Stat 451 Lecture Notes 05[12]
Simulating Random Variables

Ryan Martin
UIC
www.math.uic.edu/~rgmartin

[1]Based on Chapter 6 in Givens & Hoeting, Chapter 22 in Lange, and Chapter 2 in Robert & Casella

[2]Updated: March 7, 2016

Outline

Motivation

- Simulation is a very powerful tool for statisticians.
- It allows us to investigate the performance of statistical methods before delving deep into difficult theoretical work.
- At a more practical level, integrals themselves are important for statisticians:
  - p-values are nothing but integrals;
  - Bayesians need to evaluate integrals to produce posterior probabilities, point estimates, and model selection criteria.
- Therefore, there is a need to understand simulation techniques and how they can be used for integral approximations.

Basic Monte Carlo

- Suppose we have a function $\varphi(x)$ and we'd like to compute $\mathrm{E}\{\varphi(X)\} = \int \varphi(x) f(x)\, dx$, where $f(x)$ is a density.
- There is no guarantee that the techniques we learn in calculus are sufficient to evaluate this integral analytically.
- Thankfully, the law of large numbers (LLN) is here to help:

  If $X_1, X_2, \ldots$ are iid samples from $f(x)$, then
  $\frac{1}{n}\sum_{i=1}^{n} \varphi(X_i) \to \int \varphi(x) f(x)\, dx$ with prob 1.

- Suggests that a generic approximation of the integral be obtained by sampling lots of X_i's from $f(x)$ and replacing integration with averaging.
- This is the heart of the *Monte Carlo method*.

What follows?

- Here we focus mostly on simulation techniques.
- Some of these will be familiar, others probably not.
- As soon as we know how to produce samples from a distribution, the basic Monte Carlo above can be used to approximate any expectation.
- But there are problems where it is not possible to sample from a distribution exactly.
- We'll discuss this point more later.

Outline

Generating uniform RVs

- Generating a single U from a uniform distribution on $[0, 1]$ seems simple enough.
- However, there are a number of concerns to be addressed.
- For example, is it even possible for a computer, which is precise but ultimately discrete, to produce *any* number between 0 and 1?
- Furthermore, how can a deterministic computer possibly generate anything that's *really* random?
- While it's important to understand that these questions are out there, we will side-step them and assume that calls of `runif` in R produce bona fide uniform RVs.

Inverse cdf transform

- Suppose we want to simulate X whose distribution has a given cdf F, i.e., $\frac{d}{dx}F(x) = f(x)$.
- If F is continuous and strictly increasing, then F^{-1} exists.
- Then sampling $U \sim \text{Unif}(0, 1)$ and setting $X = F^{-1}(U)$ does the job — *can you prove it?*
- This method is (sometimes) called the *inversion method*.
- The assumptions above can be weakened to some extent.

Example – exponential distribution

- For an exponential distribution with rate λ, we have

$$f(x) = \lambda e^{-\lambda x} \quad \text{and} \quad F(x) = 1 - e^{-\lambda x}.$$

- It is easy to check that the inverse cdf is

$$F^{-1}(u) = -\log(1-u)/\lambda, \quad u \in (0,1).$$

- Therefore, to sample X from an exponential distribution:
  1. Sample $U \sim \text{Unif}(0,1)$.
  2. Set $X = -\log(1-U)/\lambda$.
- Can be easily "vectorized" to get samples of size n.
- This is in the R function `rexp` — be careful about rate vs. scale parametrization.

Example – Cauchy distribution

- The standard Cauchy distribution has pdf and cdf

$$f(x) = \frac{1}{\pi(1+x^2)} \quad \text{and} \quad F(x) = 1/2 + \arctan(x)/\pi.$$

- This distribution has shape similar to normal, but tails are much heavier — Cauchy has *no finite moments*.
- But its cdf can be inverted:

$$F^{-1}(u) = \tan[\pi(u - 1/2)], \quad u \in (0, 1).$$

- To generate X from standard Cauchy:
  - Sample $U \sim \text{Unif}(0, 1)$.
  - Set $X = \tan[\pi(U - 1/2)]$.
- Non-standard Cauchy (location μ and scale σ): $\mu + \sigma X$.
- Use `rt(n, df=1)` in R.

Example – discrete uniform distribution

- Suppose we want X to be sampled uniformly from $\{1, \ldots, N\}$.
- Here is an example where the cdf is neither continuous nor strictly increasing.
- The idea is as follows:
  1. Divide up the interval $[0, 1]$ into N equal subintervals; i.e., $[0, 1/N), [1/N, 2/N)$ and so forth.
  2. Sample $U \sim \text{Unif}(0, 1)$.
  3. If $i/N \leq U < (i+1)/N$, then $X = i + 1$.
- More simply, set $X = \lfloor nU \rfloor + 1$.
- This is equivalent to `sample(N, size=1)` in R.

