Importance Sampling: Estimating Expectations (Chapter 6)

Michael Tsiang

Stats 102C: Introduction to Monte Carlo Methods

UCLA

Acknowledgements: Qing Zhou

Outline

1. Classical Monte Carlo Integration
   - Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$
   - Example 2: Standard Normal CDF
2. Importance Sampling
   - Example 3: Folded Normal Distribution

Classical Monte Carlo Integration

Let $h(x)$ be a function, and suppose we want to compute

$$I = \int_a^b h(x)\,\mathrm{d}x.$$

- The function $h(x)$ may be complicated or difficult to integrate in closed form.

- How can we approximate I (assuming it exists)?

Classical Monte Carlo Integration

- The average value of $h(x)$ on the interval (a, b) is

$$\frac{1}{b-a}\int_a^b h(x)\,\mathrm{d}x.$$

- We can rewrite the integral as average $h(x)$ in $X \in (a,b)$

$$\frac{1}{b-a}\int_a^b h(x)\,\mathrm{d}x = \int_a^b h(x)\underbrace{\frac{1}{b-a}}_{\text{uniform function}}\,\mathrm{d}x = E[h(X)],$$

where $X \sim \mathrm{Unif}(a, b)$.

- The expectation $E[h(X)]$ can be interpreted as the average value of $h(x)$ on (a, b) with respect to a uniform weight function.

Classical Monte Carlo Integration

Simple Monte Carlo Estimator (Uniform Case)

1. Generate $X^{(1)}, X^{(2)}, \ldots, X^{(n)} \overset{\text{iid}}{\sim} \text{Unif}(a, b)$. $\frac{1}{b-a}$
2. Compute $h(X^{(1)}), h(X^{(2)}), \ldots, h(X^{(n)})$.
3. Estimate $E[h(X)]$ by the **simple Monte Carlo estimator**

mean(h(xi))

$$\bar{h}_n = \frac{1}{n} \sum_{i=1}^{n} h(X^{(i)}).$$

We can then estimate I by $I = \int_a^b h(x)\,dx \Rightarrow \frac{1}{b-a} I = \int_a^b h(x) \frac{1}{b-a} dx$

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

$\therefore \; I = \bar{h}_n \cdot (b-a)$

Classical Monte Carlo Integration

after mean( )

$$
\begin{aligned}
E(\bar{h}_n) &= E\left[\frac{1}{n}\sum_{i=1}^{n} h(X^{(i)})\right] \\
&= \frac{1}{n}\sum_{i=1}^{n} E[h(X)] \\
&= E[h(X)] \quad \text{unbiased}
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Var}(\bar{h}_n) &= \mathrm{Var}\left[\frac{1}{n}\sum_{i=1}^{n} h(X^{(i)})\right] \\
&= \frac{1}{n^2}\sum_{i=1}^{n} \mathrm{Var}\left[h(X)\right] \\
&= \frac{1}{n^2} n \mathrm{Var}[h(X)] \\
&= \frac{1}{n}\mathrm{Var}[h(X)] \quad \text{biased}
\end{aligned}
$$

Classical Monte Carlo Integration

- So $E(\bar{h}_n) = E[h(X)]$ and $\mathrm{Var}(\bar{h}_n) = \dfrac{1}{n}\mathrm{Var}[h(X)]$.
- Then

$I_n = \bar{h}_n \cdot (b-a)$

$$E(\hat{I}_n) = E[(b-a)\bar{h}_n] = (b-a)E[h(X)] = I$$

$E(\hat{I}_n) = I$

and

$$\begin{aligned}\mathrm{Var}(\hat{I}_n) &= \mathrm{Var}[(b-a)\bar{h}_n] \\ &= (b-a)^2\mathrm{Var}(\bar{h}_n) \\ &= \frac{(b-a)^2}{n}\mathrm{Var}[h(X)].\end{aligned}$$

Classical Monte Carlo Integration

- Since the simple Monte Carlo estimator $\bar{h}_n$ is a sample mean, the Strong Law of Large Numbers gives LLN

$$\bar{h}_n \xrightarrow{\text{a.s.}} E[h(X)].$$

- Also, by the Central Limit Theorem, CLT

$$\frac{\bar{h}_n - E(\bar{h}_n)}{\sqrt{\operatorname{Var}(\bar{h}_n)}} = \frac{\bar{h}_n - \overset{\text{true}}{E[h(X)]}}{\sqrt{\frac{1}{n}\operatorname{Var}[h(X)]}} \xrightarrow{d} \mathcal{N}(0,1).$$

standard normal

- Note that $\operatorname{Var}(\bar{h}_n)$ can be estimated by

$$v_n := \frac{1}{n^2}\sum_{i=1}^{n}\left[h(X^{(i)}) - \bar{h}_n\right]^2 \approx \frac{1}{n}\operatorname{Var}[h(X)].$$

