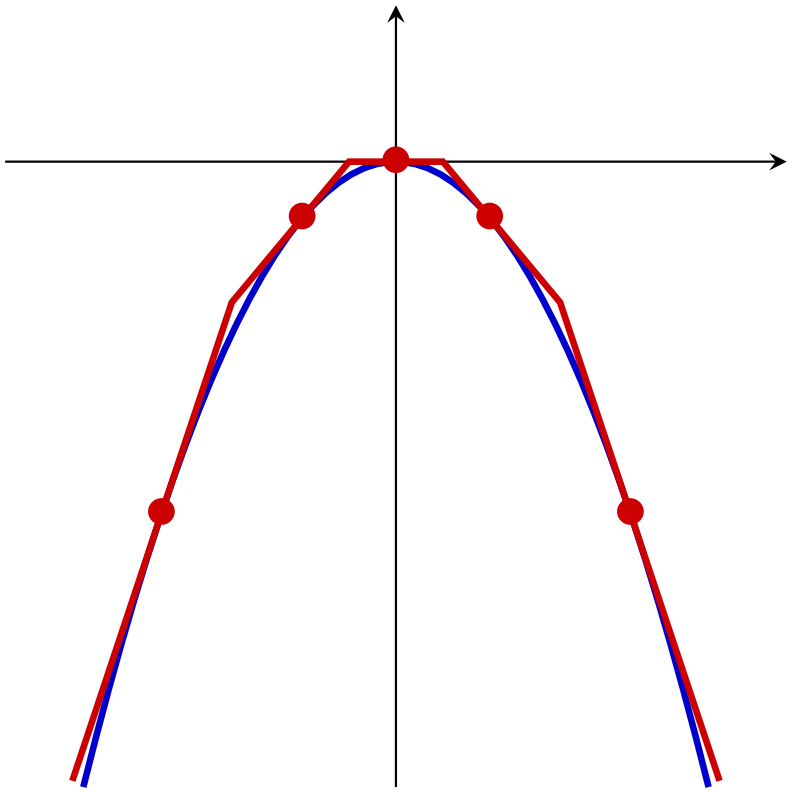
log-scale



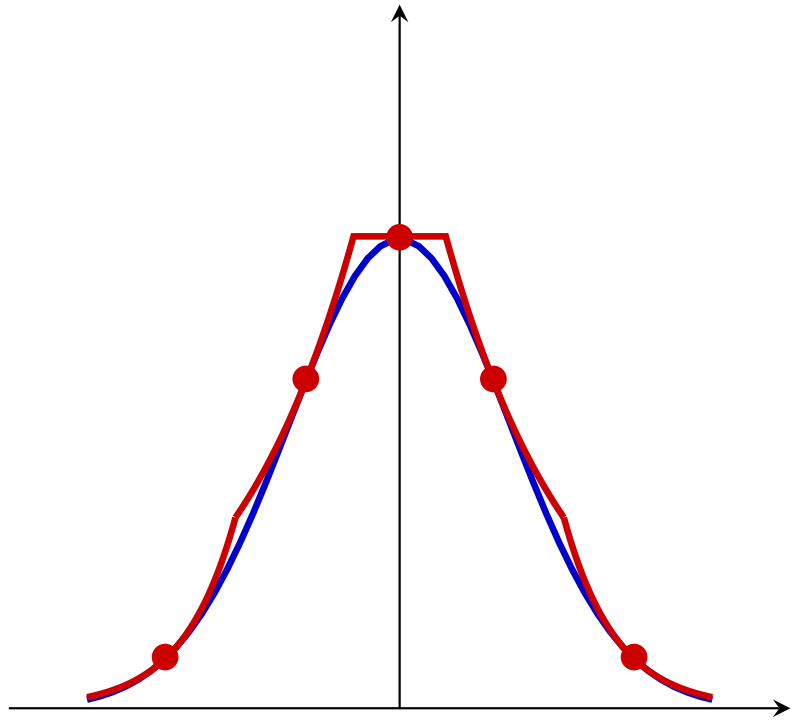
original scale

Automatic Sampling Method: TDR

Example: PDF $\propto \exp(-x^2)$ and $T(x) = \log(x)$



log-scale



original scale

Automatic Sampling Method: TDR

Advantage:

- Family of transformation T_c allows adaptation for target distribution.
- Sampling from and computing of "head PDF" is very fast.
- Normalization constant for "head PDF" for free.

Disadvantage:

- Only works for T-concave densities (i.e., $T \circ f$ is concave).
- Needs derivative of PDF.
- We still do not have the normalization constant for optimal IS density.
- Wastes rejected random variates.

R Package Runuran

Remark:

TDR and PINV (together with many other automatic methods) are implemented in our C library

UNU.RAN (Universal Non-Uniform RANdom variate generators)
`http://statmath.wu.ac.at/unuran`

R package **Runuran** provides a simple interface for **R**
`https://cran.r-project.org/package=Runuran`

MATLAB and octave interfaces are available on request.

Error-corrected Sampling

In our experience with QMC none of these “ready-to-use” automatic methods was suitable because

- the assumptions of TDR are not satisfied, and
- the setup of PINV was too slow for rare events and/or the approximation error was too large.

However, when using importance sampling we are **allowed to make** (small) **errors**.

From this point of view importance sampling can be seen as **error-corrected sampling**.

Error-corrected Sampling

- **Advantage:**
  A (rather rough) approximation of f is sufficient.

  (We do not need a hat function like in TDR,
  nor an "exact" approximation like in PINV.)

- **Disadvantage:**
  - Additional function evaluations for correction factor $\frac{f(x)}{g(x)}$.
  - This factor could introduce additional variance and/or numerical instabilities.

    How can we ensure that this cannot happen or is within some bounds?

Approximation Problem 1

(P1) Find a IS density $\tilde{g}$ that approximates the optimal density $g(x) \propto |q(x)|f(x)$ but is much cheaper to generate.