Example – triangular distribution

- The (symmetric!) pdf of X is given by

$$f(x) = \begin{cases} 1 + x & \text{if } -1 \leq x < 0, \\ 1 - x & \text{if } 0 \leq x \leq 1. \end{cases}$$

- If we restrict X to $[0, 1]$, then the cdf is simply

$$\tilde{F}(x) = 1 - (1 - x)^2, \quad x \in [0, 1].$$

- For this "sub-problem" the inverse is

$$\tilde{F}^{-1}(u) = 1 - \sqrt{1 - u}, \quad u \in [0, 1].$$

- To sample X from the triangular distribution:
  1. Sample $U \sim \text{Unif}(0, 1)$.
  2. Set $\tilde{X} = 1 - \sqrt{1 - U}$.
  3. Take $X = \pm\tilde{X}$ based on a flip of a coin.

Sampling normal RVs

- While normal RVs can, in principle, be generating using the cdf transform method, this requires evaluation of the standard normal inverse cdf, which is a non-trivial calculation.
- There are a number of fast and efficient alternatives for generating normal RVs.
- The one below, due to Box and Muller, is based on some trigonometric transformations.

Box–Muller method

- This method generates a pair of normal RVs X and Y.
- The method is based on the following facts:
  - The cartesian coordinates (X, Y) are equivalent to the polar coordinates (Θ, R), and the polar coordinates have a joint pdf

$$(2\pi)^{-1}\, r\, e^{-r^2/2}, \quad (\theta, r) \in [0, 2\pi] \times [0, \infty).$$

  - Then $\Theta \sim \text{Unif}(0, 2\pi)$ and $R^2 \sim \text{Exp}(2)$ are independent.
- So to generate independent normal X and Y:
  1. Sample $U, V \sim \text{Unif}(0, 1)$.
  2. Set $R^2 = -2 \log V$ and $\Theta = 2\pi U$.
  3. Finally, take $X = R \cos \Theta$ and $Y = R \sin \Theta$.
- Take a linear function to get different mean and variance.

Bernoulli RVs

- Perhaps the simplest RVs are Bernoulli RVs – ones that take only values 0 or 1.
- To generate $X \sim \text{Ber}(p)$:
  1. Sample $U \sim \text{Unif}(0, 1)$.
  2. If $U \leq p$, then set $X = 1$; otherwise set $X = 0$.
- In R, use `rbinom(n, size=1, prob=p)`.

Binomial RVs

- Since $X \sim \text{Bin}(n, p)$ is distributionally the same as $X_1 + \cdots + X_n$, where the X_i's are independent $\text{Ber}(p)$ RVs, the previous slides gives a natural strategy to sample X.
- That is, to sample $X \sim \text{Bin}(n, p)$, generate $X_1, \ldots, X_n$ independently from $\text{Ber}(p)$ and set X equal to their sum.

Poisson RVs

- Poisson RVs can be constructed from a Poisson process, an integer-valued continuous time stochastic process.
- By definition, the number of events of a Poisson process in a fixed interval of time is a Poisson RV with mean proportional to the length of the interval.
- But the time between events are independent exponentials.
- Therefore, if $Y_1, Y_2, \ldots$ are independent Exp(1) RVs, then

$$X = \max\left\{k : \sum_{i=1}^{k} Y_i \leq \lambda\right\}$$

  then $X \sim \text{Pois}(\lambda)$.
- In R, use `rpois(n, lambda)`.

Chi-square RVs

- The chi-square RV X (with n degrees of freedom) is defined as follows:
  - $Z_1, \ldots, Z_n$ are independent $\mathrm{N}(0, 1)$ RVs.
  - Take $X = Z_1^2 + \cdots + Z_n^2$.
- Therefore, to sample $X \sim \mathrm{ChiSq}(n)$ take the sum of squares of n independent standard normal RVs.
- Independent normals can be sampled using Box–Muller.

Student-t RVs

- A Student-t RV X (with ν degrees of freedom) is defined as the ratio of a standard normal and the square root of an independent (normalized) chi-square RV.
- More formally, let $Z \sim \mathrm{N}(0, 1)$ and $Y \sim \mathrm{ChiSq}(\nu)$; then

$$X = Z / \sqrt{Y/\nu}$$

  is a t_ν RV.
- Remember the scale mixture of normals representation...?
- In R, use `rt(n, df=nu)`.

Multivariate normal RVs

- The p-dimensional normal distribution has a mean vector $\boldsymbol{\mu}$ and a $p \times p$ variance-covariance matrix $\boldsymbol{\Sigma}$.
- The techniques above can be used to sample a vector $Z = (Z_1, \ldots, Z_p)'$ of independent normal RVs.
- But how to incorporate the dependence contained in $\boldsymbol{\Sigma}$?
- Let $\boldsymbol{\Sigma} = \boldsymbol{\Omega}\boldsymbol{\Omega}'$ be the Cholesky decomposition of $\boldsymbol{\Sigma}$.
- It can be shown that $X = \boldsymbol{\mu} + \boldsymbol{\Omega} Z$ is the desired p-dimensional normal distribution.
- Can you prove it?