Classical Monte Carlo Integration

$E(\hat{I}_n) = E(\bar{h}_n)(b-a) = (b-a)E(hX)$

- Since $\hat{I}_n = (b-a)\bar{h}_n$, then the asymptotic results for $\bar{h}_n$ translate into results for $\hat{I}_n$:

$$\hat{I}_n \xrightarrow{\text{a.s.}} I$$

and

$$\frac{\hat{I}_n - E(\hat{I}_n)}{\sqrt{\text{Var}(\hat{I}_n)}} = \frac{\hat{I}_n - I}{\sqrt{\frac{(b-a)^2}{n}\text{Var}[h(X)]}} \xrightarrow{d} \mathcal{N}(0,1).$$

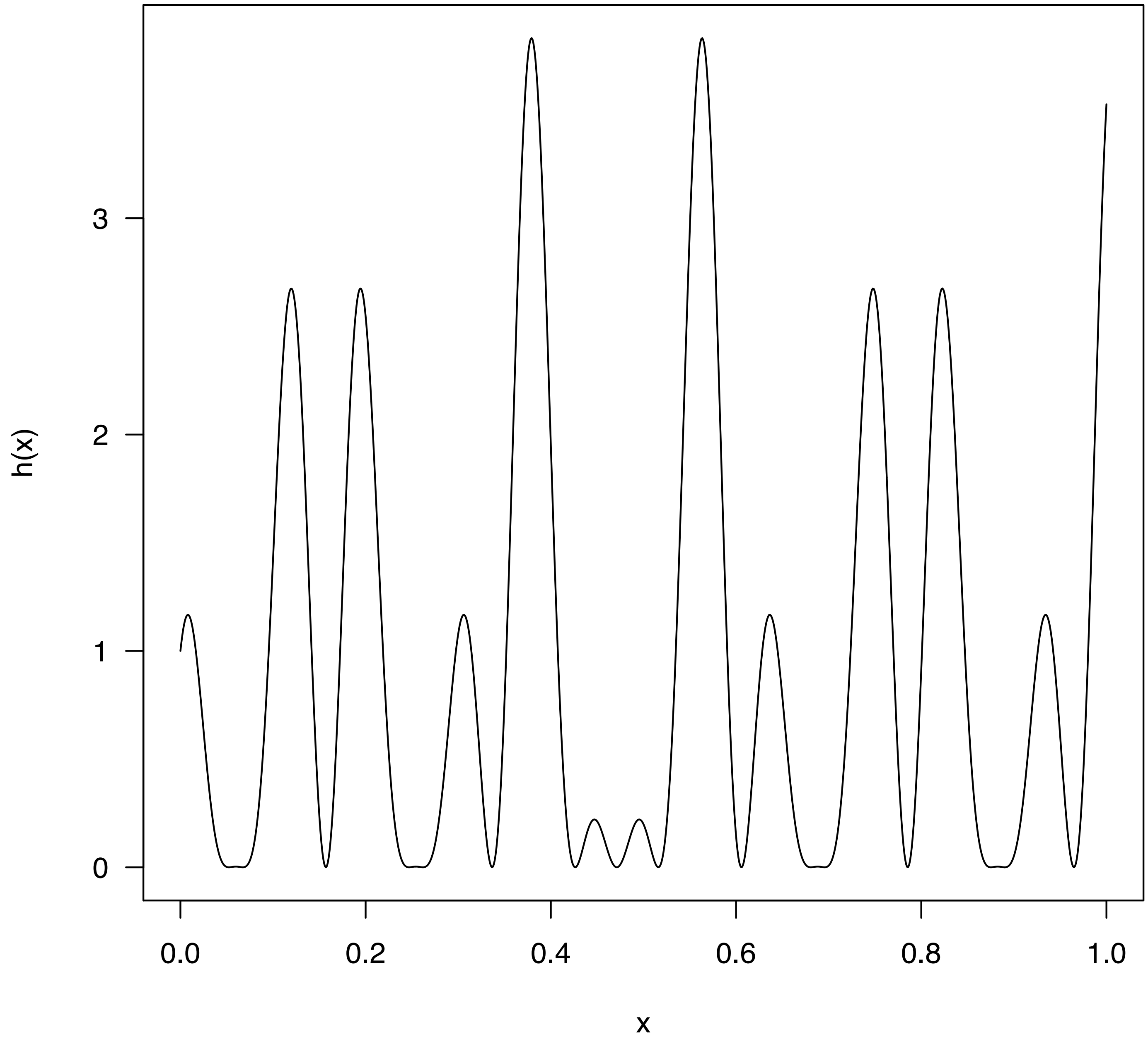
Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$

- Suppose $h(x) = [\cos(50x) + \sin(20x)]^2$, and we want to estimate

$$\int_0^1 h(x)\,\mathrm{d}x = \int_0^1 [\cos(50x) + \sin(20x)]^2\,\mathrm{d}x.$$

- This integral can be calculated in closed form, but we will estimate it using Monte Carlo integration.

Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$



Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$

R Code for the plot of $h(x)$:

```
> curve((cos(50 * x) + sin(20 * x))^2,
+       n = 1000, ylab = "h(x)", las = 1
+       )
```

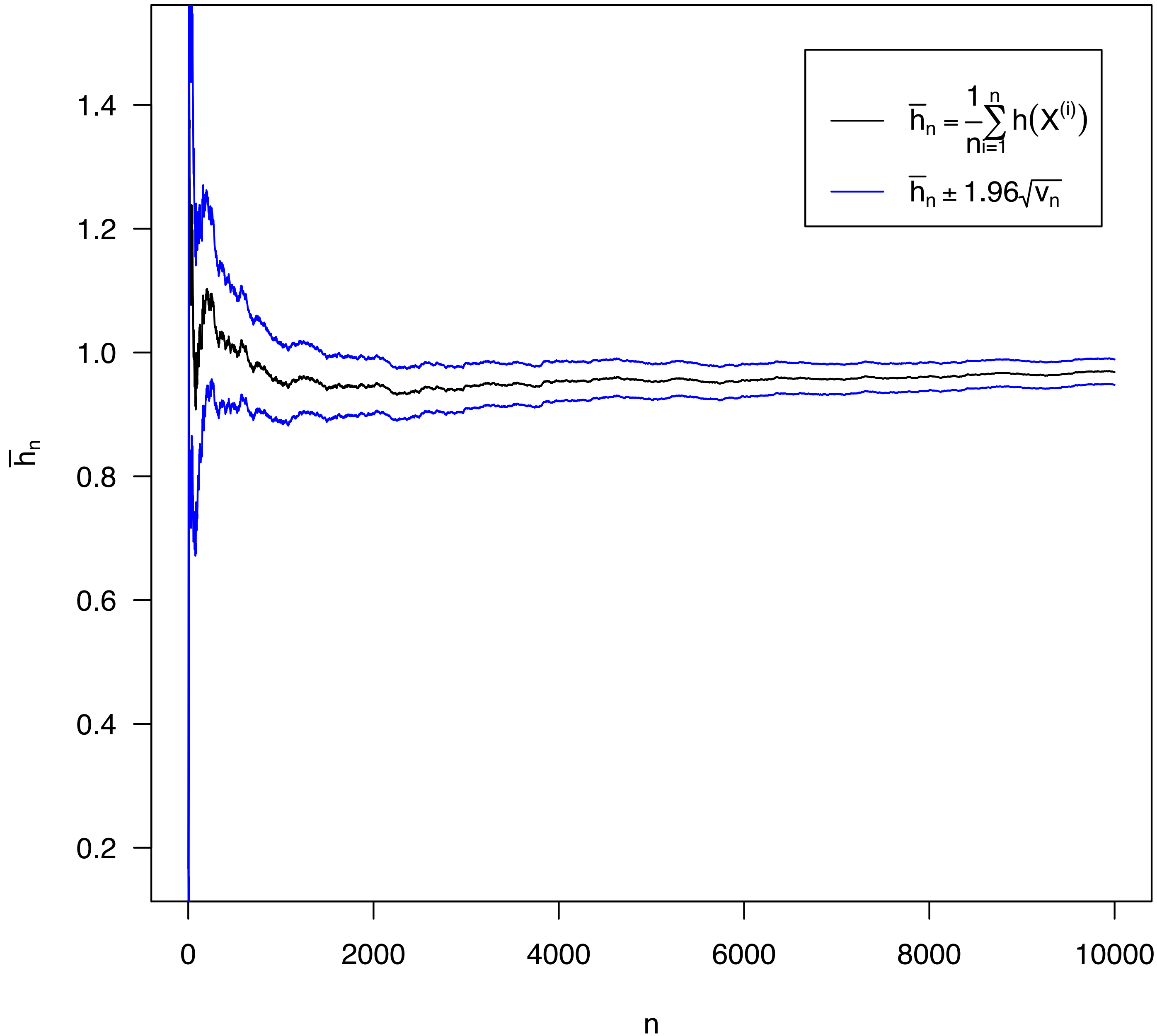
Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$

R Code to estimate $\int_0^1 [\cos(50x) + \sin(20x)]^2 \, dx$:

```
> set.seed(9999) # for reproduceability

> n <- 10000 # Specify the number of points to generate

> # Generate n points from Unif(0,1)
> X <- runif(n, 0, 1)

> # Compute h(X)
> h_X <- (cos(50 * X) + sin(20 * X))^2

> # Compute mean(h(X))
> mean(h_X)
[1] 0.9683947
```

