Week 02: Monte Carlo Integration

Mark M. Fredrickson (mfredric@umich.edu)

Computational Methods in Statistics and Data Science (Stats 406)

Expectation

Suppose we are going to compute $g(X)$ for a random variable X.

We often want to "average over" X to get a sense of a typical value for $g(X)$. We define the **expectation** of $g(X)$ as:

$$E(g(X)) = \sum_{i=-\infty}^{\infty} P(X = x)g(x) \qquad \text{(discrete)}$$

$$E(g(X)) = \int_{-\infty}^{\infty} g(x)f(x)\,dx \qquad \text{(continuous)}$$

Expectation and Inference

Recall, a **parameter** is an **unknown quantity** in a statistical model.

We previously discussed two forms of **statistical inference** for parameters:

- **Estimation**: making informed guesses about population values.
- **Testing**: checking if the data conform to a specific value of the parameter.

Expectations are useful for both:

- Important **operating characteristics** of estimators (e.g. bias)
- Computing **Type I and Type II** of tests
- Many parameters can be **expressed as expectations**.

Monte Carlo Integration

Recall your introduction performing integrals using **Riemann sums**:

$$\int_a^b h(x)\,dx \approx \sum_{i=0}^{n} h(a + di/2) \times d, \quad d = \frac{b-a}{n}$$

(or using the trapezoid rule or any other similar technique).

Straightforward, but

- How do you pick d? Alternatively, if spacing is unequal, how do you pick the regions?
- When integrating in k-dimensions we need to take n^k samples (gets big fast!)

Solution: let $h(x)$ help us pick the most important regions and integrate using randomly selected points.

Example: Universal function integrator

Suppose we want to evaluate a complex function over the region 0 to 1 (for a math class).

For example,

$$\int_0^1 \left(\log\left(\frac{1}{x}\right)\right)^3 dx$$

This integral doesn't have a **closed form solution**, so our usual techniques do not work. (BTW: This is the $\Gamma(4)$ function.)

Example continued

Recall: if $U \sim U(0,1)$ then the density function is $f(u) = 1$.

$$\int_0^1 \left(\log\left(\frac{1}{x}\right)\right)^3 dx = \int_0^1 \left(\log\left(\frac{1}{x}\right)\right)^3 f(x)\, dx = E\left(g(U)\right)$$

where

$$g(X) = \left(\log\left(\frac{1}{x}\right)\right)^3$$

Conveniently, computers are great at generating lots of $U(0,1)$ random variables!

We'll **approximate** $E(g(U))$ with **the sample mean of** $g(U_i)$.

Monte Carlo Gamma Function

We **estimate the integral** using draws from $U(0, 1)$ and the sample mean of the function values:

```
> g <- function(u) { log(1/u)^3 }
> mean(g(runif(1000000)))
```

```
[1] 6.014
```

Since the exponent was integer, $\Gamma(a) = (a - 1)!$, in this case $\mathbf{3! = 6}$.

Using other distributions

We are **not limited to the uniform distribution** when picking distributions to use.

Suppose X is a random variable with:

- Support $[a, b]$ (where either $a \to -\infty$ or $b \to \infty$)
- Density $f(x)$ that is non-zero over $[a, b]$

then

$$\int_a^b g(x)\,dx = \int_a^b g(x)\frac{f(x)}{f(x)}\,dx = \int_a^b \frac{g(x)}{f(x)}f(x)\,dx = \mathrm{E}\left(h(X)\right)$$

where $h(x) = g(x)/f(x)$.

Example: Integral of $1/2^x$

Suppose we need to compute

$$\int_0^\infty \frac{1}{2^x}\,dx$$

What kind of distribution has the support $[0, \infty)$?

The **exponential distribution** has

- Support $[0, \infty)$
- Density function $f(x) = \exp\{-x\}$ (keeping the usual parameter $\lambda = 1$)
- Random number generator `rexp`

Example continued

We've identified a variable on $[0, \infty)$, what expectation should we estimate?

Let $g(x) = 1/2^x$.

$$\int_0^\infty g(x)\,dx = \int_0^\infty \frac{g(x)}{\exp\{-x\}} \exp\{-x\}\,dx = \mathrm{E}\left(h(X)\right)$$

Implementing

```
> g <- function(x) {
+   1/(2^x)
+ }
> h <- function(x) { g(x) / dexp(x) } ## R's exp. density function

> k <- 100000
> hX <- h(rexp(k))
> mean(hX)

[1] 1.442

> 1/log(2)

[1] 1.443
```

Distributions in R

R has a naming convention for functions related to **distributions** based on **prefixes**:

- `r`: generated random values from the distribution (deviates)
- `d`: evaluates the probability density or mass function at its argument
- `p`: implements the CDF for $P(X \leq x)$
- `q`: implements the quantile function (inverse of CDF) and finds x such that $P(X \leq x) = p$.