Outline

Intuition

- Let f be a density function on an arbitrary space $\mathbb{X}$; the goal is to simulate from f.
- Note the trivial identity:

$$f(x) = \int_0^{f(x)} du.$$

- This identity implicitly introduces an *auxiliary variable* U with a conditionally uniform distribution.
- The intuition behind this viewpoint is that simulating from the joint distribution of (X, U) might be easy, and then we can just throw away U to get a sample of $X \sim f$...

The "theorem"

- **"Theorem."** Simulating $X \sim f$ is equivalent to simulating

$$(X, U) \sim \text{Unif}(\{(x, u) : 0 < u < f(x)\})$$

  and then throwing away U.
- *Proof:* Write the density of (X, U) and integrate out U.
- How to implement this?
  - One kind-of silly[3] idea:

$$X \sim f \quad \text{and} \quad U \mid (X = x) \sim \text{Unif}(0, f(x)).$$

  - A better idea: "conditioning preserves uniformity," i.e.,

$$Z \sim \text{Unif}(\mathbb{Z}) \implies Z \mid (Z \in \mathbb{Z}_0) \sim \text{Unif}(\mathbb{Z}_0).$$

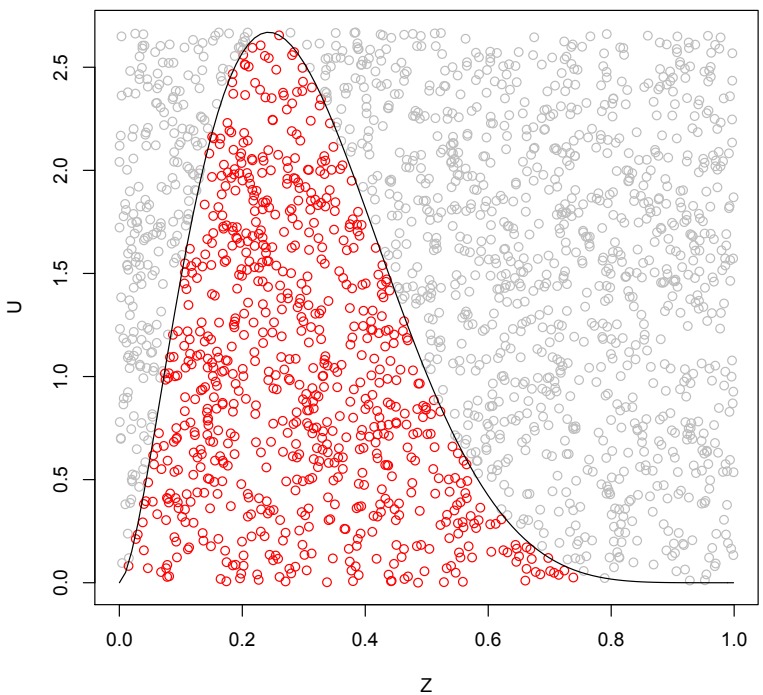
[3] Not actually silly, it's used in the accept–reject method...

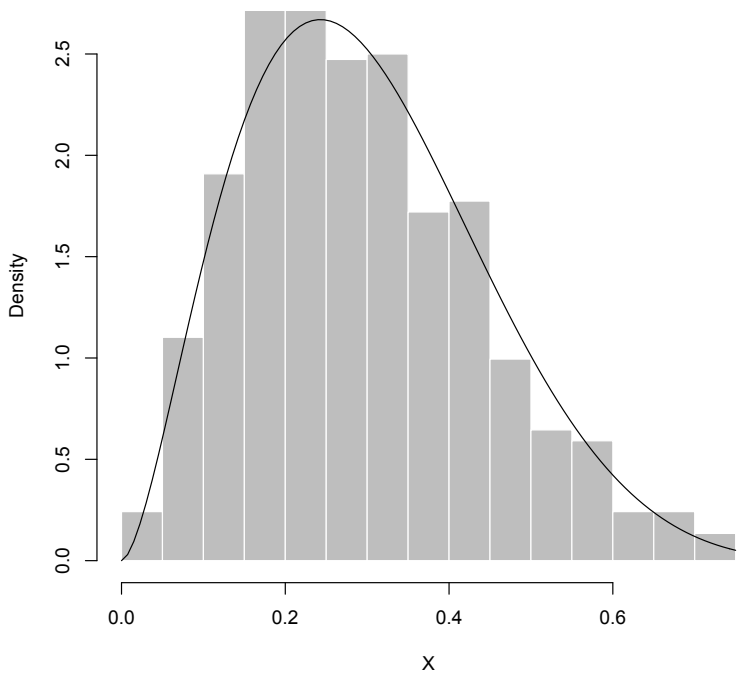
More on implementation

- The "conditioning preserves uniformity" point can be interpreted as follows.
  - Suppose that A is a set that contains $\{(x, u) : 0 < u < f(x)\}$.
  - Simulate (X, U) uniformly on A, and keep (X, U) only if $U < f(X)$.
  - Such a pair (X, U) is uniformly distributed on the constraint set, so X has the desired distribution f.
- Efficiency of sampling depends on how tightly A fits the desired constraint set.
- For a one-dimensional X, with bounded support and bounded f, a reasonable choice for A is a rectangle; see below.
- Idea extends — this is what the next sections are about!