$E(h(x))$

Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$



Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$

R Code for the plot of $\bar{h}_n$ against n:

```
> # Compute cumulative mean(h(X))
> hbar_n <- cumsum(h_X) / seq_len(n)

> # Estimate Var(hbar_n)
> var_m <- function(m){
+     # Estimate Var(hbar_m) for any given m
+     sum((h_X[seq_len(m)] - hbar_n[m])^2) / m^2
+ }

> # Compute running estimates of variance
> v_n <- vapply(seq_len(n), var_m, numeric(1))

> s_n <- sqrt(v_n) # Compute standard error
```

Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$

R Code for the plot of $\bar{h}_n$ against n:

```
> # Plot cumulative mean against iterations
> plot(hbar_n ~ seq_len(n), type = "l", xlab = "n",
+      ylab=expression(bar(h)[n])
+      )

> # Add approximate 95% confidence band
> lines(hbar_n + 1.96 * s_n, col = "blue")
> lines(hbar_n - 1.96 * s_n, col = "blue")

> # Add legend
> legend("topright", c(expression(bar(h)[n] ==
+        frac(1, n) * sum(h(X^"(i)"), i="i=1", n)),
+        expression(bar(h)[n] %+-% 1.96 * sqrt(v[n]))),
+        lty = 1, col = c("black", "blue"), inset = 0.05
+        )
```

Classical Monte Carlo Integration

Much like the uniform case of rejection sampling, there are some limitations to the simple Monte Carlo integration method:

- Drawing samples uniformly over the interval can be inefficient if the function $h(x)$ is far from uniform.
- The method does not apply to infinite (unbounded) intervals, such as $(0, \infty)$ or $(-\infty, \infty)$. $(-\infty\ 0)$ uniform fail

However, we have seen that the problem of estimating integrals can be viewed as a problem of estimating expectations, so we will reframe the problem in terms of expectations.

Classical Monte Carlo Integration

Our goal for this chapter is to estimate expectations. in region D

Suppose $X \sim f(x)$, for $x \in D$. The region D is the **support** of X:

- $f(x) > 0$, for $x \in D$
- $f(x) = 0$, for $x \notin D$
- $\displaystyle\int_D f(x)\,\mathrm{d}x = 1$

We want to compute

$$E_f[h(X)] = \int_D h(x)\overset{f(x)\text{ in }D}{f(x)}\,\mathrm{d}x = \int h(x)f(x)\,\mathrm{d}x.$$

$\frac{1}{b-a} \to$ uniform(a, b)

Classical Monte Carlo Integration

If we are able to sample from $f(x)$ directly, we can naturally generalize the simple Monte Carlo estimator:

Simple Monte Carlo Estimator (General Case)

1. Generate $X^{(1)}, X^{(2)}, \ldots, X^{(n)} \overset{\text{iid}}{\sim} f(x)$. → uniform
2. Compute $h(X^{(1)}), h(X^{(2)}), \ldots, h(X^{(n)})$.
3. Estimate $E_f[h(X)]$ by

$$\bar{h}_n = \frac{1}{n} \sum_{i=1}^{n} h(X^{(i)}).$$

Example 2: Standard Normal CDF

- Consider the CDF of the standard normal distribution $Z \sim \mathcal{N}(0, 1)$, given by

$$F(x) = P(Z \le x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-t^2/2} \, \mathrm{d}t.$$

- There is no closed form expression for $F(x)$. $E(F(x)) \rightarrow f(x)$
- We want to use a Monte Carlo estimator to estimate this integral.

No close Region for CDF

use indicator function $\longrightarrow$ CDF

Example 2: Standard Normal CDF

$Z \sim N(0,1)$

- Let $I(\cdot)$ denote the **indicator function**, so:

$$I(Z \leq x) = \begin{cases} 1 & \text{if } Z \leq x \\ 0 & \text{if } Z > x \end{cases}$$

- The expected value of $I(Z \leq x)$ is

$$\begin{aligned} E[I(Z \leq x)] &= 1 \cdot P(Z \leq x) + 0 \cdot P(Z > x) \\ &= P(Z \leq x) \\ &= \underline{F(x)}. \end{aligned}$$

- We have expressed the integral of interest as an expectation, so we can use a Monte Carlo estimator to estimate $F(x)$.

Example 2: Standard Normal CDF

We have $F(x) = P(Z \leq x) = E[I(Z \leq x)]$, so $h(x) = I(Z \leq x)$.

1. Generate $Z^{(1)}, Z^{(2)}, \ldots, Z^{(n)} \overset{\text{iid}}{\sim} \mathcal{N}(0, 1)$.

2. For each $Z^{(i)}$, compute indicator function

$$h(Z^{(i)}) = I(Z^{(i)} \leq x) = \begin{cases} 1 & \text{if } Z^{(i)} \leq x \\ 0 & \text{if } Z^{(i)} > x. \end{cases}$$ if else

3. Estimate $F(x)$ by

how many "1"

$$\widehat{F(x)} = \bar{h}_n = \frac{1}{n} \sum_{i=1}^{n} I(Z^{(i)} \leq x).$$

empirical cdf

Note: This method generalizes to produce an estimator for the CDF of any random variable (if we can sample from its distribution).