See `?Distributions` for a list of built in distributions with these functions.

Estimating a mean (in general)

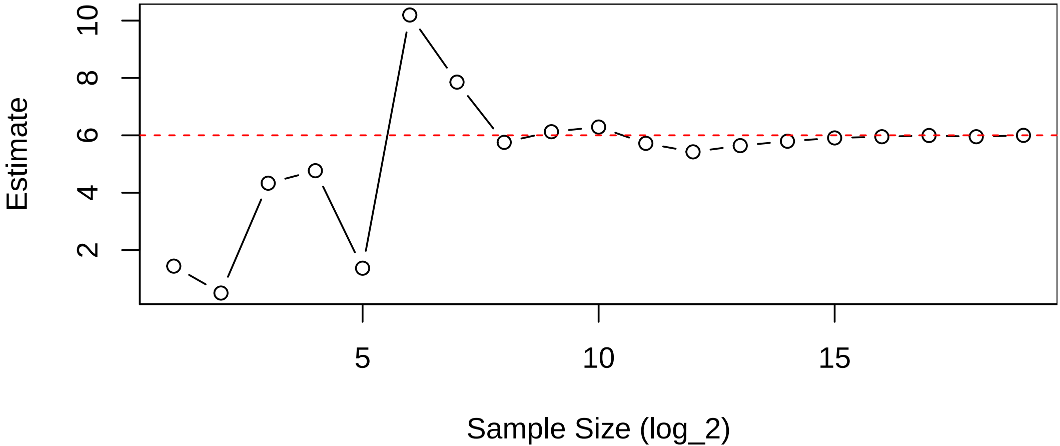
Why does this work? Suppose that $X_i \overset{\text{iid}}{\sim} F$ and $E(g(X)) < \infty$. The **sample mean of $g(X_i)$** is **unbiased** for the population quantity $E(g(X))$.

$$\begin{aligned} E\left(\frac{1}{n}\sum_{i=1}^{n} g(X_i)\right) &= \frac{1}{n}\sum_{i=1}^{n} E(g(X_i)) \\ &= \frac{1}{n} nE\left(g(X_1)\right) \\ &= E(g(X)) \end{aligned}$$

Provided we can sample from F, **the sample mean is a good estimator of $E(g(X))$**.

Larger sample → better estimates

```
> g <- function(u) { log(1/u)^3 }
> ms <- map_dbl(2:20, ~ mean(g(runif(2^.x))))
```



When $\mathrm{E}\left(g(X)\right) < \infty, \mathrm{E}\left(g(X)^2\right) < \infty$ and X_i are IID, by the **weak law of large numbers**,

$$\hat{\theta}_n = \frac{1}{n}\sum_{i=1}^{n} g(X_i) \xrightarrow{P} \mathrm{E}\left(g(X)\right) = \theta$$

as $n \to \infty$.

Reminder: The notation $\xrightarrow{P}$ indicates "convergence in probability":

$$\hat{\theta}_n \xrightarrow{P} \theta \equiv \lim_{n\to\infty} \Pr(|\hat{\theta}_n - \theta| < \epsilon) = 1, \quad \text{for any } \epsilon > 0$$

In words: we can draw a large enough sample to concentrate all the probability of $\hat{\theta}_n$ within $\pm\epsilon$ of θ.

Uncertainty in estimating $\mathrm{E}\left(g(X)\right)$

Under these same conditions ($X_i \overset{\text{iid}}{\sim} F, \mathrm{E}\left(g(X)\right) < \infty, \mathrm{E}\left(g(X)^2\right) < \infty$), the **central limit theorem** states:

$$\frac{\hat{\theta}_n - \theta}{\hat{s}_n/\sqrt{n}} \xrightarrow{D} N(0,1) \quad \text{(convergence in distribution)}$$

where:

$$\hat{\theta}_n = \frac{1}{n}\sum_{i=1}^{n} g(X_i), \quad \hat{s}_n = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(g(X_i) - \hat{\theta}_n)^2}$$

Confidence Intervals for θ

A typical use for the CLT is creating **confidence intervals** (a plausible range) for parameters.

By the CLT,

$$\hat{\theta}_n \approx N(\theta, \tau^2/n), \quad \tau^2 = \mathrm{Var}(g(X))$$

an approximate $100 \times (1 - \alpha)\%$ **confidence interval for θ** is given by:

$$\hat{\theta} \pm t_{\alpha/2}(n-1) \times \hat{s}/\sqrt{n}$$

where $t_{\alpha/2}(n-1)$ is the $\alpha/2$ quantile of Student's t-distribution with $n-1$ degrees of freedom.