Motivated by our experiences with TDR we use **linear interpolation of the transformed density**.

Class T_c of Transformations

	$c > 0$	$c = 0$	$c < 0$	$c = -1/2$
$T(x)$	x^c	$\log(x)$	$-x^c$	$-x^{-1/2}$
$T^{-1}(x)$	$x^{1/c}$	e^x	$(-x)^{1/c}$	x^{-2}
$F_T(x)$	$\frac{x^{\frac{c+1}{c}}}{\frac{c+1}{c}}$	e^x	$\frac{-(-x)^{\frac{c+1}{c}}}{\frac{c+1}{c}}$	$-1/x$
$F_T^{-1}(x)$	$\left(x\frac{c+1}{c}\right)^{\frac{c}{c+1}}$	$\log(x)$	$-\left(-x\frac{c+1}{c}\right)^{\frac{c}{c+1}}$	$-1/x$

Beware:

Except for $T = \log$ the approximating function in the transformed scale must not vanish (i.e., must not intersect the x-axis).

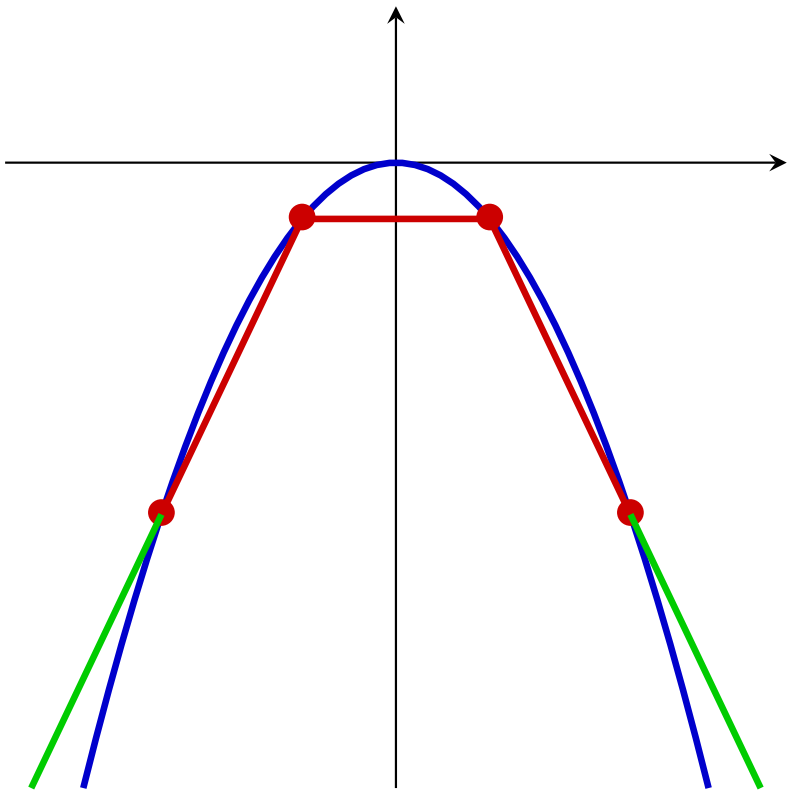
LINT

LINT: Linear Interplation of Transformed Density

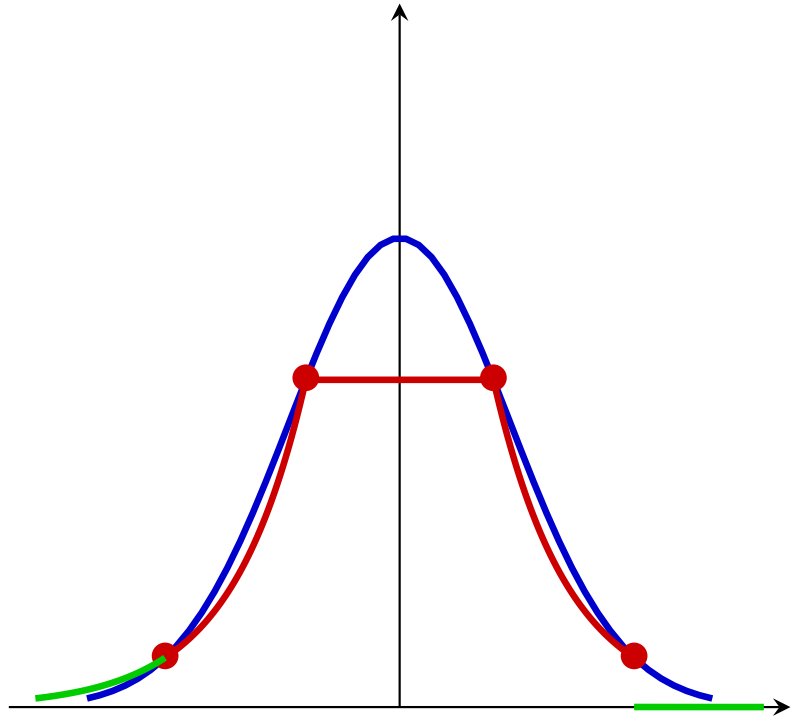
- Use linear interpolation (secants) between boundary points of subintervals.
- Use extrapolation of secants in unbounded subintervals.
- Using secants ensures that $T_c \circ \tilde{g}$ never vanishes.
- We can use different parameters c in the tails and the center to ensure that $\left| \frac{g(x)}{\tilde{g}(x)} \right|$ remains bounded from above.
- Generation from and computing of $\tilde{g}$ is very fast.
- Normalization constant for $\tilde{g}$ for free.

