Monte Carlo Integration

Max Turgeon

STAT 3150–Statistical Computing

Lecture Objectives

- Understand what Monte Carlo integration is and why it works.
- Be able to use random sampling to estimate statistical quantities of interest.
- Learn about strategies for reducing the variance of the estimates.

Motivation

- Many statistical quantities of interest can be defined as integrals.
    - E.g. the expectation.
- But symbolic integration is difficult, and some functions don't have anti-derivatives!
- We will see how integrals can be estimated by taking the average of a suitable collection of **random variates**.
    - In Module 9, we'll talk about *numerical integration*, which can also be used instead of symbolic integration.

Simple motivating example i

- Imagine we want to estimate the following definite integral:

$$\int_0^1 e^{-x} dx.$$

- From Calculus, we know that $G(x) = -e^{-x}$ is an anti-derivative for $g(x) = e^{-x}$, and so we can quickly check that the integral is equal to $1 - e^{-1} \approx 0.6321$.
- Let's generate uniform variates on $(0, 1)$ and take the average of their image by g:

Simple motivating example ii

```r
n <- 1000
unif_vars <- runif(n)
mean(exp(-unif_vars))
```

```
## [1] 0.6391107
```

```r
# Compare to actual value
1 - exp(-1)
```

```
## [1] 0.6321206
```

Simple motivating example iii

- **What's going on?** If we write $X_1, \ldots, X_n$ for the uniform variates, the Law of Large Numbers tells us that

$$\frac{1}{n}\sum_{i=1}^{n} g(X_i) \to E(g(X)), \quad \text{where } X \sim U(0,1).$$

- But since the density of a uniform random variable on $(0,1)$ is just the constant function 1, we have

$$E(g(X)) = \int_0^1 g(x)dx.$$

- Let's see what happens if we try on the interval $(0,2)$:

Simple motivating example iv

```r
unif_vars <- runif(n, max = 2)
mean(exp(-unif_vars))
```

```
## [1] 0.4163901
```

```r
# Compare to actual value
1 - exp(-2)
```

```
## [1] 0.8646647
```

- Something isn't right... We get about half of what we expect...

Simple motivating example v

- That's because the density of a uniform variable on $(0, 2)$ is no longer the constant function 1, but rather the constant function $1/2$:

$$\frac{1}{n}\sum_{i=1}^n g(X_i) \to \frac{1}{2}\int_0^2 g(x)dx.$$

- Therefore, we need to multiply the sample mean by 2:

```
2*mean(exp(-unif_vars))
```

```
## [1] 0.8327802
```

Simple Monte Carlo integration

Let $g(x)$ be an integrable function defined on the bounded interval (a, b). To estimate the integral

$$\int_a^b g(x)dx,$$

follow this algorithm:

1. Generate $X_1, \ldots, X_n$ independently from a uniform distribution on (a, b).
2. Compute the sample mean $\overline{g(X)} = \frac{1}{n}\sum_{i=1}^n g(X_i)$.
3. Estimate the integral via $(b-a)\overline{g(X)}$.

Slightly more complex example i

- Once we know that the LLN is working under the hood, we can expand our application beyond the uniform distribution.
- Let X be a continuous variable with density f. Then we know that
$$E(g(X)) = \int_{-\infty}^{\infty} g(x) f(x) dx.$$
- Therefore, if we generate $X_1, \dots, X_n$ independently from f, we can estimate $E(g(X))$ using
$$\overline{g(X)} = \frac{1}{n} g(X_i).$$

Slightly more complex example ii

- We will apply these ideas to the following integral:

$$\int_0^\infty \frac{e^{-x}}{1+x} dx.$$

- This integral is the product of a function $g(x) = \frac{1}{1+x}$ and the density of an exponential $Exp(1)$. In other words:

$$\int_0^\infty \frac{e^{-x}}{1+x} dx = E\left(\frac{1}{1+X}\right),$$

  where $X \sim Exp(1)$.