Example 2: Standard Normal CDF

- Notice that the random variable Beroulli

$$I(Z \leq x) = \begin{cases} 1 & \text{if } Z \leq x \\ 0 & \text{if } Z > x \end{cases}$$

is a Bernoulli random variable with success probability $p = P(Z \leq x) = F(x)$.

$p = p(I(Z \leq x) = 1) = p(Z \leq x) = F(x)$

- The estimator

$$\widehat{F(x)} = \bar{h}_n = \frac{1}{n} \sum_{i=1}^{n} I(Z^{(i)} \leq x)$$

$\begin{cases} E(\hat{F}(x)) = p \\ var(\hat{F}(x)) = \frac{\hat{F}(x) \cdot (1 - \hat{F}(x))}{n} \end{cases}$

$= \frac{p(1-p)}{n}$

is thus the $\boxed{\text{sample proportion of successes in } n \text{ trials.}}$

if $Z \leq x \rightarrow I = 1$

Example 2: Standard Normal CDF

- Since

$$\overset{p}{\widehat{F(x)}} = \bar{h}_n = \frac{1}{n}\sum_{i=1}^{n} I(Z^{(i)} \leq x) \qquad \frac{S}{N}$$

is the sample proportion of successes in n Bernoulli trials, then

$$E[\widehat{F(x)}] = p = F(x) \quad \text{and} \quad \mathrm{Var}[\widehat{F(x)}] = \frac{F(x)[1 - F(x)]}{n}.$$

- The maximum variance occurs when $F(x) = \dfrac{1}{2}$, so a conservative estimate of $\mathrm{Var}[\widehat{F(x)}]$ is $\dfrac{1}{4n}$.

$$\frac{(1-\frac{1}{2})\frac{1}{2}}{n} = \frac{1}{4n}$$

Outline

1. Classical Monte Carlo Integration
   - Example 1: $h(x) = [\cos(50x) + \sin(20x)]^2$
   - Example 2: Standard Normal CDF
2. Importance Sampling
   - Example 3: Folded Normal Distribution

Importance Sampling

Suppose $X \sim f(x)$, for $x \in D$, where D is the support of X:

- $f(x) > 0$, for $x \in D$ positive density
- $f(x) = 0$, for $x \notin D$ 0 density
- $\int_D f(x)\,\mathrm{d}x = 1$ Indicator function (is else)

We want to compute

$$E_f[h(X)] = \int_D h(x)f(x)\,\mathrm{d}x = \int h(x)f(x)\,\mathrm{d}x.$$

What if we are not able to sample from $f(x)$ directly?

Importance Sampling use trail distribution – rejection | inverse

Borrow intuition from rejection sampling!

Find a **trial** or **candidate distribution** $g(x)$ such that:

(i) The support of $g(x)$ contains the support of $f(x)$, i.e.,

$$g(x) > 0, \text{ for all } x \in D.$$

(ii) We can sample from $g(x)$.

How do we use the trial distribution $g(x)$ to compute

$$E_f[h(X)] = \int_D h(x) f(x)\, \mathrm{d}x,$$

an expectation in terms of $f(x)$?

Key Idea: Express $E_f[h(X)]$ as an expectation in terms of $g(x)$!

$$
\begin{aligned}
E_f[h(X)] &= \int_D h(x) f(x)\,\mathrm{d}x \\
&= \int_D h(x) f(x) \frac{g(x)}{g(x)}\,\mathrm{d}x \qquad \to 1 \\
&= \int_D h(x) \frac{f(x)}{g(x)} g(x)\,\mathrm{d}x \\
\begin{pmatrix} f(x) = 0 \\ \text{for } x \notin D \end{pmatrix} &= \int h(x) \frac{f(x)}{g(x)} g(x)\,\mathrm{d}x \\
&= E_g\left[h(X) \frac{f(X)}{g(X)} \right]
\end{aligned}
$$

change $f \longrightarrow g$

Importance Sampling

We have shown that

$$\boxed{E_f[h(X)] = E_g\left[h(X)\frac{f(X)}{g(X)}\right].}$$

Since we can sample from $g(x)$, we can generate

$$X^{(1)}, X^{(2)}, \ldots, X^{(n)} \sim g(x)$$

and use the simple Monte Carlo estimator

$$\frac{1}{n}\sum_{i=1}^{n} h(X^{(i)})\frac{f(X^{(i)})}{g(X^{(i)})} \approx E_g\left[h(X)\frac{f(X)}{g(X)}\right] = E_f[h(X)].$$

Sampling

Importance Sampling

Definition

The **importance weight** of $X^{(i)}$ is defined by

$$w(X^{(i)}) = \frac{f(X^{(i)})}{g(X^{(i)})}.$$

weighting x^i

If $X^{(1)}, X^{(2)}, \ldots, X^{(n)} \sim g(x)$, then

$$E_f[h(X)] \approx \frac{1}{n}\sum_{i=1}^{n} w(X^{(i)})h(X^{(i)}).$$

$f(x^i) = g(x^i)$

If $X^{(1)}, X^{(2)}, \ldots, X^{(n)} \sim f(x)$, then

weight $=1$

$$E_f[h(X)] \approx \frac{1}{n}\sum_{i=1}^{n} 1 \cdot h(X^{(i)}).$$

If we can sample from $f(x)$, the importance weights are all 1.