Automatic Sampling Method: LINT

Example: PDF $\propto \exp(-x^2)$ and $T(x) = \log(x)$



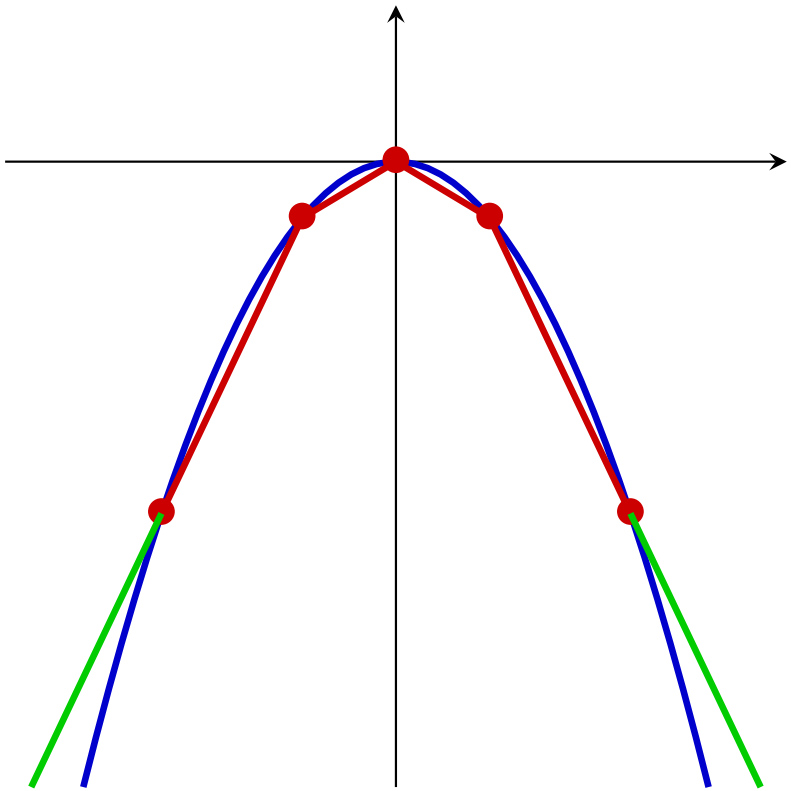
log-scale



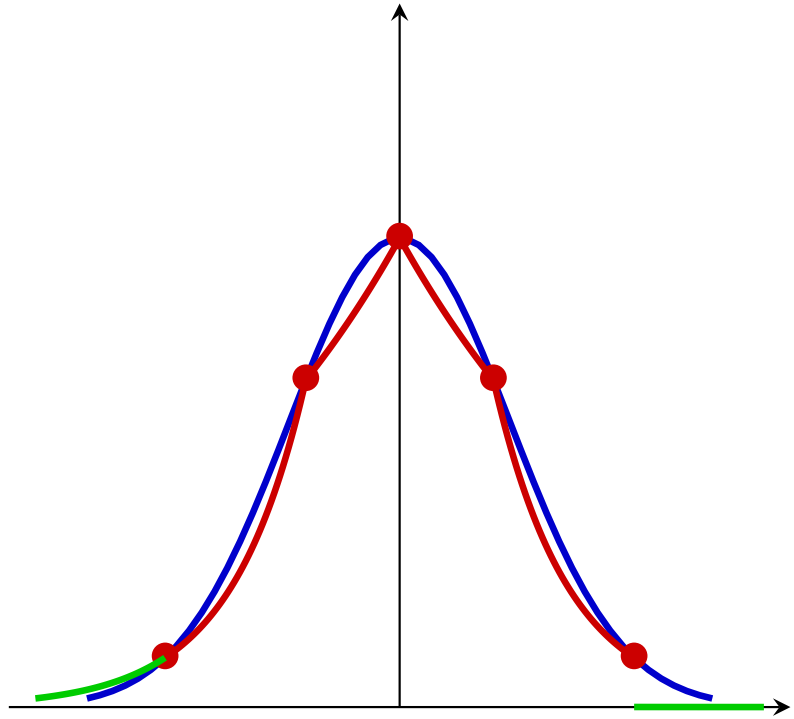
original scale

Automatic Sampling Method: LINT

Example: PDF $\propto \exp(-x^2)$ and $T(x) = \log(x)$



log-scale

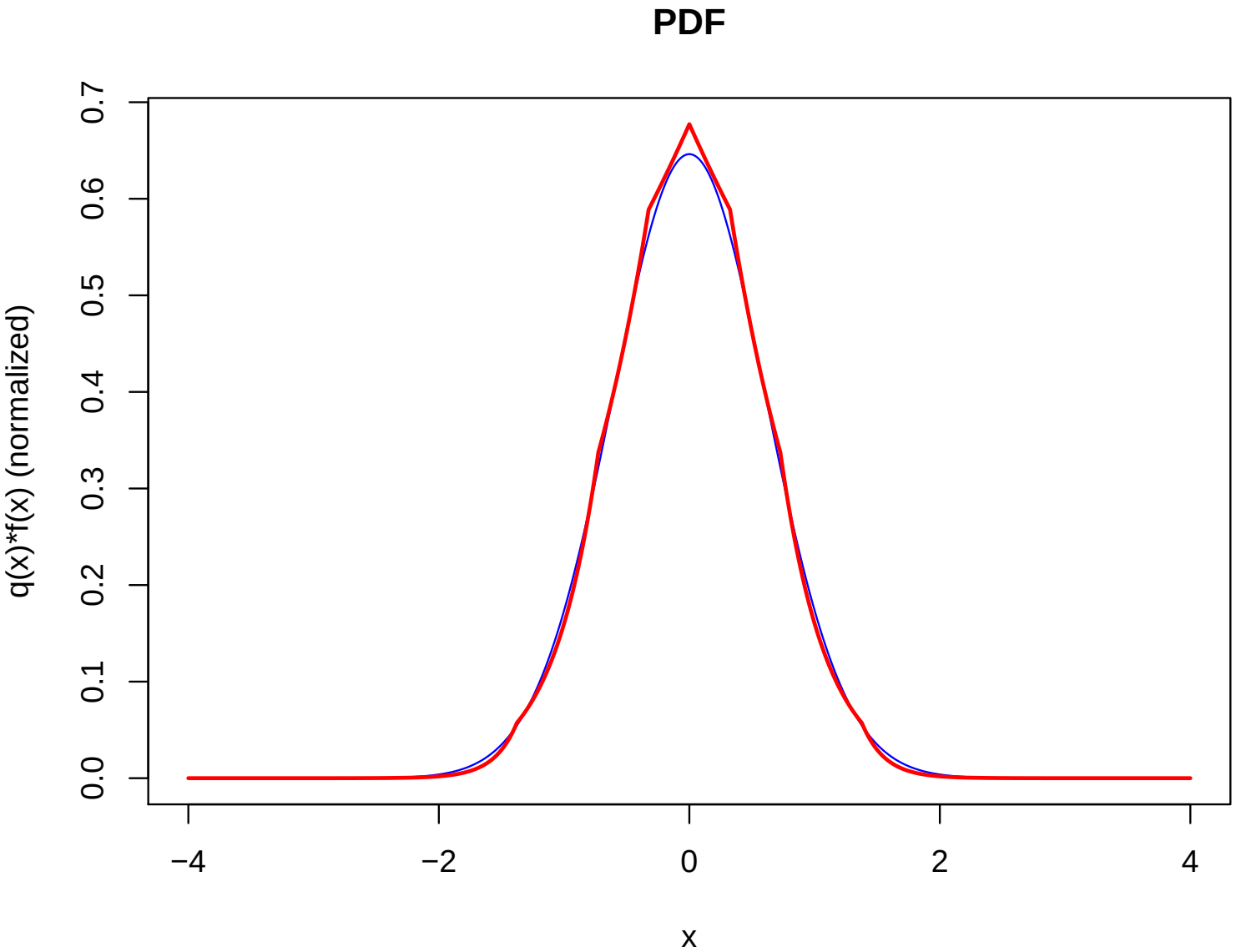


original scale

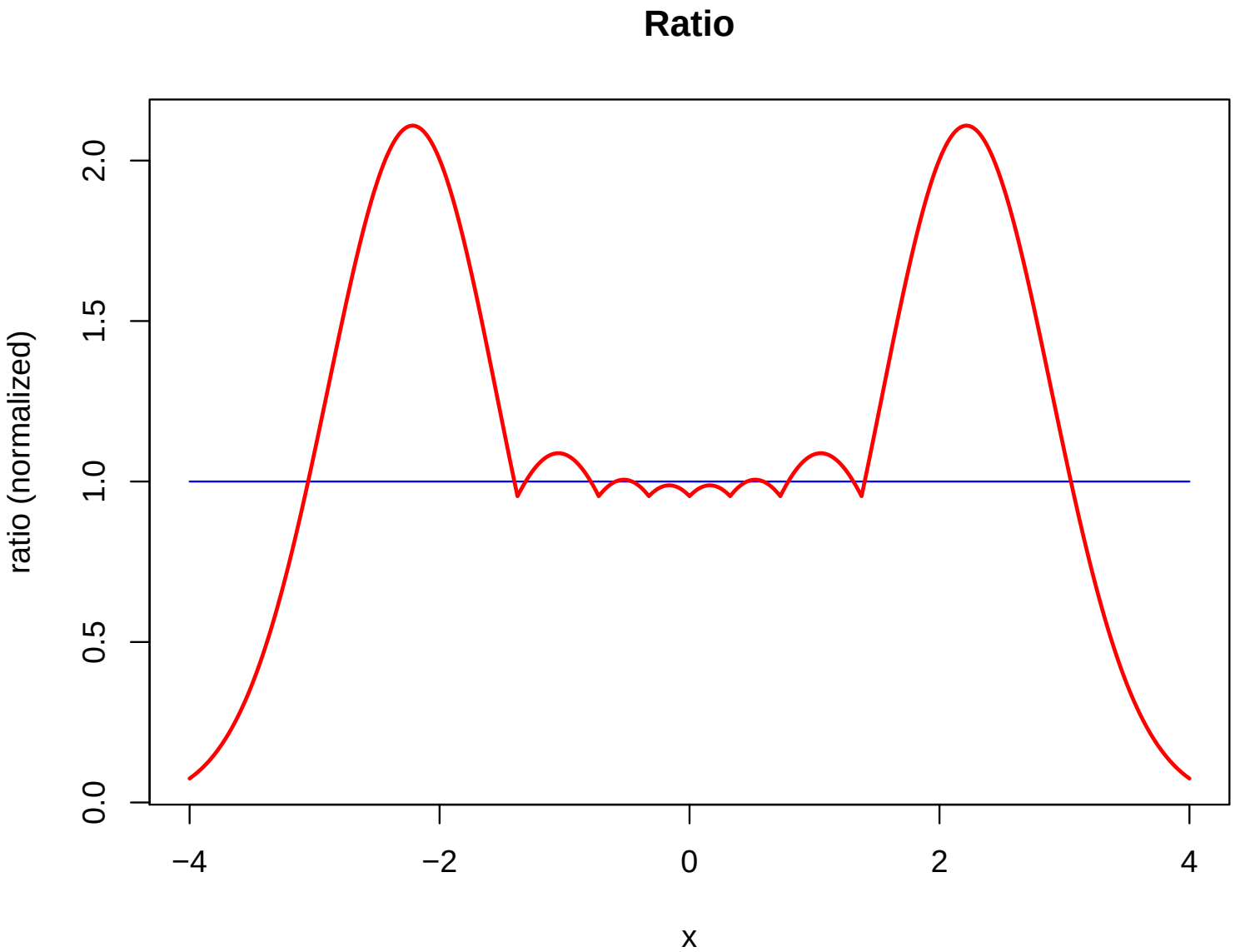
Setup for Approximating Function

- Interval boundaries $x_1 < x_1 < \ldots < x_K$
  (for not necessarily finite domain (x_0, x_K).)
- Interpolating funtion h_i on subinterval $[x_{i-1}, x_i]$.
- Approximating Function $h(x) = \sum_{i=1}^{K} h_i(x)\mathbf{1}_{[x_{i-1},x_i]}$.
- Integrals $A_i = \int_{x_{i-1}}^{x_i} h_i(x)dx$ and $A = \sum_{i=1}^{K} A_i$.
- Approximating density $\tilde{g}(x) = \frac{1}{A}h(x)$.
- Corresponding CDF $\tilde{G}(x) = \int_{-\infty}^{x} \tilde{g}(t)dt$.
  (piecewise for each subinterval)
- Propabilities $p_i = \frac{A_i}{A}$.