Example – beta simulation

- Simulate from a Beta$(2.7, 6.3)$ distribution.
- Start with uniforms in a box but keep only those that satisfy the constraint.
- Simulated 2000 uniforms, kept only 744 betas.





Outline

Outline

Majorization

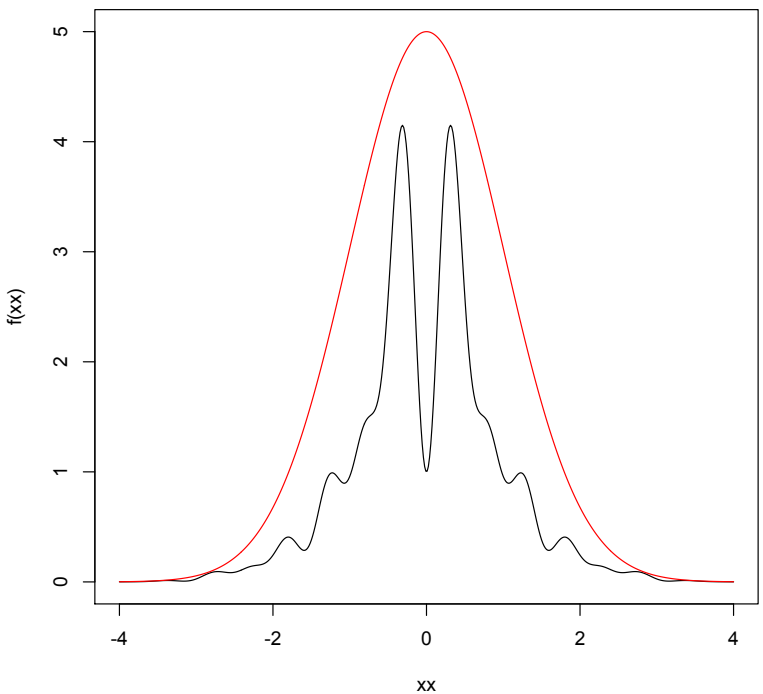
- Suppose we want to sample from a distribution with pdf $f(x)$.
- Suppose, further, that $f(x) \leq h(x) := Mg(x)$ where $g(x)$ is a (simple) pdf and M is a constant > 1.
- We say $h(x)$ majorizes $f(x)$.
- The goal is to use samples from the (easy to sample) pdf $g(x)$ as "approximate samples" from $f(x)$.
- But it is clear that unless $g(x) \approx f(x)$, there will be samples from g that are not representative of f.
- The idea is to throw away those "bad" samples.

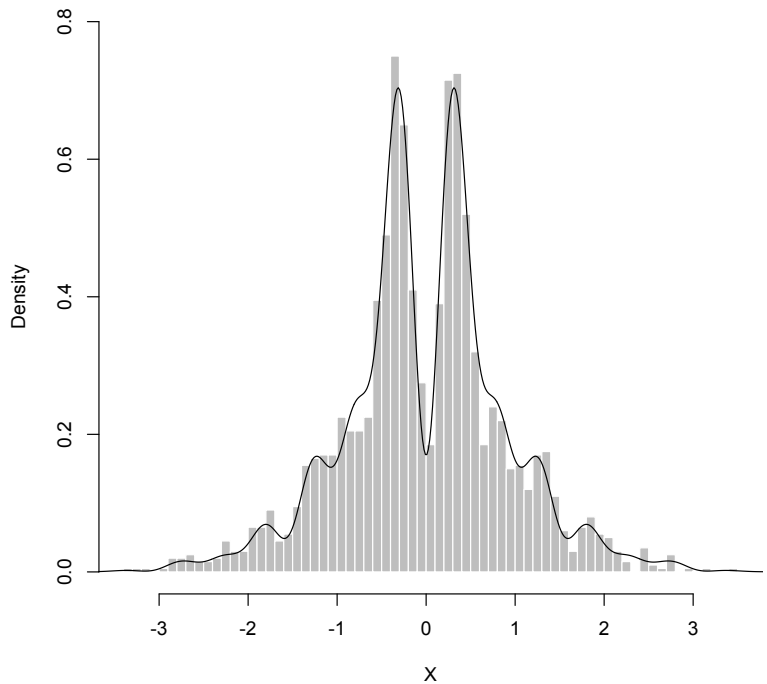
Acceptance–rejection method

- The rule that determines when a sample is "thrown away" is called the *acceptance–rejection method*.
- To approximately sample X from $f(x)$:
  1. Sample $U \sim \text{Unif}(0, 1)$.
  2. Sample $X' \sim g(x)$.
  3. Keep $X = X'$ if $U \leq f(X')/h(X')$
- Try to prove the following:
  - accept–reject returns a sample from f (see "theorem");
  - acceptance probability is $1/M$.
- Goal is to make acceptance prob high, i.e., $M \approx 1$... (?)
- Note: it is not necessary to know f exactly—it's enough to know f only up to a proportionality constant.