Importance Sampling

1. Generate $X^{(1)}, X^{(2)}, \ldots, X^{(n)} \sim g(x)$, and compute the importance weights
$$w(X^{(i)}) = \frac{f(X^{(i)})}{g(X^{(i)})}, \text{ for } i = 1, 2, \ldots, n.$$

2. Estimate $E_f[h(X)]$ by the **importance sampling estimator**
$$\widehat{E_f[h(X)]} = \frac{1}{n}\sum_{i=1}^{n} w(X^{(i)})h(X^{(i)}).$$

Example 3: Folded Normal Distribution

- Suppose we want to estimate $E_f(X)$, where $f(x)$ is the PDF of the folded normal distribution,

$$f(x) = 2 \cdot \frac{1}{\sqrt{2\pi}} e^{-x^2/2} = \sqrt{\frac{2}{\pi}} e^{-x^2/2}, \qquad \text{for } x \geq 0.$$

- The support of $f(x)$ is $D = [0, \infty)$:
  - $f(x) > 0$ for all $x \in D$
  - $\int_0^\infty f(x)\,\mathrm{d}x = 1$

- We want to use importance sampling to estimate $E_f(X)$. (Notice that $h(x) = x$ for this example.)

Example 3: Folded Normal Distribution

$D = [0, \infty)$

Consider the PDF of $\mathrm{Exp}(\lambda = 2)$, given by

$$g(x) = 2e^{-2x}, \quad \text{for } x \geq 0.$$

We can check that $g(x)$ satisfies the conditions to be a suitable trial distribution:

1. The support of $g(x)$ contains the support of $f(x)$, i.e.,

$$g(x) > 0, \text{ for all } x \in D.$$

   (The support of $g(x)$ is actually the same as $D = [0, \infty)$ in this case.)

2. We can sample from $g(x)$ using the inverse CDF method.

Example 3: Folded Normal Distribution

Importance sampling to estimate $E_f(X)$:

1. Generate $X^{(1)}, X^{(2)}, \ldots, X^{(n)} \sim \mathrm{Exp}(\lambda = 2)$, and compute the importance weights

$$\begin{aligned} w(X^{(i)}) &= \frac{f(X^{(i)})}{g(X^{(i)})} \\ &= \frac{\sqrt{\frac{2}{\pi}} e^{-(X^{(i)})^2/2}}{2e^{-2X^{(i)}}} \\ &= \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{(X^{(i)})^2}{2} + 2X^{(i)} \right]. \end{aligned}$$

2. Estimate $E_f(X)$ by

$f(x)=x$

$$\widehat{E_f(X)} = \frac{1}{n} \sum_{i=1}^{n} w(X^{(i)}) X^{(i)}.$$

Example 3: Folded Normal Distribution

R Code to estimate $E_f(X)$ for the folded normal distribution:

```
> set.seed(9999) # for reproduceability

> n <- 10000 # Specify the number of points to generate

> # Generate n points from Exp(lambda = 2)
> X <- rexp(n, rate = 2)
```

