Statistical Models & Computing Methods

Lecture 4: Numerical Integration

Cheng Zhang

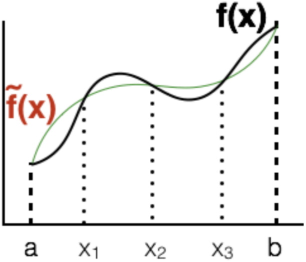
School of Mathematical Sciences, Peking University

September 22, 2025

Overview

- Statistical inference often depends on intractable integrals $I(f) = \int_{\Omega} f(x)dx$
- This is especially true in Bayesian statistics, where a posterior distribution is usually non-trivial.
- In some situations, the likelihood itself may depend on intractable integrals so frequentist methods would also require numerical integration
- In this lecture, we start by discussing some simple numerical methods that can be easily used in low dimensional problems
- Next, we will discuss several Monte Carlo strategies that could be implemented even when the dimension is high

- Consider a one-dimensional integral of the form $I(f) = \int_a^b f(x)dx$
- A common strategy for approximating this integral is to use a tractable approximating function $\tilde{f}(x)$ that can be integrated easily
- We typically constrain the approximating function to agree with f on a grid of points: $x_1, x_2, \ldots, x_n$



- Newton-Côtes methods use equally-spaced grids
- The approximating function is a polynomial
- The integral then is approximated with a weighted sum as follows

$$\hat{I} = \sum_{i=1}^{n} w_i f(x_i)$$

- In its simplest case, we can use the Riemann rule by partitioning the interval $[a, b]$ into n subintervals of length $h = \frac{b-a}{n}$; then

$$\hat{I}_L = h \sum_{i=0}^{n-1} f(a + ih)$$

  This is obtained using a piecewise constant function $\tilde{f}$ that matches f at the left points of each subinterval

- Alternatively, the approximating function could agree with the integrand at the right or middle point of each subinterval

$$\hat{I}_R = h \sum_{i=1}^{n} f(a + ih), \quad \hat{I}_M = h \sum_{i=0}^{n-1} f(a + (i + \frac{1}{2})h)$$

- In either case, the approximating function is a zero-order polynomial
- To improve the approximation, we can use the trapzoidal rule by using a piecewise linear function that agrees with $f(x)$ at both ends of subintervals

$$\hat{I} = \frac{h}{2} f(a) + h \sum_{i=1}^{n-1} f(x_i) + \frac{h}{2} f(b)$$

Newton-Côtes Quadrature

- We would further improve the approximation by using higher order polynomials
- Simpson's rule uses a quadratic approximation over each subinterval

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{x_{i+1}-x_i}{6}\left(f(x_i)+4f(\frac{x_i+x_{i+1}}{2})+f(x_{i+1})\right)$$

- In general, we can use any polynomial of degree k

- Newton-Côtes rules require equally spaced grids
- With a suitably flexible choice of $n+1$ nodes, $x_0, x_1, \ldots, x_n$, and corresponding weights, $A_0, A_1, \ldots, A_n$,

$$\sum_{i=0}^{n} A_i f(x_i)$$

  gives the exact integration for all polynomials with degree less than or equal to $2n+1$
- This is called Gaussian quadrature, which is especially useful for the following type of integrals $\int_a^b f(x)w(x)dx$ where $w(x)$ is a nonnegative function and $\int_a^b x^k w(x)dx < \infty$ for all $k \geq 0$

- In general, for squared integrable functions,

$$\int_a^b f(x)^2 w(x) dx \leq \infty$$

denoted as $f \in \mathcal{L}^2_{w,[a,b]}$, we define the inner product as

$$\langle f, g \rangle_{w,[a,b]} = \int_a^b f(x) g(x) w(x) dx$$

where $f, g \in \mathcal{L}^2_{w,[a,b]}$

- We said two functions to be *orthogonal* if $\langle f, g \rangle_{w,[a,b]} = 0$. If f and g are also scaled so that $\langle f, f \rangle_{w,[a,b]} = 1$, $\langle g, g \rangle_{w,[a,b]} = 1$, then f and g are orthonormal