Slightly more complex example iii

```r
n <- 1000
exp_vars <- rexp(n)
mean(1/(1 + exp_vars))
```

```
## [1] 0.5838972
```

Variance and standard error i

- As we saw earlier, MC integration with $n = 1000$ samples gave an estimate “close” to the true value.
    - Can we measure how close?
- Let $\hat{\theta} = \frac{1}{n} f(X_i)$ be our sample mean.
    - By the LLN, it converges to $\theta = E(f(X))$.
- **Exercise**: If σ^2 is the variance of $f(X)$, check that the variance of $\hat{\theta}$ is equal to σ^2/n.
- For a general function $f(x)$, we don’t know the variance σ^2, so we need to estimate it:

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^{n} \left(f(X_i) - \hat{\theta} \right)^2 .$$

Variance and standard error ii

- Now, we can use the Central Limit Theorem:

$$\frac{\hat{\theta} - \theta}{\sqrt{\hat{\sigma}^2/n}} \to N(0, 1).$$

- We can construct an approximate 95% confidence interval around $\hat{\theta}$ as follows:

$$\hat{\theta} \pm 1.96\sqrt{\hat{\sigma}^2/n}.$$

Examples i

```r
# The first uniform example
n <- 1000
unif_vars <- runif(n)
theta_hat <- mean(exp(-unif_vars))
sigma_hat <- sd(exp(-unif_vars))

c("Lower" = theta_hat - 1.96*sigma_hat/sqrt(n),
  "Upper" = theta_hat + 1.96*sigma_hat/sqrt(n))
```

```
##     Lower     Upper
## 0.6231372 0.6448018
```

Examples ii

```
# Exponential example
exp_vars <- rexp(n)
theta_hat <- mean(1/(1 + exp_vars))
sigma_hat <- sd(1/(1 + exp_vars))

c("Lower" = theta_hat - 1.96*sigma_hat/sqrt(n),
  "Upper" = theta_hat + 1.96*sigma_hat/sqrt(n))
```

```
##     Lower     Upper
## 0.5817675 0.6093070
```

Convergence i

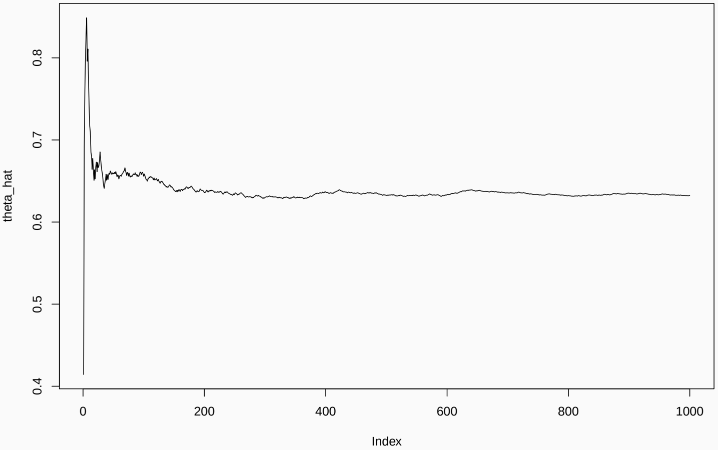
- How can we assess convergence of our Monte Carlo estimate?
    - Look at **trace plots**
- A trace plot displays the estimate as a function of the sample size.
    - Instead of recomputing for different sample sizes, use `dplyr::cummean` function to compute the cumulative mean.

Convergence ii

```r
library(dplyr)
# Recall our first example
n <- 1000
unif_vars <- runif(n)
theta_hat <- cummean(exp(-unif_vars))

plot(theta_hat,
     type = "l")
```

Convergence iii



Convergence iv

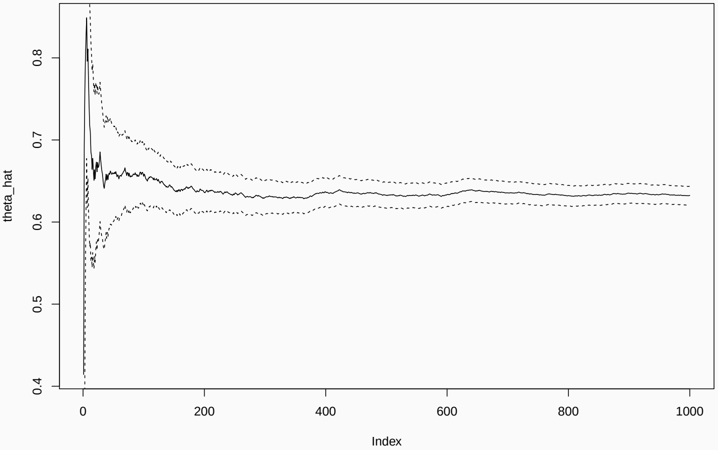
- We have evidence of convergence, because the line has stopped “bouncing around”, i.e. the movement happens in a very narrow range.
- Using our computations above, we can also put a confidence band around the trace plot.