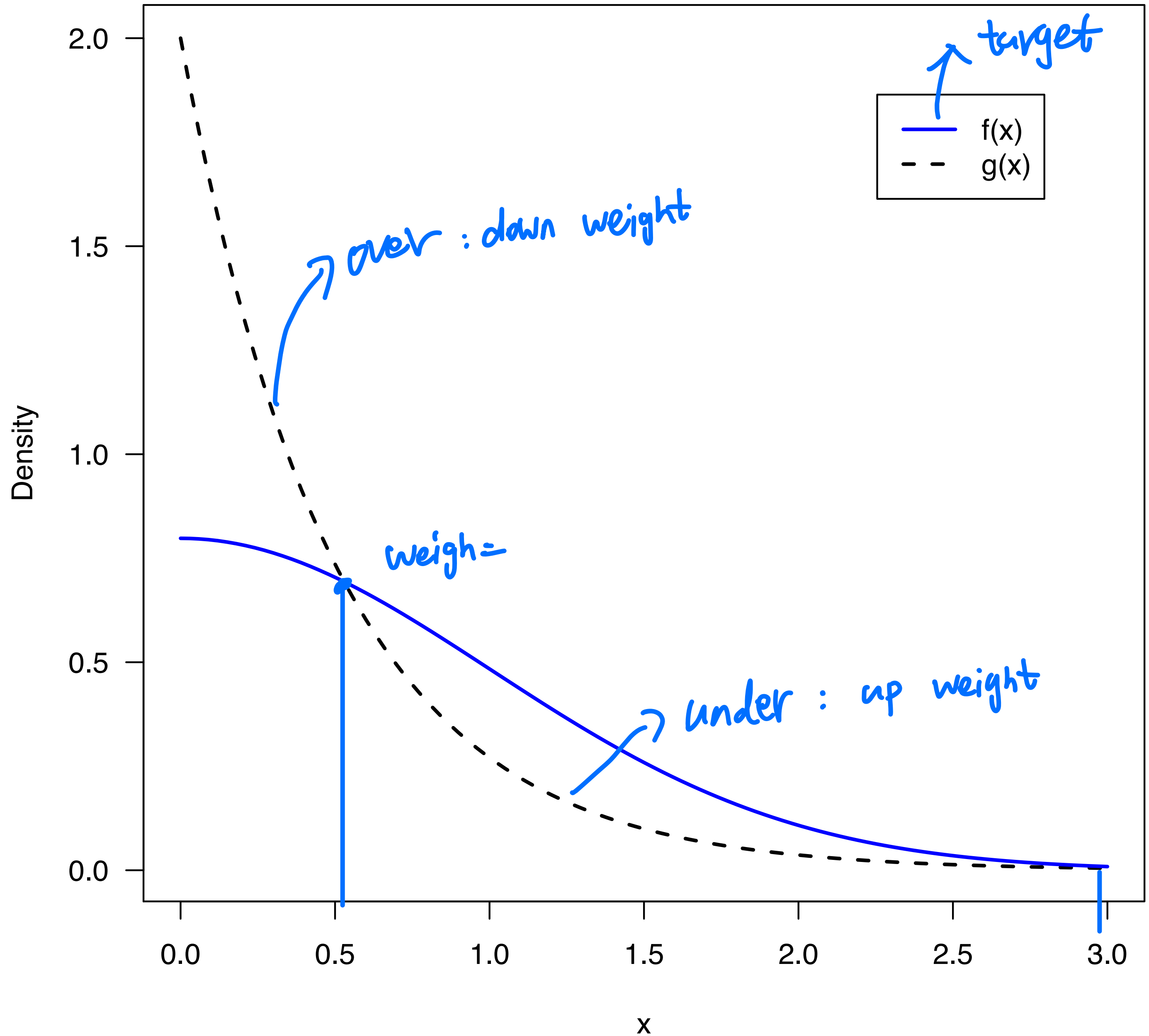
~ exp $g(x)$

```
> # Compute importance weights
```

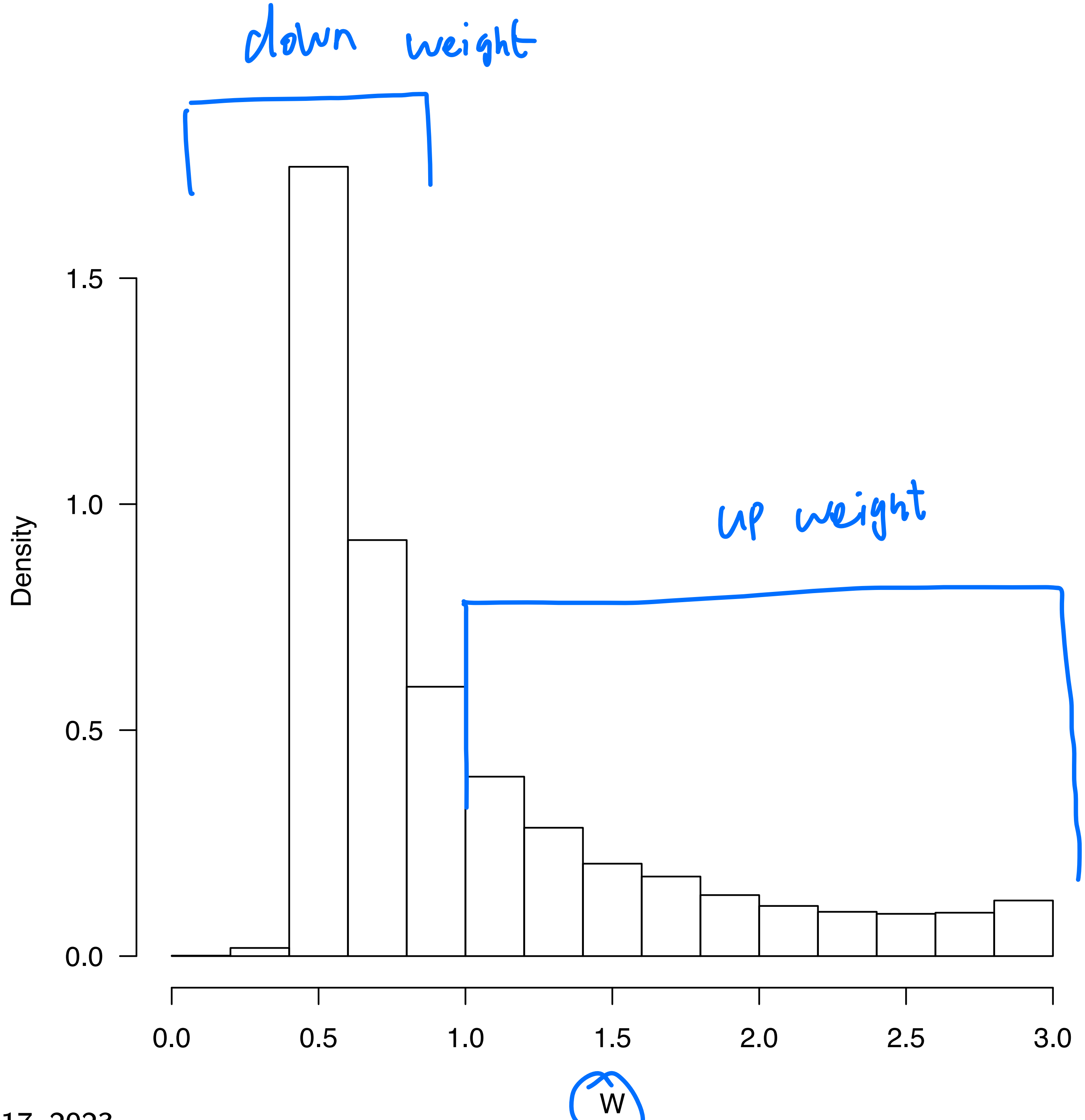
weight function

```
> W <- exp(-X^2 / 2 + 2 * X) / sqrt(2 * pi)

> # Compute mean(w(X) * h(X)) (h(X) = X here)
> mean(W * X)
[1] 0.801565

> # Theoretical value
> sqrt(2 / pi)
[1] 0.7978846
```