- We can define a sequence of orthogonal polynomials by a recursive rule

$$T_{k+1}(x) = (\alpha_{k+1} + \beta_{k+1}x)T_k(x) - \gamma_{k+1}T_{k-1}(x)$$

- Example: Chebyshev polynomials (first kind).

$$T_0(x) = 1, \quad T_1(x) = x$$
$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

- $T_n(x)$ are orthogonal with respect to $w(x) = \frac{1}{\sqrt{1-x^2}}$ and $[-1, 1]$

$$\int_{-1}^{1} T_n(x)T_m(x)\frac{1}{\sqrt{1-x^2}}dx = 0, \quad \forall n \neq m$$

Orthogonal Polynomials

- In general orthogonal polynomials are not unique since $\langle f, g \rangle = 0$ implies $\langle cf, dg \rangle = 0$
- To make the orthogonal polynomial unique, we can use the following standarizations
  - make the polynomial orthonormal: $\langle f, f \rangle = 1$
  - set the leading coefficient of $T_j(x)$ to 1
- Orthogonal polynomials form a basis for $\mathcal{L}^2_{w,[a,b]}$ so any function in this space can be written as

$$f(x) = \sum_{n=0}^{\infty} a_n T_n(x)$$

where $a_n = \frac{\langle f, T_n \rangle}{\langle T_n, T_n \rangle}$

- Let $\{T_n(x)\}_{n=0}^{\infty}$ be a sequence of orthogonal polynomials with respect to w on $[a, b]$.
- Denote the $n+1$ roots of $T_{n+1}(x)$ by

$$a < x_0 < x_1 < \ldots < x_n < b.$$

- We can find weights $A_1, A_2, \ldots, A_{n+1}$ such that

$$\int_a^b P(x)w(x)dx = \sum_{i=0}^{n} A_i P(x_i), \quad \forall \deg(P) \leq 2n+1$$

- To do that, we first show: there exists weights $A_1, A_2, \ldots, A_{n+1}$ such that

$$\int_a^b P(x)w(x)dx = \sum_{i=0}^{n} A_i P(x_i), \quad \forall \deg(P) < n+1$$

- Sketch of proof. We only need to satisfy

$$\int_a^b x^k w(x)dx = \sum_{i=0}^{n} A_i x_i^k, \quad \forall\, k = 0, 1, \ldots, n$$

This leads to a system of linear equations

$$\begin{bmatrix} 1 & 1 & \ldots & 1 \\ x_0 & x_1 & \ldots & x_n \\ \vdots & \vdots & \vdots & \vdots \\ x_0^n & x_1^n & \ldots & x_n^n \end{bmatrix} \begin{bmatrix} A_0 \\ A_1 \\ \vdots \\ A_n \end{bmatrix} = \begin{bmatrix} I_0 \\ I_1 \\ \vdots \\ I_n \end{bmatrix}$$

where $I_k = \int_a^b x^k w(x)dx$. The determinant of the coefficient matrix is a Vandermonde determinant, and is non-zero since $x_i \neq x_j, \forall i \neq j$

- Now we show that the above Gaussian Quadrature can be exact for polynomials of degree $\leq 2n+1$
- Let $P(x)$ be a polynomial with $\deg(P) \leq 2n+1$, there exist polynomials $g(x)$ and $r(x)$ such that

$$P(x) = g(x)T_{n+1}(x) + r(x)$$

with $\deg(g) \leq n, \deg(r) \leq n$, Therefore,

$$\begin{aligned}\int_a^b P(x)w(x)dx = \int_a^b r(x)w(x)dx = \sum_{i=0}^{n} A_i r(x_i) \\ = \sum_{i=0}^{n} A_i P(x_i)\end{aligned}$$

Monte Carlo Method

- We now discuss the Monte Carlo method mainly in the context of statistical inference
- As before, suppose we are interested in estimating $I(h) = \int_a^b h(x)dx$
- If we can draw iid samples, $x^{(1)}, x^{(2)}, \ldots, x^{(n)}$ uniformly from (a, b), we can approximate the integral as

$$\hat{I}_n = (b-a)\frac{1}{n}\sum_{i=1}^{n} h(x^{(i)})$$

- Note that we can think about the integral as

$$(b-a)\int_a^b h(x) \cdot \frac{1}{b-a} dx$$

where $\frac{1}{b-a}$ is the density of Uniform(a, b)

