Department of Statistics and Data Science at SUSTech

MAT7035: Computational Statistics

Tutorial 8: Monte Carlo Integration

A. Numerical integration methods

A.1 Aim

- Suppose that we are interested in evaluating the integral

$$\mu = E\{h(X)\} = \int_{\mathbb{X}} h(x) \cdot f(x)\,\mathrm{d}x,$$

where $h(\cdot) \geqslant 0$ is a function and $f(\cdot)$ is the pdf of a r.v. X with support $\mathbb{X}$.

A.2 Classical Monte Carlo integration

- Let $\{X^{(i)}\}_{i=1}^m \stackrel{\text{iid}}{\sim} f(\cdot)$, then

$$\bar{\mu}_m = \frac{1}{m}\sum_{i=1}^m h\big(X^{(i)}\big)$$

is called the (classical) Monte Carlo integration of μ.

A.3 Riemannian simulation

- Let $\{X^{(i)}\}_{i=1}^{m+1} \stackrel{\text{iid}}{\sim} f(\cdot)$, $\{X_{(i)}\}_{i=1}^{m+1}$ are the order statistics of $\{X_{(i)}\}_{i=1}^{m+1}$, then

$$\hat{\mu}^{\mathrm{R}} = \sum_{i=1}^m h\big(X_{(i)}\big) f\big(X_{(i)}\big)\big[X_{(i+1)} - X_{(i)}\big]$$

is called the Riemannian sum estimator of μ.

A.4 The importance sampling method

— Let $H(x) \mathrel{\hat{=}} h(x)f(x)$ be defined on $\mathbb{X}$.

— If we could find an easy-sampling density function $g(\cdot)$ with support $\mathbb{X}$, we can write

$$\mu = \int_{\mathbb{X}} H(x)\,\mathrm{d}x = \int_{\mathbb{X}} \frac{H(x)}{g(x)} \cdot g(x)\,\mathrm{d}x = \int_{\mathbb{X}} \mathrm{w}(x) \cdot g(x)\,\mathrm{d}x,$$

where $\mathrm{w}(x) \hat{=} H(x)/g(x)$ is called ratio function.

— Let $X^{(1)}, \ldots, X^{(m)} \overset{\text{iid}}{\sim} g(x)$, then μ can be estimated by

$$\tilde{\mu}_m = \frac{1}{m} \sum_{i=1}^{m} \mathbf{w}(X^{(i)}),$$

which is called the *importance sampling* (IS) estimator.

Example T8.1 (Classical Monte Carlo integration method). Let $\gamma = -0.5$. Use the classical Monte Carlo integration method to compute

$$\mu = \frac{1}{\sqrt{\pi}} \int_{-\infty}^{\infty} \frac{1 + \mathrm{e}^x}{1 + \gamma}\, \mathrm{e}^{-0.5x^2}\,\mathrm{d}x.$$

Solution: Let

$$h(x) = \sqrt{2}\frac{1 + \mathrm{e}^x}{1 + \gamma},$$

then

$$\mu = \int_{-\infty}^{\infty} h(x) \cdot \phi(x)\,\mathrm{d}x \quad \text{and} \quad \bar{\mu}_m = \frac{1}{m} \sum_{i=1}^{m} h\big(X^{(i)}\big),$$

where $\{X^{(i)}\}_{i=1}^{m}$ are i.i.d. samples from the standard normal density $\phi(x)$.

R code:

```
> MC.in1 <- function(m)
{   # Function name: MC.in1
    # ------- Input ---------------
    # m = the number of samples
    # ------- Onput ---------------
    # mu = numerical integration by
    #      Classical MC method
    # -----------------------------
```

```
    g <- -0.5
    x <- rnorm(m,0,1)
    h <- sqrt(2)*(1+exp(x))/(1+g)
    mu<- mean(h)
    return(mu)
}
```

Results:

m	$\bar{\mu}_m$
10^3	7.306
10^5	7.511
10^7	7.493

Example T8.2 (Classical Monte Carlo integration and Riemannian simulation). Use the classical Monte Carlo integration and Riemannian simulation to compute

$$\mu = \int_0^1 \frac{4}{1+x^2}\,\mathrm{d}x.$$

Solution: Let $h(x) = 4(1+x^2)^{-1}$ and $f(x) = 1$ for $x \in (0,1)$, then the integral is rewritten as

$$\mu = \int_0^1 h(x)f(x)\,\mathrm{d}x.$$

We have

$$\bar{\mu}_m = \frac{1}{m}\sum_{i=1}^{m} h\big(X^{(i)}\big) \quad \text{and} \quad \mu_m^{\mathrm{R}} = \sum_{i=1}^{m-1} h\big(X_{(i)}\big) f\big(X_{(i)}\big)\big[X_{(i+1)} - X_{(i)}\big],$$

where $\{X^{(i)}\}_{i=1}^m \overset{\text{iid}}{\sim} U(0,1)$ and $\{X_{(i)}\}_{i=1}^m$ are the order statistics of $\{X^{(i)}\}_{i=1}^m$.

