Lab 10

November 17, 2016

This is an R Markdown Notebook.

Monte Carlo Estimate, Error and C.I.

We will briefly review basics of *Monte Carlo integration*, and the quantities *Monte Carlo Estimate*, *Monte Carlo Error* and *Confidence Interval*.

The idea of *Monte Carlo integration* is to turn the problem of integration into evaluating an expectation by sample mean.

Question

We want to integrate $f : \mathbb{R}^d \to \mathbb{R}$ over $\mathbb{R}^d$. We decompose f into a product of h and π. Now we can evaluate the integral as an expectation $\int f(x)\mathrm{d}x = \int h(x)\pi(x)\mathrm{d}x = \mathrm{E}_\pi h(X)$.

Monte Carlo Estimate

$\pi_n(h) \triangleq \frac{1}{n}\sum\limits_{k=1}^{n} h(X_k)$ is called *Monte Carlo estimate* of $\int h(x)\pi(x)dx$, where $X_1, X_2, \ldots, X_n$ are iid random sample drawn from the density $\pi(x)$. $\pi_n(h)$ is estimated by the sample average of $h(X_1), \ldots, h(X_n)$.

Monte Carlo Error

$\frac{s_n(h)}{\sqrt{n}} = \sqrt{s_n(h)^2/n}$ is called the *Monte Carlo error* of the estimate $\pi_n(h)$, where $s_n(h)^2/n = \frac{1}{n-1}\sum\limits_{k=1}^{n}\Big(h(X_k) - \pi_n(h)\Big)^2\Big/n$, which in R is simply `var(h(X))/n`.

Confidence Interval

$\pi_n(h) \mp z_{\alpha/2}\frac{s_n(h)}{\sqrt{n}}$ gives the $(1-\alpha)$ level approximate confidence interval for the mean $\mathrm{E}h(X)$ that we wanted to estimate.

Example

Consider $f(x) = x^3\exp\{-0.1x - \frac{1}{2}[(x-3)^2 + (x-5)^2]\}$. Compute the integral

$$\int_0^\infty f(x)dx.$$

Idea

consider $\int_{-\infty}^{\infty} f(x)dx = \int_{-\infty}^{\infty}\frac{f(x)}{\pi(x)}\pi(x)dx = \mathrm{E}\frac{f(X)}{\pi(X)}$, where $X \sim \pi(x)$ the standard normal density.

Here f/π plays the role of h above.

Algorithm

1. draw n samples $\{x_1, x_2, \ldots, x_n\}$ from standard normal distribution $\pi(x)$ by `rnorm(n)`;
2. compute sample `mean()`, π_n, of the sample $\{\frac{f(x_1)}{\pi(x_1)}, \frac{f(x_n)}{\pi(x_n)}, \ldots, \frac{f(x_n)}{\pi(x_n)}\}$, which is the Monte Carlo estimate of the integral.

3. compute the Monte Carlo error $\frac{s_n}{\sqrt{n}} = \sqrt{\frac{1}{n-1}\sum_{k=1}^n \left(f(x_k)/\pi(x_k) - \pi_n\right)^2 \Big/ n}$, where one can evaluate $\frac{1}{n-1}\sum_{k=1}^n \left(f(x_k)/\pi(x_k) - \pi_n\right)^2$ by the sample variance `var()` of the sample $\{\frac{f(x_1)}{\pi(x_1)}, \frac{f(x_n)}{\pi(x_n)}, \dots, \frac{f(x_n)}{\pi(x_n)}\}$.
4. compute the 95% confidence interval $\pi_n \mp q_{0.975}\frac{s_n}{\sqrt{n}}$, where $q_{0.975} = z_{0.025}$ denotes the 0.975 quantile of standard normal.