Convergence v

```r
sigma2_hat <- cumstats::cumvar(exp(-unif_vars))
sigma_hat <- sqrt(sigma2_hat)

plot(theta_hat, type = "l")
lines(theta_hat + 1.96*sigma_hat/sqrt(seq(1, n)),
      lty = 2)
lines(theta_hat - 1.96*sigma_hat/sqrt(seq(1, n)),
      lty = 2)
```

Convergence vi



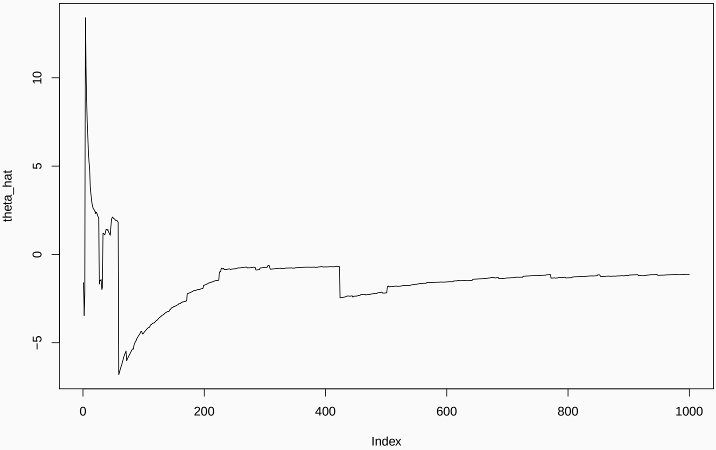
Convergence vii

- **Can we find an example that doesn't converge?**
- Recall: the LLN requires that the expectation of the random variables be *finite*.
    - So we can cook up an example using the Cauchy distribution.

```
n <- 1000
cauchy_vars <- rcauchy(n)
theta_hat <- cummean(cauchy_vars)

plot(theta_hat,
     type = "l")
```

Convergence viii



Example i

- Let's say we want to estimate the following integral:

$$\int_0^1 \frac{1}{x} dx.$$

- Can you spot the problem?

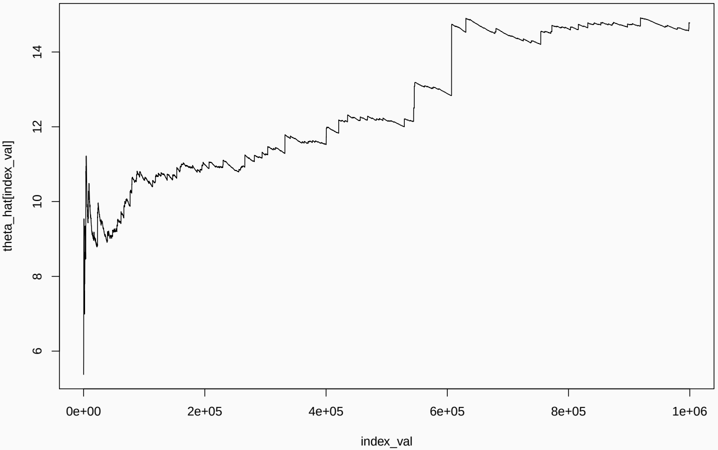
```
n <- 1000000
unif_vars <- runif(n)
theta_hat <- mean(1/unif_vars)
sigma_hat <- sd(1/unif_vars)
c(theta_hat, sigma_hat/sqrt(n))
```

Example ii

```
## [1] 14.776024  1.357136
```

```
# Let's look at a trace plot
theta_hat <- cummean(1/unif_vars)
# We'll only look at every 100th value
index_val <- seq(100, n, by = 100)
plot(x = index_val,
     y = theta_hat[index_val],
     type = "l")
```

Example iii



theta_hat[index_val]
6
8
10
12
14
0e+00
2e+05
4e+05
6e+05
8e+05
1e+06
index_val

Example iv

- **Conclusion**: Be careful! Monte Carlo integration will always give you a number. It's your job as a statistician to decide if you can trust it.
    - In this example, we knew from calculus that the integral is infinite.
    - In general cases, either prove analytically the integral exists, or at least look at a trace plot.