Example: Sum of Lognormals



Example: Sum of Lognormals



Inversion

Sampling from $\tilde{g}$ is fast by means of the inversion method:

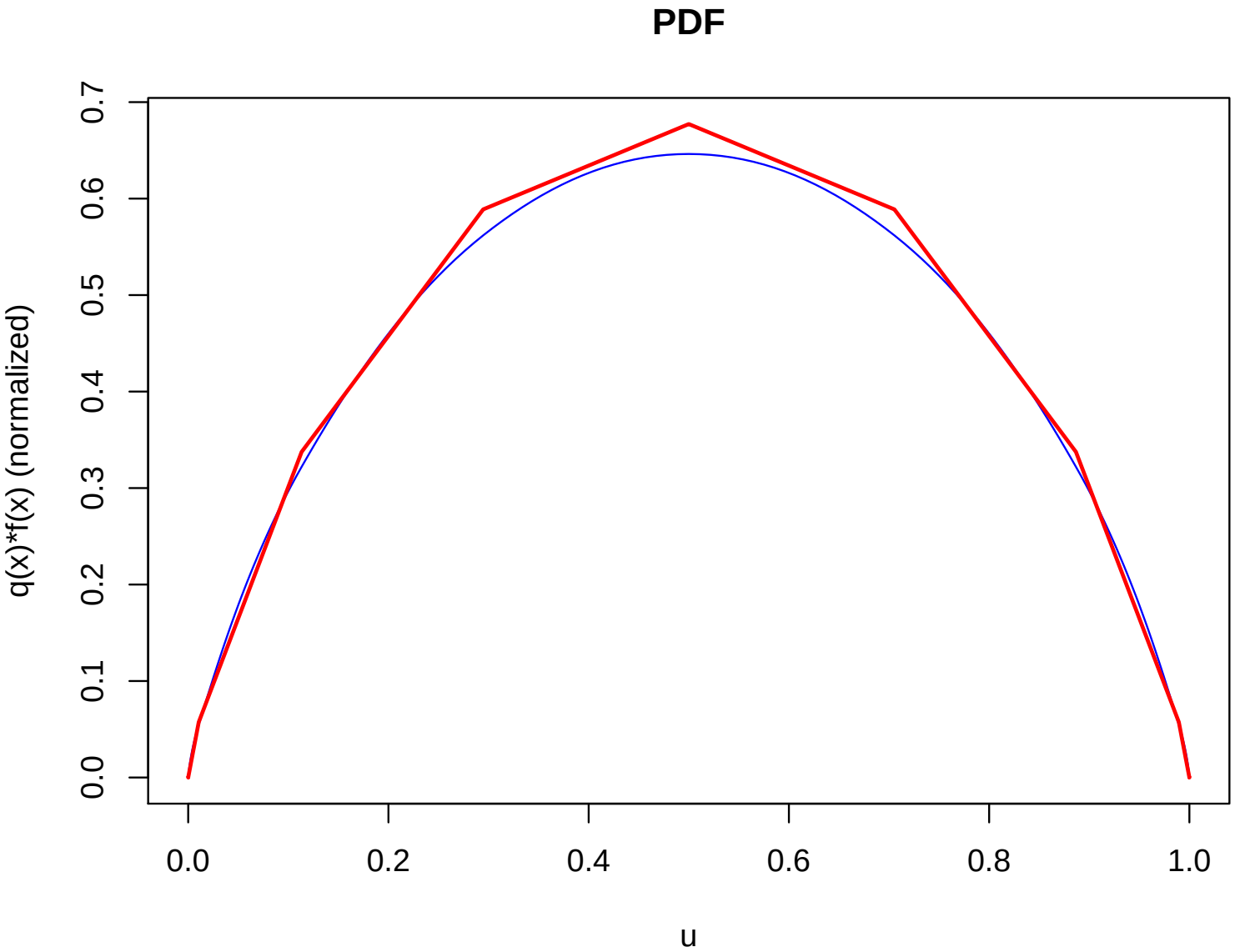
$$X = \tilde{G}^{-1}(U), \quad U \sim \mathcal{U}(0,1)$$

(See below)