Example – Robert & Casella book cover

- True density is a weird trig function – see code.
- Majorant is a normal.
- Histogram of $n = 2000$ samples from trig density.





Example – von Mises distribution

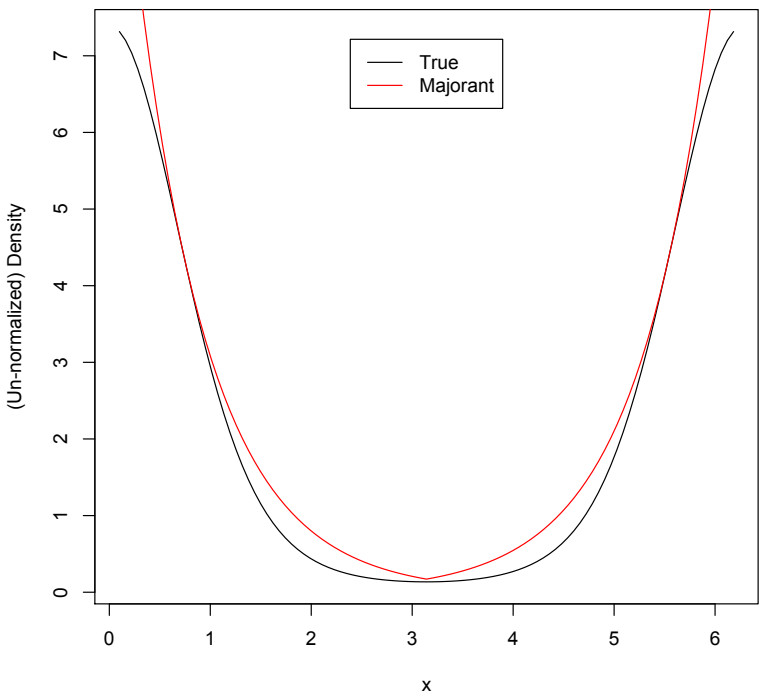
- Consider the von Mises distribution with pdf

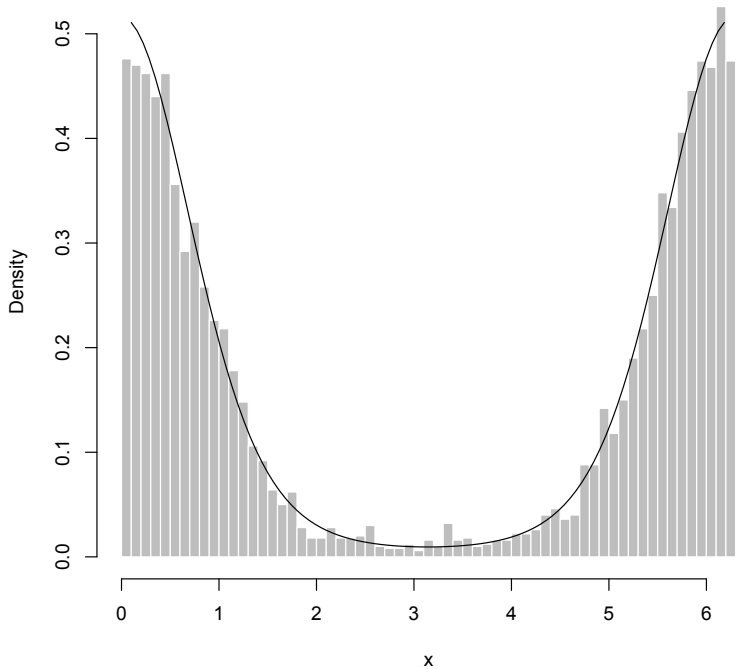
$$f(x) = \frac{\exp\{-\kappa \cos x\}}{2\pi I_0(\kappa)}, \quad x \in [0, 2\pi],$$

  where $I_0(\kappa)$ is a Bessel function.
- This distribution is often referred to as a circular normal distribution, and is a popular model in *directional statistics*.
- To sample from the von Mises distribution for fixed κ, we implement the acceptance-rejection method with majorant made out of exponential pdfs.
- Only trick is to get a "good" envelope/majorant.
- See code for more details.

von Mises distribution (cont.)