R code:

```
> MC.in2 <- function(m)
{   # Function name: MC.in2
    # ------- Input ---------------
    # m     = the number of samples
```

```
    # ------- Onput ---------------
    # m    = (m[1],m[2])
    # m[1] = Classical MC Method
    # m[2] = Riemannian Simulation
    # ----------------------------
    mu <- rep(0, 2)
    x  <- runif(m,0,1)
    x  <- sort(x)
    h  <- 4/(1+x^2)
    mu[1] <- 1/m*sum(h)
    mu[2] <- sum(h[2:m]*diff(x))
    return(mu)
}
```

Results:

m	$\bar{\mu}_m$	μ_m^{R}
10^3	3.15126	3.12872
10^5	3.13665	3.14124
10^7	3.14049	3.14152

Remark: For an integral in a finite interval, the uniform distribution on the interval, i.e., a constant pdf $f(\cdot)$ is a good choice.

Example T8.3 (Classical Monte Carlo integration and the importance sampling method). Compute the value of normal cdf

$$\mu = \Phi(t) = \int_{-\infty}^{t} \frac{1}{\sqrt{2\pi}} \mathrm{e}^{-\frac{x^2}{2}} \, \mathrm{d}x,$$

for $t = 0$ by the classical Monte Carlo integration and the importance sampling method with the standard Cauchy density $g(x) = [\pi(1+x^2)]^{-1}$ as the proposal density.

Solution: (a) Note that

$$\mu = \int_{-\infty}^{\infty} I(x \leqslant t) \cdot \phi(x) \, \mathrm{dx},$$

where $I(\cdot)$ is the indicator function, and $\phi(x)$ is the pdf of the standard normal distribution. Then the classical Monte Carlo estimator is

$$\bar{\mu}_m = \frac{1}{m}\sum_{i=1}^{m} I(X^{(i)} \leqslant t),$$

where $\{X^{(i)}\}_{i=1}^m \overset{\text{iid}}{\sim} N(0,1)$.

(b) Use the importance sampling method, we have

$$\begin{aligned}
\mu &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}}\,\mathrm{e}^{-\frac{x^2}{2}} I(x \leqslant t)\,\mathrm{d}x \\
&= \int_{-\infty}^{\infty} \frac{\frac{1}{\sqrt{2\pi}}\,\mathrm{e}^{-\frac{x^2}{2}} I(x \leqslant t)}{g(x)} \cdot g(x)\,\mathrm{d}x \\
&= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}}\,\mathrm{e}^{-\frac{x^2}{2}} I(x \leqslant t)\pi(1+x^2)\cdot g(x)\,\mathrm{d}x \\
&= \sqrt{\frac{\pi}{2}}\int_{-\infty}^{\infty} \mathrm{e}^{-\frac{x^2}{2}}(1+x^2)I(x \leqslant t)\cdot g(x)\,\mathrm{d}x,
\end{aligned}$$

so that

$$\tilde{\mu}_m = \frac{1}{m}\sqrt{\frac{\pi}{2}}\sum_{i=1}^{m} \mathrm{e}^{-\frac{(X^{(i)})^2}{2}}\big[1+(X^{(i)})^2\big] I\big(X^{(i)} \leqslant t\big),$$

where $\{X^{(i)}\}_{i=1}^m$ are i.i.d. samples from the standard Cauchy density. We choose $t=0$, which simply makes $\Phi(t) = 0.5$, to test the algorithms.

R code:

```
> MC.in3 <- function (m)
{   # Function name: MC.in3
    # ------- Input ---------------
    # m  = the number of samples
    # ------- Onput ---------------
    # mu = numerical integration by
    #      Classical MC Method
    # -----------------------------
    t <- 0
```

```
    x <- rnorm(m,0,1)
    h <- (x<=t)
    mu <- mean(h)
    return(mu)
}
```

```
> MC.in4 <- function (m)
{   # Function name: MC.in4
    # ------- Input ------------------
    # m  = the number of samples
    # ------- Onput ------------------
    # mu = numerical integration by
    #      Importance Sampling Method
    # --------------------------------
    t <- 0
    x <- rcauchy(m,0,1)
    w <- exp(-x^2/2)*(1+x^2)*(x<=t)
    mu <- sqrt(pi/2)*mean(w)
    return(mu)
}
```

Results:

m	$\bar{\mu}_m$	$\tilde{\mu}_m$
10^3	0.5170	0.4828
10^5	0.5043	0.4988
10^7	0.5001	0.4996