Implementation:

```
f <- function(x){
  (x^3*exp(-0.1*x-1/2*((x-1)^2+(x-5)^2)))*(x>0)
}
MCI <- function(n)
{
  x <- rnorm(n,3,1)
  integral <- mean(f(x)/dnorm(x,3,1))
  mc_error <- sqrt(var(f(x)/dnorm(x,3,1))/n)
  z = qnorm(0.975)
  CIl = integral - z * mc_error;
  CIr = integral + z * mc_error;
  output <- c(MCestimate = integral, MCerror = mc_error,
              Lower.limit = CIl, Upper.limit = CIr)
  return(output)
}

print(MCI(100000))
```

```
##  MCestimate     MCerror Lower.limit Upper.limit
## 0.723554215 0.001345233 0.720917606 0.726190824
```

Importance Sampling

Importance Sampling is another Monte Carlo integration technique. Built on the Basic Monte Carlo which transforms an integral to an expectation estimated by appropriate sample mean, Importance Sampling turns one expectation representation to another

Question

Let $h : \mathbb{R}^d \to \mathbb{R}$ be a function, and $\pi(x)$ be a density on $\mathbb{R}^d$. We want to evaluate the mean $\mathrm{E}h(X) = \int h(x)\pi(x)dx$.

Basic Monte Carlo

Draw a random sample $\{x_1, x_2, \dots, x_n\}$ from $\pi(x)$ and esimate $\int h(x)\pi(x)dx$ by the sample mean $\pi_n(h) = \frac{1}{n}\sum\limits_{k=1}^{n} h(x_k)$.

Problem with Basic Monte Carlo

It can be difficult or inefficient, or even impossible to draw a random sample from $\pi(x)$ that falls into the integration region, we will further explain this by two examples later, for now, how to solve this problem? We turn to importance sampling:

Importance Sampling with fully specified target density $\pi(x)$

consider

$$\begin{aligned}\pi(h) &= \mathrm{E}_\pi[h(X)]\\ &= \int h(x)\pi(x)\mathrm{d}x\\ &= \int h(x)\frac{\pi(x)}{g(x)}g(x)\mathrm{d}x\\ &= \int h(x)\omega(x)g(x)\mathrm{d}x\\ &= \mathrm{E}_g[h(X)\omega(X)] = g(h\omega)\end{aligned}$$

where $\omega(x) = \frac{\pi(x)}{g(x)}$. now draw a random sample $X_1, X_2, \ldots, X_n$ from $g(x)$ and esimate $\int h(x)\pi(x)dx$ by the sample mean $\pi_{n,IS}(h) = \frac{1}{n}\sum\limits_{k=1}^{n} h(x_k)\omega(x_k)$.

Importance Sampling Error and C.I.

Similiar to Basic Monte Carlo, the Importance sampling error is given by $\frac{s_{n,IS}(h)}{\sqrt{n}} = \sqrt{s^2_{n,IS}(h)/n}$, where $s^2_{n,IS}(h) = \frac{1}{n-1}\sum\limits_{k=1}^{n}(h(x_k)\omega(x_k) - \pi_{n,IS}(h))^2$ and the $1-\alpha$ level confidence interval is given by $\pi_{n,IS}(h) \pm z_{\alpha/2}\frac{s_{n,IS}(h)}{\sqrt{n}}$.

Importance Sampling with unknown constant in $\pi(x) = \tilde{\pi}(x)/C$:

Imagine the situation: You know $X \sim \pi$, but you only know π up to a normalizing constant, i.e., $\pi(x) = \tilde{\pi}(x)/C$, where $\tilde{\pi}$ is known but C is not.

Example from last homework:

$$\pi(x) = \frac{x^{\alpha-1}\exp\{-\beta x - \frac{1}{2}\sum_{i=1}^{10}(x-m_i)^2\}}{\int_0^\infty x^{\alpha-1}\exp\{-\beta x - \frac{1}{2}\sum_{i=1}^{10}(x-m_i)^2\}\mathrm{d}x} \quad \text{on} \quad [0,\infty)$$

and vanishes on $(\infty, 0)$.

Evaluating $\mathrm{E}h(X)$ where $X \sim \pi$ is difficult.

Our strategy is to use Monte Carlo integration to approximate the normalizing constant.