- von Mises distribution with $\kappa = 2.$[4]
- Uses two oppositely oriented exponentials for the envelope.
- $n = 5000$ samples; acceptance rate (for $\kappa = 2$) is ≈ 0.81.





[4]For some reason, majorant construction in my code only works for $\kappa > 1$...

Example – small-shape gamma distribution

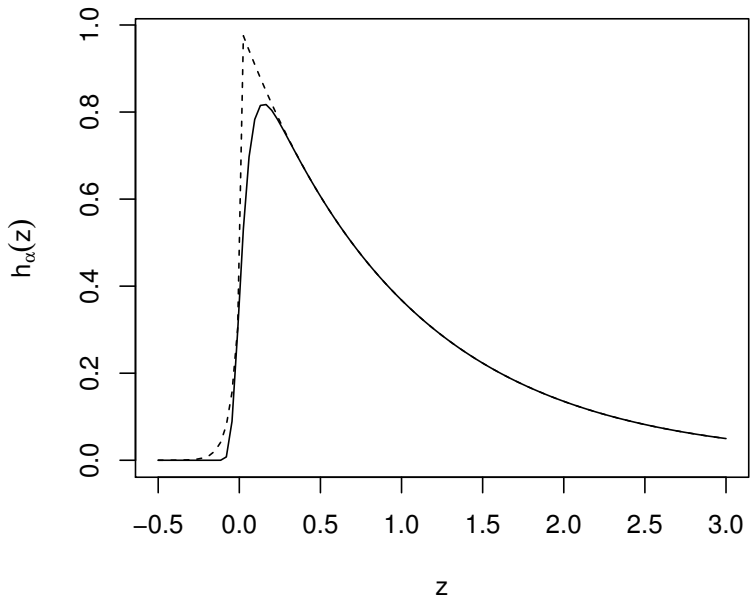
- Suppose we want $X \sim \text{Gamma}(\alpha, 1)$, with $\alpha \leq 0.001$.
- Try using `rgamma` for this—you'll get lots of *exact zeros!*
- Can we develop a better/more efficient method?
- Towards an accept–reject method, we have the following:

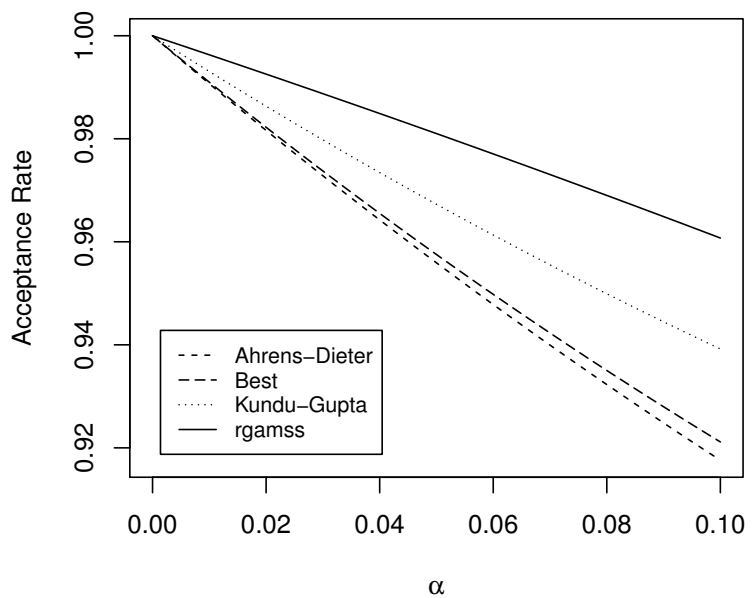
$$-\alpha \log X \to \text{Exp}(1) \quad \text{in distribution, as } \alpha \to 0.$$

- Suggests that we can get good samples of $\log X$ by doing accept–reject with (basically) an exponential envelope.
- This is some work of mine:
  - a version of the paper is at `arXiv:1302.1884`;
  - R code is on my research website.

Example – small-shape gamma (cont.)

- Left panel shows (un-normalized) density of $Z = -\alpha \log X$, for $\alpha = 0.005$, along with the proposed envelope.
- It turns out that this approach is more efficient than other accept–reject methods for this problem, based on acceptance rate (as a function of α).





Outline

Bounding set

- Suppose we want to sample from a pdf $f(x)$.
- All that matters is the shape of $f(x)$, so we can remove any constants and consider $h(x) = cf(x)$, for some $c > 0$.
- Define the set in $\mathbb{R}^2$:

$$\mathcal{S}_h = \left\{(u, v) : 0 < u \leq h(v/u)^{1/2}\right\}.$$

- If $\mathcal{S}_h$ is bounded, then we can find a bounding set that encloses it.
- Ideally, the bounding set should be simple, e.g., a rectangle.