Confidence Intervals in R

To estimate

$$\int_0^\infty \frac{1}{2^x}\, dx$$

we used an exponential random variable so generate the estimate:

```
> mean(hX)
```

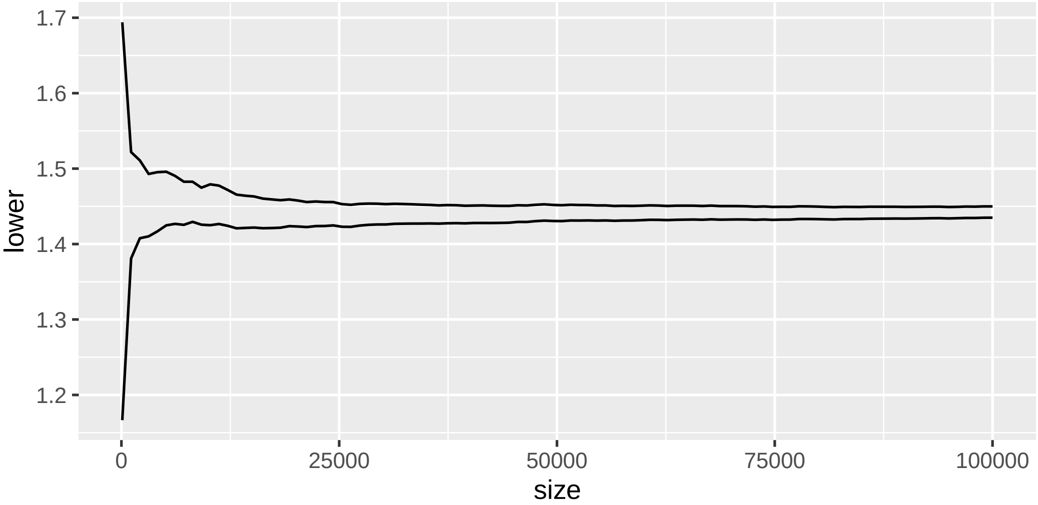
```
[1] 1.442
```

To add a 99.9% confidence intervals:

```
> ## t.test will do other things, we only need the conf.int
> t.test(hX, conf.level = 0.999)$conf.int
```

```
[1] 1.435 1.450
attr(,"conf.level")
[1] 0.999
```

Confidence Intervals with Larger Samples



Statistical Integrals

Statistical Integrals

Thus far, we've have not considered **why we want to integrate** a given function.

Many tasks in statistical inference are **based on integrals**:

- Computing probabilities of events
- Computing expectations
- The cumulative distribution functions
- Operating characteristics of estimators and tests

Estimating Means and Variances of RVS

Previously we saw a random variable with the following distribution:

$$f(x) = \theta x^{\theta-1}$$

and

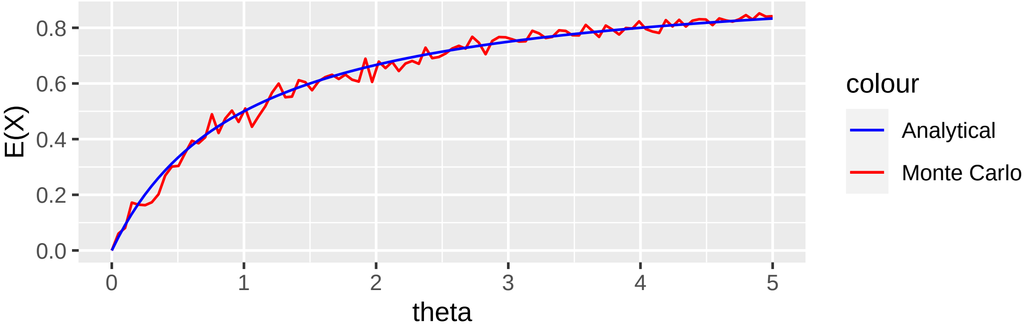
$$E(X) = \frac{\theta}{\theta+1}$$

This is a special case of a **Beta random variable** (with $\alpha = \theta$ and $\beta = 1$, in the usual parameterization).

Let's see if we get the same answer with Monte Carlo integration.