- In general, we are interested in integrals of the form $\int_{\mathcal{X}} h(x)f(x)dx$, where $f(x)$ is a probability density function
- Analogous to the above argument, we can approximate this integral (or expectation) by drawing iid samples $x^{(1)}, x^{(2)}, \ldots, x^{(n)}$ from the density $f(x)$ and then

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} h(x^{(i)})$$

- Based on the law of large numbers, we know that

$$\lim_{n\to\infty} \hat{I}_n \xrightarrow{p} I$$

- And based on the central limit theorem

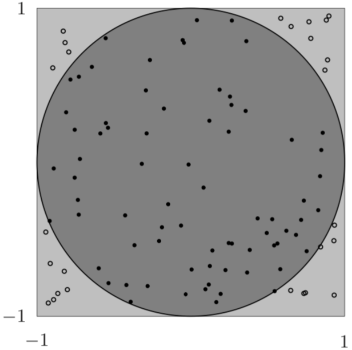
$$\sqrt{n}(\hat{I}_n - I) \to \mathcal{N}(0, \sigma^2), \quad \sigma^2 = \mathbb{V}\mathrm{ar}(h(X))$$

Example: estimating π

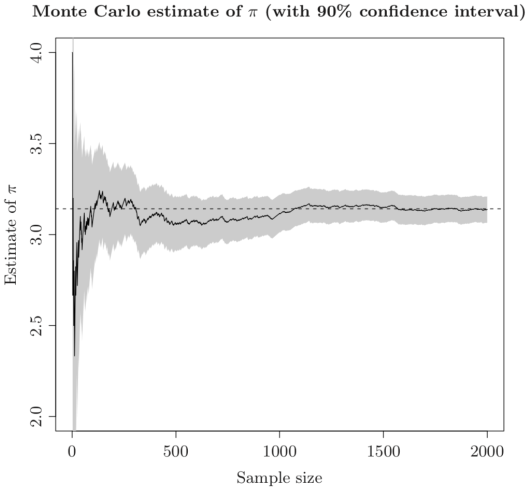
- Let $h(x) = \mathbf{1}_{B(0,1)}(x)$, then $\pi = 4\int_{[-1,1]^2} h(x) \cdot \frac{1}{4}\, dx$
- Monte Carlo estimate of π

$$\hat{I}_n = \frac{4}{n} \sum_{i=1}^{n} \mathbf{1}_{B(0,1)}(x^{(i)})$$

$$x^{(i)} \sim \text{Uniform}([-1,1]^2)$$



Example: estimating π



Monte Carlo vs Quadrature

- Convergence rate for Monte Carlo: $\mathcal{O}(n^{-1/2})$

$$p\left(|\hat{I}_n - I| \leq \frac{\sigma}{\sqrt{n\delta}}\right) \geq 1 - \delta, \quad \forall \delta$$

  often slower than quadrature methods ($\mathcal{O}(n^{-2})$ or better)
- However, the convergence rate of Monte Carlo does not depend on dimensionality
- On the other hand, quadrature methods are difficult to extend to multidimensional problems, because of the curse of dimensionality. The actual convergence rate becomes $\mathcal{O}(n^{-k/d})$, for any order k method in dimension d
- This makes Monte Carlo strategy very attractive for high dimensional problems

Exact Simulation

- Monte Carlo methods require sampling a set of points chosen randomly from a probability distribution
- For simple distribution $f(x)$ whose inverse cumulative distribution functions (CDF) exists, we can sampling x from f as follows

$$x = F^{-1}(u), \quad u \sim \mathrm{Uniform}(0, 1)$$

where F^{-1} is the inverse CDF of f

- Proof.