Example 3: Folded Normal Distribution



Example 3: Folded Normal Distribution



R Code for the plots:

```
> # Plot of f(x) and g(x)
> curve(2 * exp(-2 * x),
+       lty = 2, lwd = 2, 0, 3, las = 1, ylab = "Density"
+       )
> curve(sqrt(2 / pi) * exp(-x^2 / 2),
+       col = "blue", lwd = 2, add = TRUE
+       )
> legend("topright", c("f(x)", "g(x)"),
+        lty = 1:2, lwd = 2,
+        col = c("blue", "black"), inset = 0.1
+        )

> # Histogram of importance weights
> hist(W, prob = TRUE, las = 1, main = "")
```

Example 3: Folded Normal Distribution

Theoretical calculation of $E_f(X)$:

$$
\begin{aligned}
E_f(X) &= \int_0^\infty x\sqrt{\frac{2}{\pi}}e^{-x^2/2}\,\mathrm{d}x \\
&= \sqrt{\frac{2}{\pi}}\int_0^\infty e^{-x^2/2}x\,\mathrm{d}x \\
\begin{pmatrix} \mathrm{d}x^2 = 2x\,\mathrm{d}x \\ x\,\mathrm{d}x = \frac{1}{2}\,\mathrm{d}x^2 \end{pmatrix} &= \sqrt{\frac{2}{\pi}}\int_0^\infty \frac{1}{2}e^{-x^2/2}\,\mathrm{d}x^2 \\
&= \sqrt{\frac{2}{\pi}}\left[-e^{-x^2/2}\right]_0^\infty \\
&= \sqrt{\frac{2}{\pi}}[0-(-1)] \\
&= \sqrt{\frac{2}{\pi}}
\end{aligned}
$$

So $E_f(X) = \sqrt{\frac{2}{\pi}} \approx 0.7978846.$ by math

Importance Sampling

- What is the mean of the importance weights $w(X^{(i)})$?
- By the Strong Law of Large Numbers,

$$\frac{1}{n}\sum_{i=1}^{n} w(X^{(i)}) \xrightarrow{\text{a.s.}} E_g[w(X^{(i)})] = \int \frac{f(x)}{g(x)} g(x)\,\mathrm{d}x = 1.$$

- The mean of the importance weights should be close to 1.

converge to 1

Importance Sampling

We can interpret the efficiency of the importance sampling estimator as how close $g(x)$ is to $f(x)$:

weight is less more efficient

- The closer $g(x)$ is to $f(x)$, the more efficient the estimator will be.

match

- If $g(x) = f(x)$, then the importance weights would all be 1.
- The further $g(x)$ is from $f(x)$, the more variability there will be in the importance weights. more weigh less efficient
- We can thus compute the efficiency by

$$\text{Efficiency} = \frac{1}{\text{Var}_g[w(X)]}.$$

→ how far g(x) and f(x)