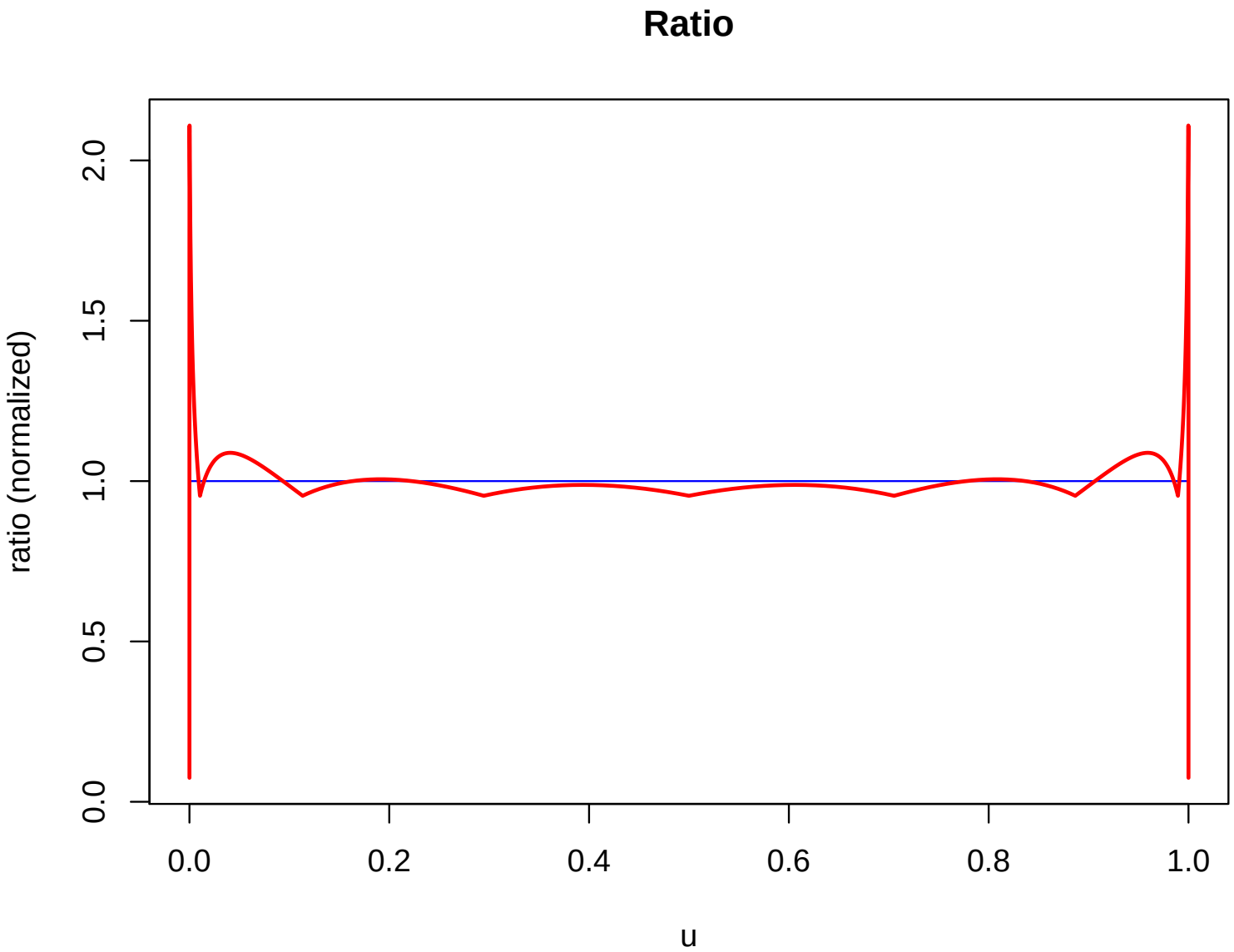
So our original integration problem can be transformed into

$$I = \int_{\mathbb{R}} \frac{q(x)f(x)}{\tilde{g}(x)} d\tilde{G}(x) = \int_0^1 \frac{q(\tilde{G}^{-1}(u))f(\tilde{G}^{-1}(u))}{\tilde{g}(\tilde{G}^{-1}(u))} du$$

Example: Sum of Lognormals



Example: Sum of Lognormals



Improving $\tilde{g}$

The above figure is created with 10 initial subintervals.

How can we improve this approximation of g?

- Try to decrease the mean square deviation (MSD) of $\tilde{g}$ from the constant function I (the blue curve).

  This is exactly $\mathbb{V}_{\tilde{g}}\left(q(x)f(x)/\tilde{g}(x)\right)$ which we want to minimize.

- Try to minimize Kullback-Leibler divergence as there are connections to the expected integration error.
  (see, e.g., Chatterjee and Diaconis, 2018)

Since our motivation is variance reduction we discuss MSD here.