Variance reduction i

- We argued above using the CLT that the standard error of our estimate is

$$\frac{\sigma}{\sqrt{n}}.$$

- The parameter σ is a constant–it's determined by the integral we are trying to estimate.
    - $\sigma^2 = \mathrm{Var}(f(X))$
- Therefore, the only parameter we can control is n.
    - By increasing n, we can *decrease* the standard error.
- But because of the square root in the denominator, improvements are smaller as n increases.

Variance reduction ii

- For example, if for n_1 samples, the standard error is approximately 0.01, you need to increase the sample size by a factor of $100^2 = 10000$ to *decrease* the standard error to 0.0001.
- In other words, we would need $n_2 = 10000 n_1$ random samples!

Example i

```r
# Going back to second example
# Recall: Need to multiply by 2!
n <- 1000
unif_vars <- runif(n, max = 2)
theta_hat <- 2*mean(exp(-unif_vars))
sigma_hat <- 2*sd(exp(-unif_vars))
sigma_hat/sqrt(n)
```

```
## [1] 0.01524081
```

Example ii

```r
# What if we want a standard error of 0.0001?
factor <- (sigma_hat/sqrt(n)/0.0001)^2
(n2 <- factor * n)
```

```
## [1] 23228244
```

```r
unif_vars2 <- runif(n2, max = 2)
2*sd(exp(-unif_vars2))/sqrt(n2)
```

```
## [1] 0.0001003638
```

- **Antithetic variables** is a general strategy for reducing the variance *without changing the sample size*.
- The motivation is as follows: if we have random variables X, Y, the variance of their average is

$$\begin{aligned}\mathrm{Var}\left(\frac{X+Y}{2}\right) &= \frac{1}{4}\mathrm{Var}\left(X+Y\right)\\ &= \frac{1}{4}\left(\mathrm{Var}(X)+\mathrm{Var}(Y)+2\mathrm{Cov}(X,Y)\right).\end{aligned}$$

- If X and Y are independent, their covariance is zero and the variance of the sample mean is

$$\frac{1}{4}\left(\mathrm{Var}(X)+\mathrm{Var}(Y)\right).$$

Antithetic variables ii

- However, if X and Y are **negatively** correlated, we can actually achieve a **smaller** variance.
  - For example, if $U \sim U(0,1)$, then $X = U$ and $Y = 1 - U$ are uniform on $(0,1)$, and they are negatively correlated: $\mathrm{Cov}(U, 1-U) = -1/12$

Monotone functions

- More generally, we are interested in the following question: if f is an integrable function, $U \sim U(0,1)$, when are $f(U)$ and $f(1-U)$ **negatively** correlated.
    - Answer: when f is a **monotone** function.
- Recall the following definitions:
    - We say f is *increasing* if $f(x) \leq f(y)$ whenever $x \leq y$.
    - We say f is *decreasing* if $f(x) \geq f(y)$ whenever $x \leq y$.
    - We say f is *monotone* if f is either increasing or decreasing.

Example i

- We will look at the following integral:

$$\int_0^1 \sin\left(\frac{\pi x}{2}\right) dx.$$

- Note that on this interval, the function $f(x) = \sin\left(\frac{\pi x}{2}\right)$ is increasing.
- We will compare both the classical approach and the one based on antithetic variables.

Example ii

```r
# Classical approach
n <- 1000
unif_vars <- runif(n)
theta_hat <- mean(sin(0.5*pi*unif_vars))
sigma_hat <- sd(sin(0.5*pi*unif_vars))
c(theta_hat, sigma_hat/sqrt(n))
```

```
## [1] 0.643636085 0.009672182
```

Example iii

```r
# Antithetic variables
n <- 500
unif_vars <- runif(n)
theta_hat <- mean(sin(0.5*pi*c(unif_vars,
                                1 - unif_vars)))
sigma_hat <- sd(sin(0.5*pi*c(unif_vars,
                              1 - unif_vars)))
c(theta_hat, sigma_hat/sqrt(2*n))
```

```
## [1] 0.640439836 0.009482971
```

Example iv

- In other words, we get the same standard error **with half the number of samples**.