$$p(a \leq x \leq b) = p(F(a) \leq u \leq F(b)) = F(b) - F(a)$$

- Exponential distribution: $f(x) = \theta \exp(-\theta x)$. The CDF is

$$F(a) = \int_0^a \theta \exp(-\theta x) = 1 - \exp(-\theta a)$$

therefore, $x = F^{-1}(u) = -\frac{1}{\theta} \log(1-u) \sim f(x)$. Since $1-u$ also follows the uniform distribution, we often use $x = -\frac{1}{\theta} \log(u)$ instead

- Normal distribution: $f(x) = \frac{1}{\sqrt{2\pi}} \exp(-\dfrac{x^2}{2})$. Box-Muller Transform

$$X = \sqrt{-2 \log U_1} \cos 2\pi U_2$$
$$Y = \sqrt{-2 \log U_1} \sin 2\pi U_2$$

where $U_1 \sim \text{Uniform}(0,1), \quad U_2 \sim \text{Uniform}(0,1)$

Intuition for Box-Muller Transform

- Assume $Z = (X, Y)$ follows the standard bivariate normal distribution. Consider the following transform

$$X = R\cos\Theta, \quad Y = R\sin\Theta$$

- From symmetry, clearly Θ follows the uniform distribution on the interval $(0, 2\pi)$ and is independent of R
- What distribution does R follow? Let's take a look at its CDF

$$\begin{aligned} p(R \le r) &= p(X^2 + Y^2 \le r^2) \\ &= \frac{1}{2\pi}\int_0^r t\exp(-\frac{t^2}{2})dt \int_0^{2\pi} d\theta = 1 - \exp(-\frac{r^2}{2}) \end{aligned}$$

Therefore, using the inverse CDF rule, $R = \sqrt{-2\log U_1}$

Rejection Sampling

- If it is difficult or computationally intensive to sample directly from $f(x)$ (as described above), we need to use other strategies
- Although it is difficult to sample from $f(x)$, suppose that we can evaluate the density at any given point up to a constant $f(x) = f^*(x)/Z$, where Z could be unknown (remember that this make Bayesian inference convenient since we usually know the posterior distribution only up to a constant)
- Furthermore, assume that we can easily sample from another distribution with the density $g(x) = g^*(x)/Q$, where Q is also a constant

Rejection Sampling

- Now we choose the constants c such that $cg^*(x)$ becomes the envelope (blanket) function for $f^*(x)$:

$$cg^*(x) \geq f^*(x), \quad \forall x$$

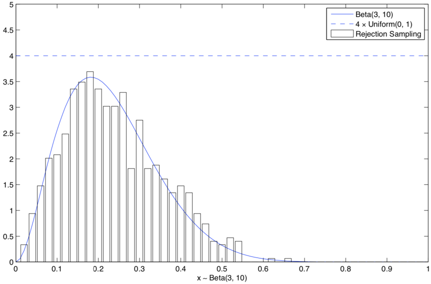
- Then, we can use a strategy known as *rejection sampling* in order to sample from $f(x)$ indirectly
- The rejection sampling method works as follows
  1. draw a sample x from $g(x)$
  2. generate $u \sim \text{Uniform}(0, 1)$
  3. if $u \leq \frac{f^*(x)}{cg^*(x)}$ we accept x as the new sample, otherwise, reject x (discard it)
  4. return to step 1

Rejection sampling generates samples from the target density, no approximation involved

$$
\begin{aligned}
p(X^R \leq y) &= p(X^g \leq y | U \leq \frac{f^*(X^g)}{cg^*(X^g)}) \\
&= p(X^g \leq y, U \leq \frac{f^*(X^g)}{cg^*(X^g)}) / p(U \leq \frac{f^*(X^g)}{cg^*(X^g)}) \\
&= \frac{\int_{-\infty}^{y} \int_0^{\frac{f^*(z)}{cg^*(z)}} du g(z) dz}{\int_{-\infty}^{\infty} \int_0^{\frac{f^*(z)}{cg^*(z)}} du g(z) dz} \\
&= \int_{-\infty}^{y} f(z) dz
\end{aligned}
$$