Ratio method

- Suppose $\mathcal{S}_h$ is bounded, in which case we can find a rectangle that encloses it.
- To sample X from $f(x)$, the ratio method goes as follows:
  1. Sample (U, V) uniformly from the bounding rectangle.
  2. If $(U, V) \in \mathcal{S}_h$, then $X = V/U$ is a sample from $f(x)$.
- The proof is not too hard, but requires some tedious jacobian-type of calculations; see Lange.
- Naturally, some draws from the rectangle will be rejected – the efficiency of the sampling algorithm depends on how closely the bounding set matches $\mathcal{S}_h$.

Example – gamma distribution

- Sample $X \sim \text{Gamma}(\alpha, 1)$ for non-integer α.
- To apply the ratio method, take $h(x) = x^{\alpha-1}e^{-x}$ for $x > 0$.
- It can be shown that, in general, if $h(x) = 0$ for $x < 0$, then the rectangle $[0, k_u] \times [0, k_v]$, with

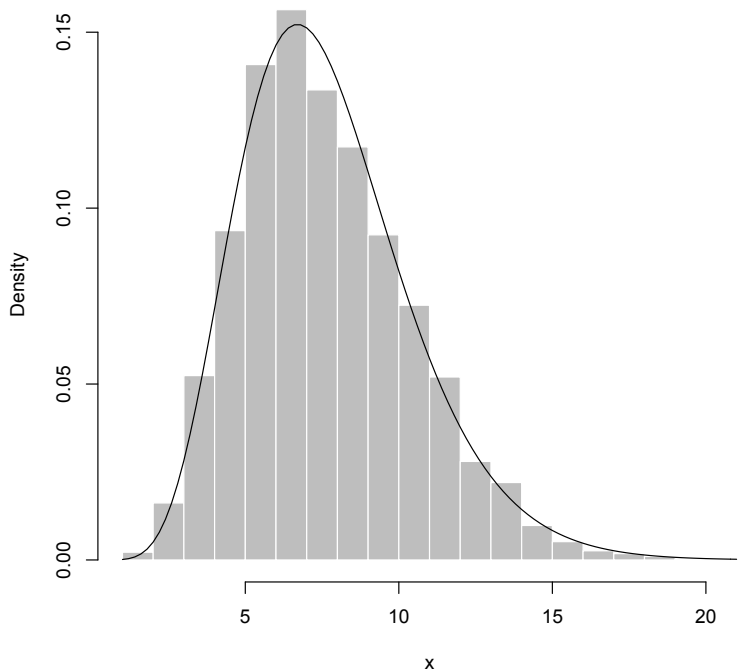
$$k_u = \sup_x h(x)^{1/2} \quad \text{and} \quad k_v = \sup_x \{|x| h(x)^{1/2}\},$$

  encloses $\mathcal{S}_h$.
- For the gamma case,

$$k_u = [(\alpha - 1)/e]^{(\alpha-1)/2} \quad \text{and} \quad k_v = [(\alpha + 1)/e]^{(\alpha+1)/2}.$$

Example – gamma distribution (cont.)

- Ratio method samples $X \sim \text{Gamma}(\alpha, 1)$ as follows:
  1. Sample $U', V' \sim \text{Unif}(0, 1)$, set $U = k_u U'$ and $V = k_v V'$.
  2. Set $X = V/U$.
  3. If $U \leq X^{(\alpha-1)/2} e^{-X/2}$, then accept X.
- Example: $\alpha = 7.7$.
- Ratio method has acceptance probability ≈ 0.44.



Outline

Motivation

- Previous methods are "exact" in the sense that the distribution of the draw X (given that it's accepted) is the target distribution f.
- Is it necessary for the sampling to be exact?
- An interesting idea is to sample from a different distribution $g(x)$, which is similar to $f(x)$, and weight these samples in such a way that a resample according to the given weights looks like a sample from $f(x)$.
- This is the idea of *sampling importance resampling* (SIR).
- Not unlike acceptance–rejection sampling.

SIR algorithm

- Suppose that the target pdf is $f(x)$, which may be known only up to a proportionality constant.
- Let $g(x)$ be another pdf with the same support as $f(x)$.
- SIR algorithm:
  1. Take an independent sample $Y_1, \ldots, Y_m$ from g.
  2. Calculate the *standardized importance weights*

$$w(Y_j) \propto f(Y_j)/g(Y_j), \quad j = 1, \ldots, m.$$

  3. Resample $X_1, \ldots, X_n$ with replacement from $\{Y_1, \ldots, Y_m\}$ with probabilities $w(Y_1), \ldots, w(Y_m)$.
- The resulting sample $X_1, \ldots, X_n$ is *approximately* distributed according to $f(x)$.

Remarks

- One can prove that, as $m \to \infty$, $\mathrm{P}(X_1 \in A) \to \int_A f(x)\,dx$, and it is in this sense that the sampling is approximate.
- The choice of envelope g is not trivial.
  - Need $f(x)/g(x)$ to not be too large.
  - If this is violated, then it may happen that one $w(Y_j)$ will be almost identically 1, meaning that the X-sample would all be the same value.
- Theory suggests m should be large, but "large" here depends on the desired size n of the sample from the target.
- In general, the Monte Carlo estimates based on the SIR sample will have variances larger than could be obtained with other methods (e.g., importance sampling, next).