Since $\pi(x) = \tilde{\pi}(x)/C$, and $\int \pi(x)dx = 1$ for any density function $\pi(x)$, we have

$$C = \int \tilde{\pi}(x)\mathrm{d}x = \int \frac{\tilde{\pi}(x)}{g(x)}g(x)\mathrm{d}x = \int \tilde{\omega}(x)g(x)\mathrm{d}x = \mathrm{E}_g[\tilde{\omega}(X)]$$

where $\tilde{\omega}(x) = \frac{\tilde{\pi}(x)}{g(x)}$.

Now the original problem can be converted into

$$\begin{aligned}\int h(x)\pi(x)\mathrm{d}x &= \int h(x)\frac{\tilde{\pi}(x)}{C}\frac{g(x)}{g(x)}\mathrm{d}x\\ &= \frac{1}{C}\int h(x)\frac{\tilde{\pi}(x)}{g(x)}g(x)\mathrm{d}x\\ &= \frac{\int h(x)\tilde{\omega}(x)g(x)\mathrm{d}x}{C}\\ &= \frac{\int h(x)\tilde{\omega}(x)g(x)\mathrm{d}x}{\int \tilde{\omega}(x)g(x)\mathrm{d}x}\\ &= \frac{\mathrm{E}_g[h(X)\tilde{\omega}(X)]}{\mathrm{E}_g[\tilde{\omega}(X)]}\end{aligned}$$

where X follows the distribution g. Now draw a random sample $X_1, X_2, \ldots, X_n$ from g and esimate $\int h(x)\pi(x)dx$ by the ratio of two sample means

$$\tilde{\pi}_{n,IS}(h) = \frac{\frac{1}{n}\sum\limits_{k=1}^{n} h(X_k)\tilde{\omega}(X_k)}{\frac{1}{n}\sum\limits_{k=1}^{n} \tilde{\omega}(X_k)} = \frac{\sum\limits_{k=1}^{n} h(X_k)\tilde{\omega}(X_k)}{\sum\limits_{k=1}^{n} \tilde{\omega}(X_k)}$$

The denominator is also a random variable (and converges to C by law of large numbers). Therefore the whole thing is unstable if the estimate $\sum\limits_{k=1}^{n} \tilde{\omega}(X_k)$ has large variance. So we have a

A rule of thumb:

$$CV = \sqrt{\frac{1}{n-1}\sum_{k=1}^{n}\left(\frac{\tilde{w}(x_k)}{\bar{w}} - 1\right)^2} \quad \text{where} \quad \bar{\omega} = \frac{1}{n}\sum_{k=1}^{n}\tilde{\omega}(x_k)$$

should be small, say, < 5.

CV in R is simply $\text{sqrt}(\text{var}(\tilde{\omega}(\vec{x})/\bar{\omega}))$, where $\bar{\omega} = \text{mean}(\tilde{\omega}(\vec{x}))$.

The Monte Carlo error of Importance Sampling in this scheme is $\sqrt{s^2_{n,IS}/n}$, where $s^2_{n,IS}(h) = \frac{1}{n-1}\sum\limits_{k=1}^{n}(Z_{n,k} - \tilde{\pi}_{n,IS}(h))^2$ and $Z_{n,k} = \frac{h(X_k)\tilde{\omega}(X_k)}{\frac{1}{n}\sum_{j=1}^{n} h(X_k)\tilde{\omega}(X_k)}$.

Again the $1-\alpha$ level confidence interval is given by $\tilde{\pi}_{n,IS}(h) \pm z_{\alpha/2}\frac{s_{n,IS}(h)}{\sqrt{n}}$.