Example

- Assume that it is difficult to sample from the Beta(3, 10) distribution (this is not the case of course)
- We use the Uniform$(0, 1)$ distribution with $g(x) = 1,\ \forall x \in [0, 1]$, which has the envelop proporty: $4g(x) > f(x),\ \forall x \in [0, 1]$. The following graph shows the result after 3000 iterations

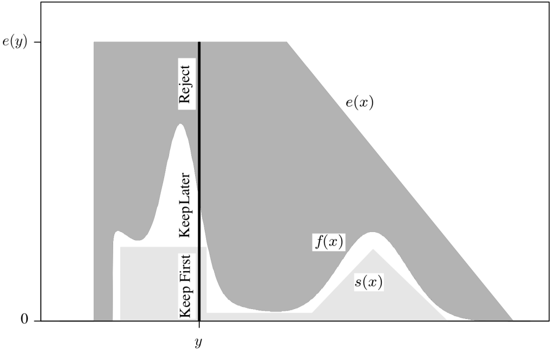


Rejection sampling becomes challenging as the dimension of x increases. A good rejection sampling algorithm must have three properties

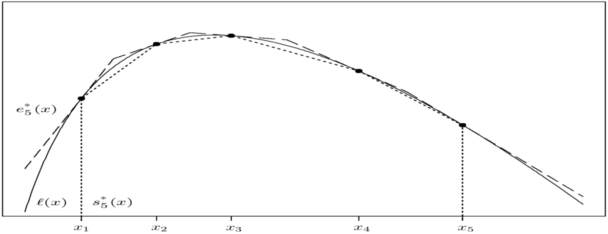
- It should be easy to construct envelops that exceed the target everywhere
- The envelop distributions should be easy to sample from
- It should have a low rejection rate

Squeezed Rejection Sampling

- When evaluating f^* is computationally expensive, we can improve the simulation speed of rejection sampling via *squeezed rejection sampling*
- Squeezed rejection sampling reduces the evaluation of f via a nonnegative squeezing function s that does not exceed f^* anywhere on the support of f: $s(x) \leq f^*(x), \forall x$
- The algorithm proceeds as follows:
  1. draw a sample x from $g(x)$
  2. generate $u \sim \text{Uniform}(0, 1)$
  3. if $u \leq \frac{s(x)}{cg^*(x)}$, we accept x as the new sample, return to step 1
  4. otherwise, determine whether $u \leq \frac{f^*(x)}{cg^*(x)}$. If this inequality holds, we accept x as the new sample, otherwise, we reject it.
  5. return to step 1

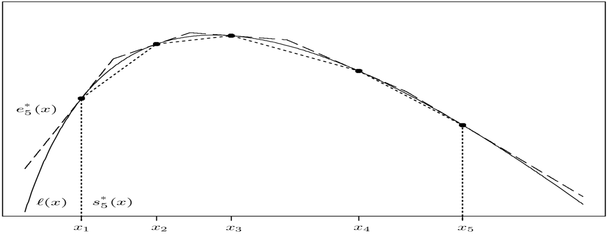


Remark: The proportion of iterations in which evaluation of f is avoided is $\int s(x)dx / \int e(x)dx$



- For a continuous, differentiable, log-concave density on a connected region of support, we can adapt the envelope construction (Gilks and Wild, 1992)
- Let $T = \{x_1, \ldots, x_k\}$ be the set of k starting points.
- We first sample x^* from the piecewise linear upper envelop $e(x)$, formed by the tangents to the log-likelihood ℓ at each point in T_k.

Adaptive Rejection Sampling



- To sample from the upper envelop, we need to transform from log space by exponentiating and using properties of the exponential distribution
- We then either accept or reject x^* as in squeeze rejection sampling, with $s(x)$ being the piecewise linear lower bound formed from the chords between adjacent points in T
- Add x^* to T whenever the squeezing test fails.

References

- P. J. Davis and P. Rabinowitz. Methods of Numerical Integration. Academic, New York, 1984.
- W. R. Gilks and P. Wild. Adaptive rejection sampling for Gibbs sampling. Applied Statistics, 41:337–348, 1992.