Example i

- We will look at the exponential expectation again:

$$\int_0^\infty \frac{e^{-x}}{1+x}dx.$$

- We know from the last module that if $U \sim U(0,1)$, we also have

$$-\log(U) \sim Exp(1), \qquad -\log(1-U) \sim Exp(1).$$

Example ii

```r
# Classical approach
n <- 1000
exp_vars <- rexp(n)
theta_hat <- mean(1/(1 + exp_vars))
sigma_hat <- sd(1/(1 + exp_vars))
c(theta_hat, sigma_hat/sqrt(n))
```

```
## [1] 0.596290451 0.007017817
```

Example iii

```r
# Antithetic variables
n <- 1000
unif_vars <- runif(n)
exp_vars <- c(-log(unif_vars), -log(1 - unif_vars))
theta2_hat <- mean(1/(1 + exp_vars))
sigma2_hat <- sd(1/(1 + exp_vars))
c(theta2_hat, sigma2_hat/sqrt(2*n))
```

```
## [1] 0.596234111 0.004990663
```

Control variates i

- **Control variates** are a more general idea than antithetic variables.
- The setting is the same: we want to estimate $\theta = E(g(X))$.
- Now, let's assume that for a function h, we know the value $\mu = E(h(X))$.
    - E.g. $h(x) = x$ implies μ is the mean of X.
- For any constant $c \in \mathbb{R}$, we can define

$$\hat{\theta}_c = g(X) + c(h(X) - \mu).$$

- **Exercise**: Check that $E(\hat{\theta}_c) = \theta$ for all c.

Control variates ii

- Let's compute the variance of $\hat{\theta}_c$:

$$\begin{aligned}\mathrm{Var}\left(\hat{\theta}_c\right) &= \mathrm{Var}\left(g(X) + c(h(X) - \mu)\right) \\ &= \mathrm{Var}\left(g(X)\right) + c^2\mathrm{Var}\left(h(X)\right) + 2c\mathrm{Cov}\left(g(X), h(X)\right).\end{aligned}$$

- The variance of $\hat{\theta}_c$ is a function of c, and it attains its minimum at

$$c^* = -\frac{\mathrm{Cov}\left(g(X), h(X)\right)}{\mathrm{Var}\left(h(X)\right)}.$$

- **No free lunch**: We still need to compute $\mathrm{Cov}\left(g(X), h(X)\right)$ and $\mathrm{Var}\left(h(X)\right)$...

Example i

- The exponential expectation:

$$\int_0^\infty \frac{e^{-x}}{1+x} dx.$$

- Let's take $h(x) = 1 + x$. Then if $X \sim Exp(1)$, we know

$$E(1+X) = 2, \qquad \mathrm{Var}(1+X) = 1.$$

Example ii

- To compute the covariance, note that

$$
\begin{aligned}
E(g(X)h(X)) &= \int_0^\infty g(X)h(X)\exp(-x)dx \\
&= \int_0^\infty \frac{1+x}{1+x}\exp(-x)dx \\
&= \int_0^\infty \exp(-x)dx \\
&= 1.
\end{aligned}
$$

Example iii

- From this, we get

$$\begin{aligned}\operatorname{Cov}\left(g(X), h(X)\right) &= E(g(X)h(X)) - E(g(X))E(h(X)) \\ &= 1 - 2E(g(X)).\end{aligned}$$

- **Wait:** we can't compute the covariance analytically without knowing $E(g(X))$. But if we knew that quantity, we wouldn't need MC integration...
    - **Solution**: Estimate $\operatorname{Cov}\left(g(X), h(X)\right)$ using the sample covariance.

Example iv

```r
n <- 1000
exp_vars <- rexp(n)
g_est <- 1/(1 + exp_vars)
h_est <- 1 + exp_vars

(c_star <- -cov(g_est, h_est)) # Var(h(X)) = 1
```

```
## [1] 0.2039655
```

Example v

```r
thetac_hat <- mean(g_est + c_star*(h_est - 2))
sigmac_hat <- sd(g_est + c_star*(h_est - 2))
c(thetac_hat, sigmac_hat/sqrt(n))
```

```
## [1] 0.602070468 0.003508384
```

```r
# Compare variance of classical MC vs control vars
(var(g_est) - sigmac_hat^2) / var(g_est)
```

```
## [1] 0.7533091
```

Example vi

- In other words, by using a control variate, we reduced the variance by approximately 75%!