Example 1: IS where $\pi(x)$ has known normalizing constant

We will look at a case where importance sampling provides a reduction in the variance of an integral approximation. Suppose that we want to calculate $\text{E}(h(X))$, where $X \sim \text{Uniform}(0, 10)$ and $h(x) = 10\exp(-2|x-5|)$. That is, we want to calculate the integral

$$\int_0^{10} 10\exp(-2|x-5|)\cdot\frac{1}{10}\text{d}x = \int_0^{10}\exp(-2|x-5|)\cdot\text{d}x$$

The true value of this integral is about 1 (exact value $1 - e^{-10} \approx 0.99995$). The simple way to do this is to use the basic monte carlo approach and generate X_i from the Uniform(0,10) density and look at the sample mean of $h(X_i)$ (notice this is equivalent to importance sampling with importance function $w(x) = 1$):

```r
X <- runif(1e5,0,10)
Y <- 10*exp(-2*abs(X-5))
c(mean = mean(Y), variance = var(Y))
```

```
##      mean  variance
## 0.9985579 4.0012130
```

The function h in this case is peaking at 5, and decays quickly on both sides, therefore, under the uniform distribution, many of the points contributes little to this weighted average.

A distribution like a gaussian function (ce^{-x^2}) with mean 5 and a small variance, say, 1, is closer to the shape of the integrand $|h(x)|\pi(x)$, and should enable a better estimate.

We can re-write the integral as

$$\int_0^{10} 10\exp(-2|x-5|)\frac{1/10}{\frac{1}{\sqrt{2\pi}}e^{-(x-5)^2/2}}\frac{1}{\sqrt{2\pi}}e^{-(x-5)^2/2}dx = \int_0^{10} 10\exp(-2|x-5|)\frac{1}{10}\sqrt{2\pi}e^{(x-5)^2/2}\frac{1}{\sqrt{2\pi}}e^{-(x-5)^2/2}dx$$

That is, $\mathrm{E}[h(X)w(X)]$, where $X \sim N(5,1)$, and $w(x) = \frac{\sqrt{2\pi}}{10}e^{(x-5)^2/2}$ is the importance weight.

```
X=rnorm(1e5,mean=5,sd=1)
hX = 10*exp(-2*abs(X-5))
Y=hX*dunif(X,0,10)/dnorm(X,mean=5,sd=1)
c(mean = mean(Y), variance = var(Y))
```

```
##      mean  variance
## 1.0001907 0.3576264
```

Notice that the integral calculation is still accurate, but with a variance this is approximately 1/10 of the naive monte carlo approximation. This is a case where importance sampling provided a substantial improvement in precision.

Example 2: IS where normalizing constant of π is unknown

Compute the mean of the distribution with density $\pi(x) \propto \tilde{\pi}(x) = \frac{e^{-x}}{1+x}$ for $x > 0$. Use exponential density for different choices of rate parameter λ as your trial density g. Find the one which minimizes the CV (rule of thumb).

Solution: we want to compute

$$\int_0^{\infty} x\pi(x)dx \quad (1)$$

but we only know the density $\pi(x)$ up to a normalizing constant since $\pi(x) \propto \tilde{\pi}(x) = \frac{e^{-x}}{1+x}$, i.e., $\pi(x) = \frac{\tilde{\pi}(x)}{C}$ where the constant C is unknown.

Since $\pi(x)$ is a density, $C = \int_0^{\infty} \tilde{\pi}(x)dx$.

Using the idea of "weighted average Importance Sampling", choosing $g(x) = \lambda e^{-\lambda x}$, we esimate (1) by the following algorithm:

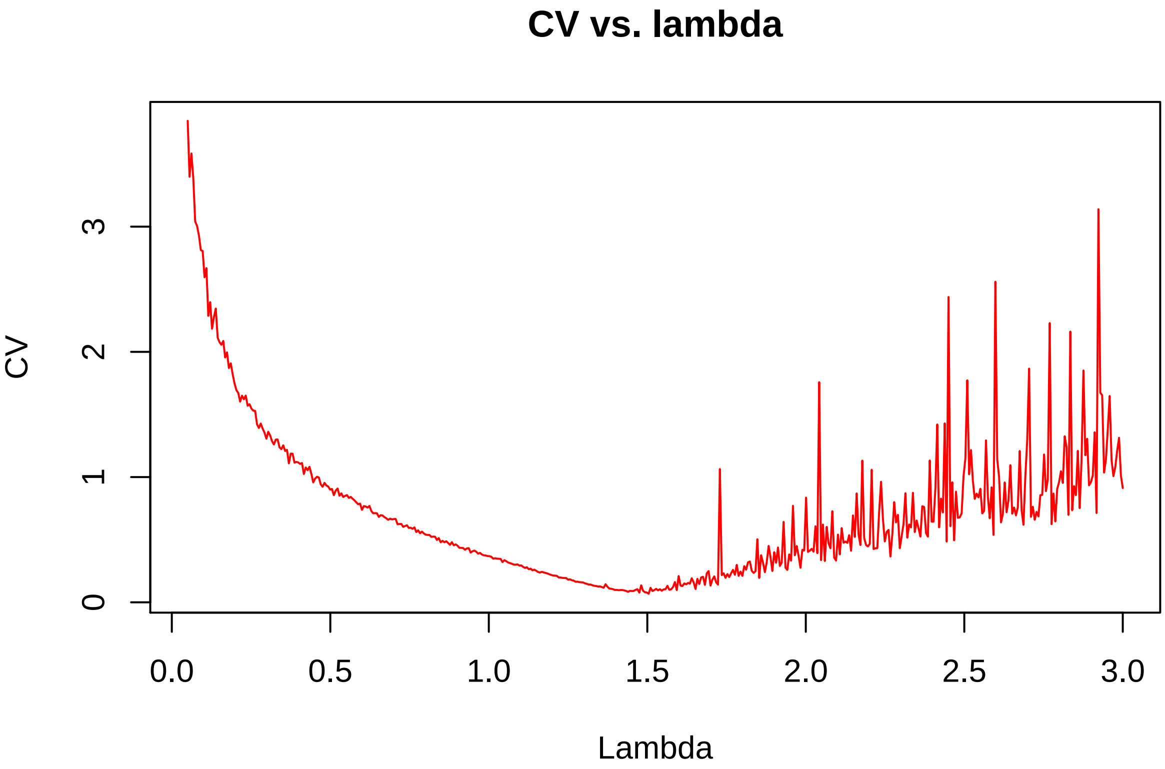
Algorithm:

1. draw a sample $\{x_k\}_{k=1:n}$ of size n from $g(x) = \lambda e^{-\lambda x}$.
2. compute the sample mean $(S1)$ of $\{h(x_k)\tilde{\omega}(x_k)\}_{k=1:n}$, where $h(x) = x$, $\tilde{\omega}(x) = \frac{\tilde{\pi}(x)}{g(x)} = \frac{e^{(\lambda-1)x}}{\lambda(1+x)}$;
3. compute the sample mean $(S2)$ of $\{\tilde{\omega}(x_k)\}_{k=1:n}$,
4. Take the ratio of $\frac{S1}{S2}$, this is the importance sampling estimate of (1),
5. To choose the best λ that minimizes CV in $g(x)$, use a grid of λ values for $g(x)$ and compute the corresponding CV.
6. compute the estimation error as square root of $1/n$ times sample variance of $\left\{\frac{h(x_k)\tilde{\omega}(x_k)}{\bar{\omega}}\right\}_{k=1:n}$.