Example – Bayesian inference via SIR

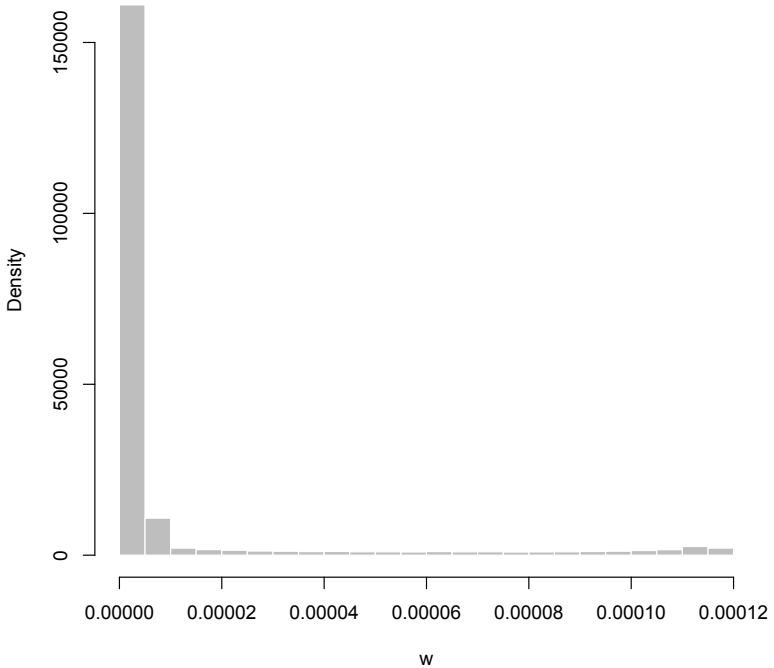
- Consider the problem from old homework, where the likelihood function is

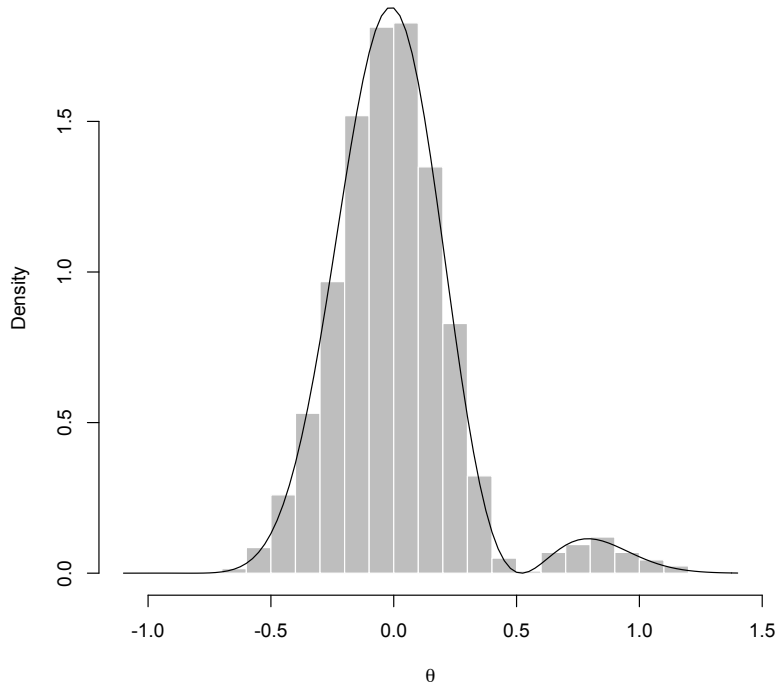
$$L(\theta) \propto \prod_{i=1}^{n} \{1 - \cos(X_i - \theta)\}, \quad -\pi \leq \theta \leq \pi.$$

- Observed data $(X_1, \ldots, X_n)$ given in the code.
- Assume that θ is given a Unif$(-\pi, \pi)$ prior distribution.
- Use SIR algorithm, with the prior as the envelope; $N = 10^3$.
- Plots below.

Example – Bayesian inference via SIR (cont.)

- Left is a histogram of the importance weights.
- Right is a histogram the SIR sample (density overlaid).





Outline

Remarks

- Simulating random variables is important for various applications of the Monte Carlo method.
- Some distributions can be easily simulated via the inversion method, while others require more care.
- "Fundamental theorem" provides a general strategy for simulating from non-standard distributions, though its implementation may not be straightforward.
- The accept–reject method is a clever implementation.
  - Method's efficiency relies on how close the envelope function is to the target density.
  - Not easy to make a good choice.
  - Various automatic/adaptive methods are available.
- The "accept–reject" idea appeared in the SIR context, and will appear again later...