Minimizing MSD

Obviously

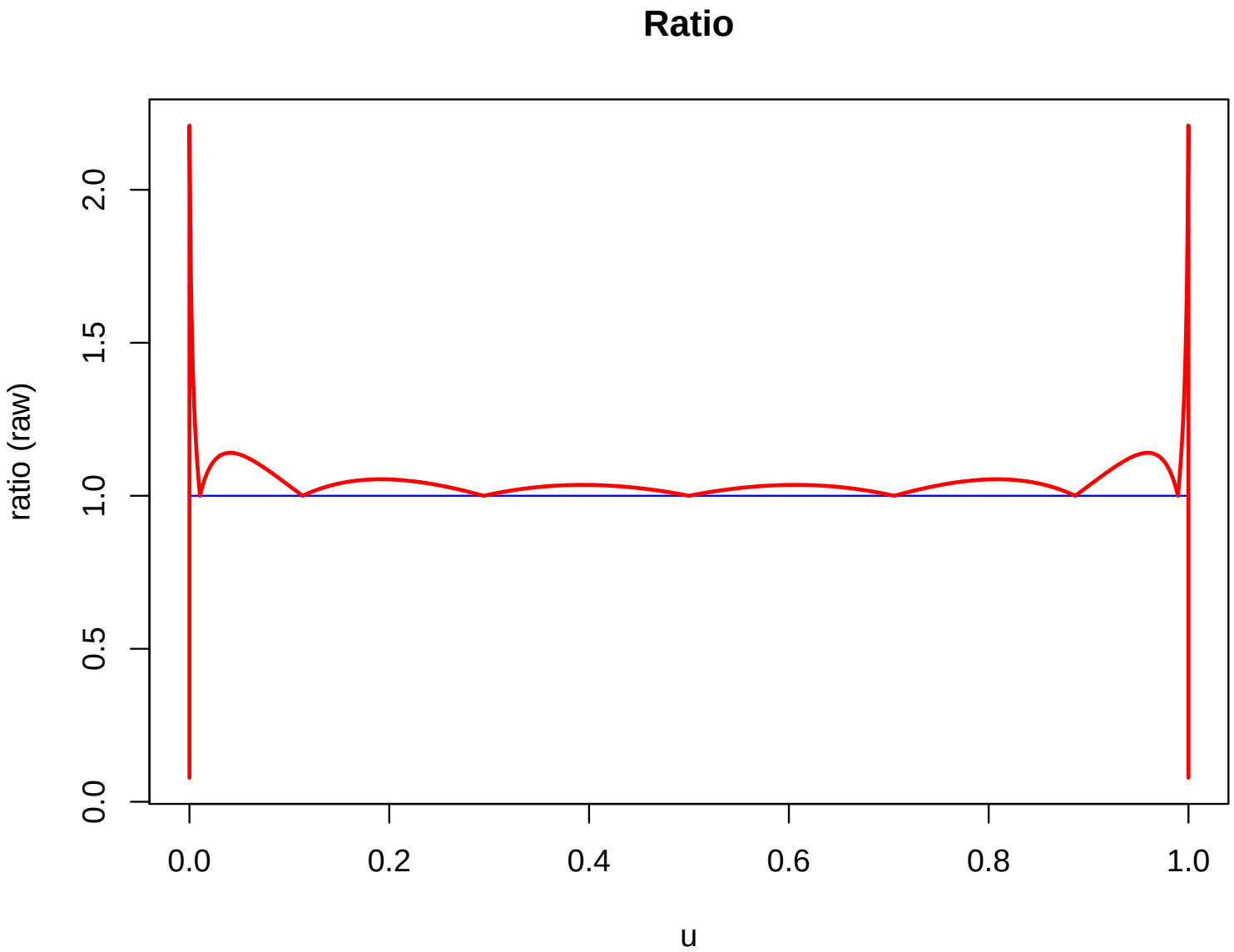
$$\mathbb{V}_{\tilde{g}}[g(X)/\tilde{g}(X)] = \int_0^1 \left(\frac{g(\tilde{G}^{-1}(u))}{\tilde{g}(\tilde{G}^{-1}(u))} - I \right)^2 du$$

is not available.

However, we can use the MSD of $\tilde{g}$ from the constant function A (the blue curve on next slide):

$$\begin{aligned} MSD(g, \tilde{g}) &= \int_{\mathbb{R}} \left(\frac{g(x)}{\tilde{g}(x)} - A \right)^2 d\tilde{G}(x) \\ &= \mathbb{V}_{\tilde{g}}[g(X)/\tilde{g}(X)] + (I - A)^2 \\ &\geq \mathbb{V}_{\tilde{g}}[g(X)/\tilde{g}(X)] \end{aligned}$$

Example: Sum of Lognormals



Estimating MSD

$$
\begin{aligned}
MSD(g, \tilde{g}) &= \int_{\mathbb{R}} \left(\frac{g(x)}{\tilde{g}(x)} - A \right)^2 d\tilde{G}(x) \\
&= A^2 \int_0^1 \left(\frac{g(\tilde{G}^{-1}(u))}{h(\tilde{G}^{-1}(u))} - 1 \right)^2 du \\
&= A^2 \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} \left(\frac{g(x)}{h(x)} - 1 \right)^2 d\tilde{G}(x) \\
&\approx A^2 \sum_{i=1}^{n} \frac{A_i}{A} \left(\frac{g(y_i)}{h(y_i)} - 1 \right)^2
\end{aligned}
$$

for some appropriate points $y_i \in (x_{i-1}, x_i)$.

Adaptively Split Intervals

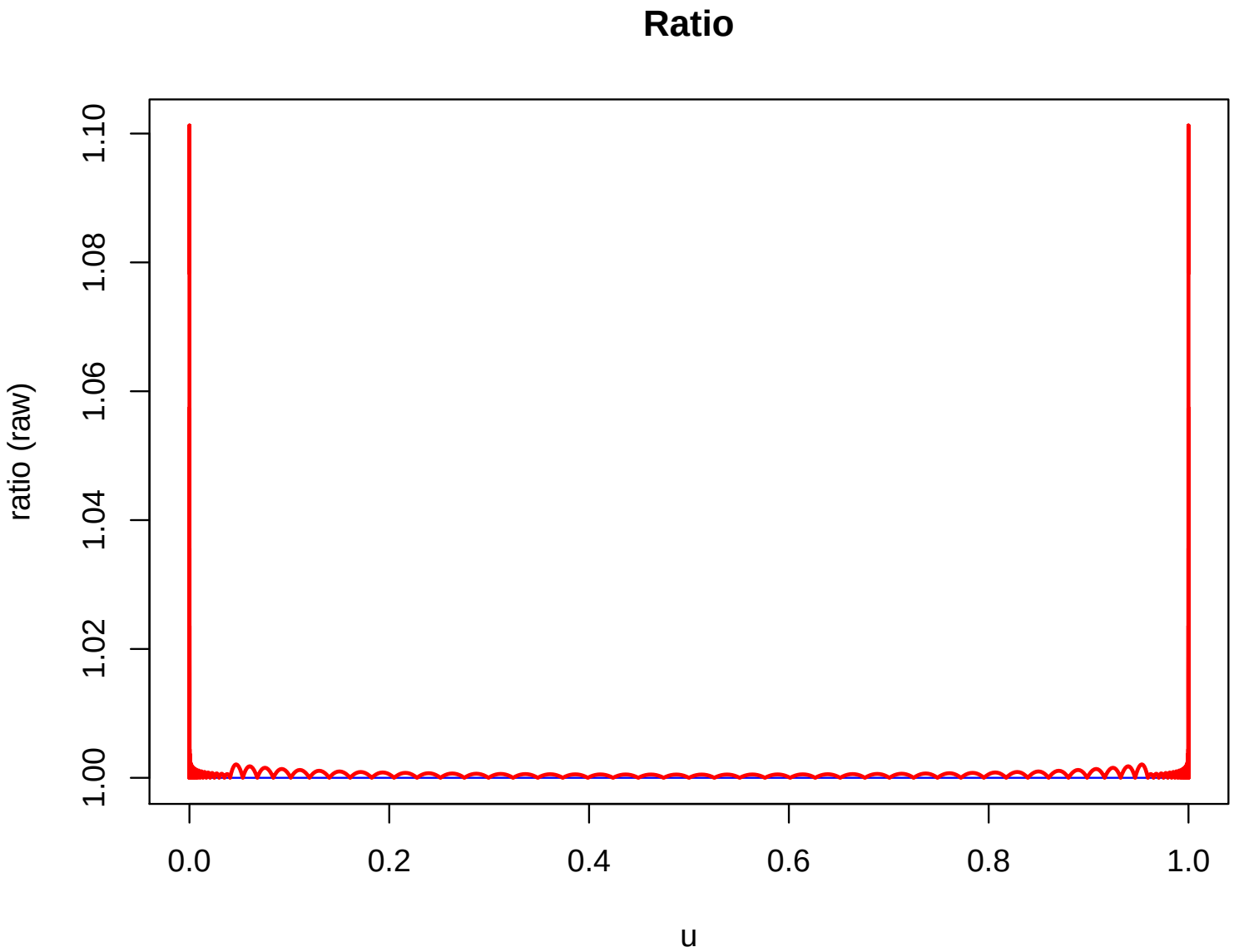
We can improve approximation $\tilde{g}$ by the following procedure:

1. Compute
$$V_i = A_i \left(\frac{g(y_i)}{h(y_i)} - 1 \right)^2$$
for each subinterval, where y_i is some "central point" in subinterval (x_{i-1}, x_i).

2. Split all subintervals where V_i is maximal among all intervals.

3. Repeat this procedure until a requested maximal MSD is reached.

Example: Sum of Lognormals



Remarks

- Observe that $\sum_{i=1}^{K} AV_i \approx MSD(g, \tilde{g})$.
  So we also get an estimate for $\mathbb{V}_{\tilde{g}}[g(X)/\tilde{g}(X)]$.

- From our experiences with TDR with conjecture that $(A - I)^2$ also converges to 0 for $K \to \infty$.

- A bad choice of y_i in the estimator V_i may result in an underestimate of $MSD(g, \tilde{g})$.

  So the result is not as accurate as requested
  but not incorrect.

  Recall: we are allowed to make small errors.

Inversion – Classical

The "classical" approach for inversion is

1. Draw a random index J with probability vector $(p_1, \ldots, p_K)$. (in constant time using guide table method)
2. Draw uniform random U (reuse from 1.)
3. Compute $\tilde{G}^{-1}(U)$ in subinterval $[x_{i-1}, x_i]$.
4. Repeat from Step 1 until required sample size is reached.
5. Compute mean $\hat{\mu}$ and variance $\hat{\sigma}^2$ for this sample.

However, Step 1 requires a table look and thus the creating of that table.

Inversion – Multinomial

However, we just need a **set** of IID random numbers not a sequence. So we can reorder it anyway.

Simpler approach: (no guide table)

1. Draw random vector $N = (N_1, \ldots, N_n)$ from the multinomial distribution with probabilities $(p_1, \ldots, p_n)$ and $\sum_{i=1}^{n} N_i$ is the requested sample size.
2. For each subinterval i draw a random sample $X_{i1}, \ldots, X_{iN_i}$ of size N_i using inversion via $\tilde{G}_i^{-1}$.
3. Compute mean $\hat{\mu}$ and variance $\hat{\sigma}^2$ for the union of these samples.

Stratified Sampling

There is a quite natural way for stratified sampling:

1. Set $N = (\lfloor nA_1/A \rfloor, \ldots, \lfloor nA_n/A \rfloor)$
2. For each subinterval i draw a random sample $X_{i1}, \ldots, X_{iN_i}$ of size N_i using inversion via $\tilde{G}_i^{-1}$.
3. Compute mean $\hat{\mu}_i$ and variance $\hat{\sigma}_i^2$ for each subinterval.
4. Compute $\hat{\mu} = \sum_{j=1}^{K} A_i/A \; \hat{\mu}_i$ and $\hat{\sigma}^2 = \sum_{j=1}^{K} (A_i/A)^2 \; \hat{\sigma}_i^2/N_i$.

Remark: Instead of Step 1 we can use the simple estimate V_i to get a simple estimate for the optimal strata size.

Some Experiments

S2LOG (Sum of 2 lognormal):

$$f(x)q(x) = \phi(x)\Phi\left(-\sqrt[4]{2}\log\left(\left(\exp(x)+\exp(-x)\right)/2\right)\right)$$

ECALL (European call)

$$f(x)q(x) = \phi(x)\,(x-3)^{+}$$

TTAIL (Tail probability of t-distribution)

$$f(x)q(x) = t_5(x)\,\mathbf{1}_{x\geq 3}$$

where t_5 is the PDF of the t-distrbution with 5 degrees of freedom.

Results

	$\text{MSD}_{\max}$	# intervals	VRF
S2LOG	10^{-4}	28	3.38e+04
S2LOG	10^{-6}	90	3.01e+06
S2LOG	10^{-8}	287	2.92e+08
ECALL	10^{-4}	38	3.27e+08
ECALL	10^{-6}	99	2.69e+10
ECALL	10^{-8}	283	2.75e+12
TTAIL	10^{-4}	42	3.41e+07
TTAIL	10^{-6}	111	1.12e+09
TTAIL	10^{-8}	316	1.52e+11

Link

Yuji Nakatsukasa's talk:

Variance reduction in Monte Carlo integration via function approximation

Thank You
for your attention!