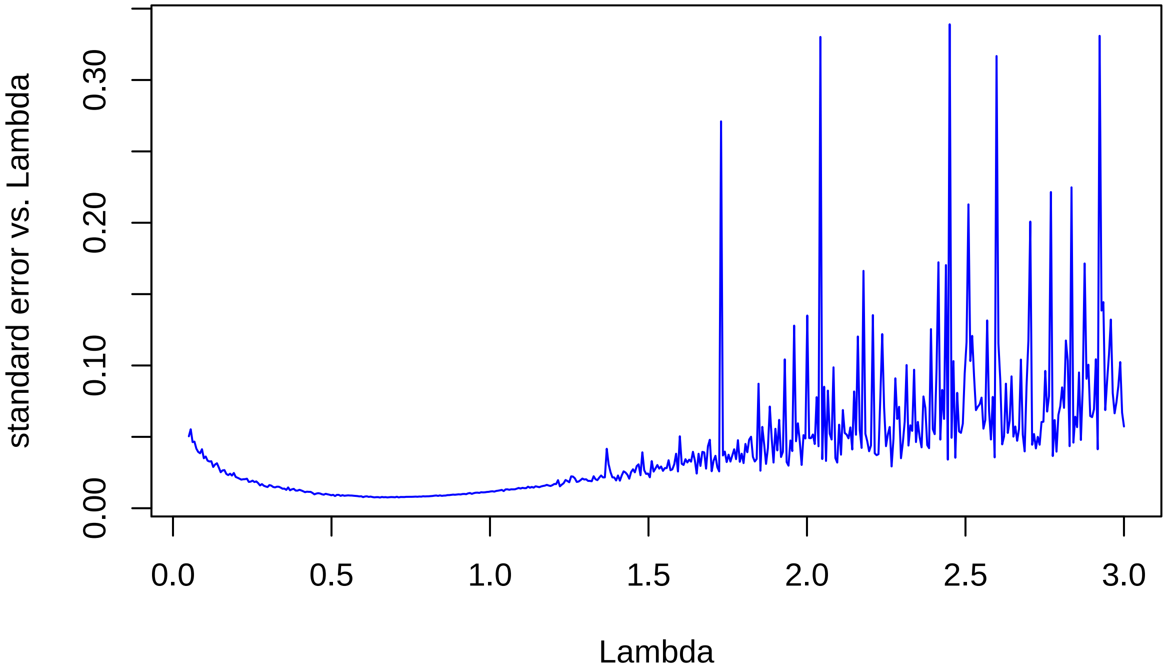
Implementation:

```
# n is the number of monte carlo samples,
# lambda is the rate parameter for the g (exponential) distribution

# Target density with unknow constant C
pi = function(t) (exp(-t) / (t+1))
```

```r
IS = function(input){

    lambda = input[1]
    n = input[2]

    # trial density, g
  g = function(t) dexp(t, lambda)

  # importance weight function, tilde w in algorithm step 2
  w = function(t) pi(t) / g(t)

    # draw sample from g
    X = rexp(n, lambda)

    # calculate the list of importance values
    LW = w(X)

    # importance sampling estimate
    I = mean( X * LW ) / mean(LW)

    # calculate sample coefficient of variation CV
    CV = sqrt( var( LW / mean(LW) ) )

    # calculate importance sampling error
    sig.sq = var( LW*X / mean(LW) )
    se = sqrt( sig.sq / n )

    output = c(estimate = I, se = se, CV = CV)
    return(output)
}

## calculate CV for a grid of values of lambda
lambda.val <- seq(.05, 3, length=500)
n.val <- rep(1000, 500)
# inpt.mat is a 500 times 2 matrix containing all the inputs, each row
# of inpt.mat is an input
inpt.mat <- matrix(c(lambda.val, n.val), nrow = 500, ncol = 2)

## apply the weighted average Importance Sampling function

# IS() to every row in inpt.mat
# each output of IS() function as c(I, se, CV) is put as a column of A
A <- apply(inpt.mat,1, IS)
# transpose A to get each row of A as one output.
A <- t(A)

# see where CV is low enough
plot(lambda.val, A[,3], ylab="CV", xlab="Lambda", main="CV vs. lambda", col=2,
type="l")
abline(h=5) # only those with a CV value below 5 are considered stable
```



```
# importance sampling error estimates (standard errors of IS estimates)
plot(lambda.val, A[,2], xlab="Lambda", ylab="standard error vs. Lambda", col=4,
type="l")
```



```
# final answer : the one exp(lambda) denstiy resulting in the
# smallest CV and
#its corresponding IS() outputs c(I, se, CV). You can see
#its estimation error se is also small among other choices of lambda.
```

```r
indx = which.min(A[,3])
(fin.ans <- c(lambda = lambda.val[indx], A[indx,]))
```

```
##     lambda   estimate         se         CV
## 1.50430862 0.64122494 0.02167610 0.06843904
```