Beta example

```
> k <- 100 ## intentionally small sample!
> thetas <- seq(0, 5, length.out = 100)
> estimated_means <- map_dbl(thetas, ~ mean(rbeta(k, .x, 1)))
```



Beta' distribution

If $X \sim \text{Beta}(\alpha, \beta)$ then

$$Y = \frac{X}{1 - X}$$

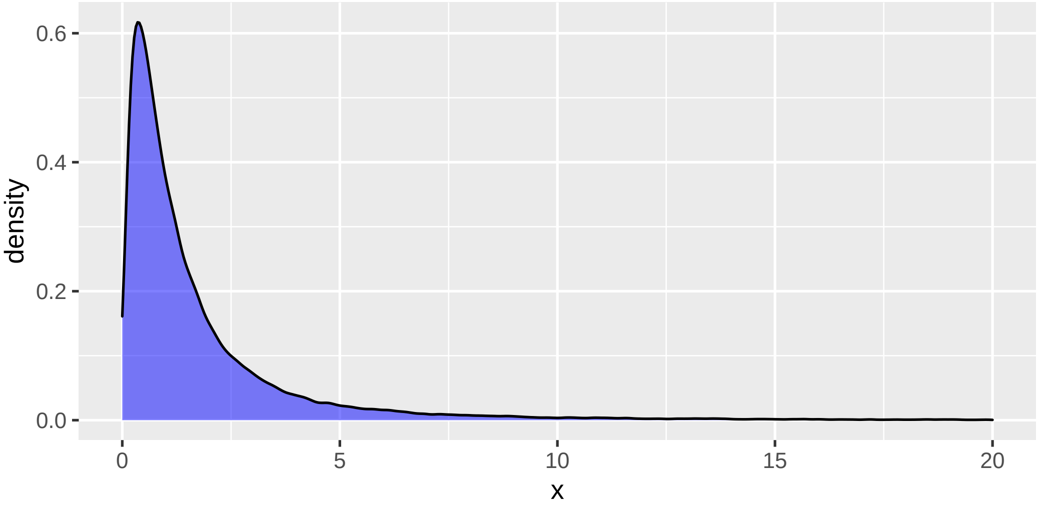
has a **Beta' (prime)(α, β)** distribution (used to model wait times for extremely rare events).

The Beta' distribution is not built into R, but we can create them:

```
> k <- 100000
> x <- rbeta(k, 2, 2)
> y <- x / (1 - x)
```

(We'll explore this idea more in the coming weeks.)

Density of Beta'(2,2)



Mean of Beta'(2,2)

```
> mean(y) ## should be about 2 / (2 - 1) = 2

[1] 2.014

> t.test(y)$conf.int

[1] 1.980 2.048
attr(,"conf.level")
[1] 0.95
```

Estimating a CDF

Recall for a continuous RV with density f, the **cumulative distribution function** is defined by

$$F(t) = \int_{-\infty}^{t} f(x)\, dx$$

Suppose we wanted to compute the CDF of the **log-Normal distribution**:

$$X = \exp(Z), \quad Z \sim N(0,1) \Rightarrow X > 0$$

A little calculus shows it has PDF:

$$f(x) = \frac{1}{x\sqrt{2\pi}} \exp\left(-\frac{[\log(x)]^2}{2}\right)$$

Estimating CDF, cont.

Let's estimate $F(1.25)$. We can't use $U(0,1)$ but we can use $W \sim U(0, 1.25)$:

$$F(1.25) = \int_0^{1.25} f(x)\,dx = 1.25 \int_0^{1.25} f(x) \frac{1}{1.25}\,dx = 1.25\,\mathrm{E}\left(f(W)\right)$$

```
> f <- function(x) { 1 / (x * sqrt(2 * pi)) * exp(- log(x)^2 / 2)}
> gW <- 1.25 * f(runif(10e6, min = 0, max = 1.25))
> mean(gW)
```

```
[1] 0.5882
```

R has a built-in version of the log-Normal CDF:

```
> plnorm(1.25)
```

```
[1] 0.5883
```

Example: 99% CI for log-Normal CDF

Recall we had:

```
> mean(gW)

[1] 0.5882

> t.test(gW, conf.level = 0.99)$conf.int

[1] 0.5880 0.5884
attr(,"conf.level")
[1] 0.99
```

Probability as an Integral

Many statistical tasks can be viewed as taking an expectation for a suitable function g.

In particular, if g is an **indicator function**, then the expectation of g is a **probability**.

Indicator functions:

$$I(s) = \begin{cases} 1 & : s \text{ is true} \\ 0 & : s \text{ is false} \end{cases}$$

Note: In R we get indicators "for free" by writing things like: `x <= 3`.

Expectation of Indicators

Suppose we have a **continuous random variable** X and g **be an indicator function**. For example, $g(X) = I(X > 3)$.

Since $g(X)$ can be either 1 or 0 (it is a **random variable**), let A **be the event** (values of X) that $g(X)$ is 1.

Notice: A and A^c partition the sample space.

Claim: $E(g(X)) = P(A)$

Example: $E(I(X > 3)) = P(X > 3)$

Proof

Remember: $g(X) = 1$ if A is true and $g(X) = 0$ if A^c is true ("not A").

$$
\begin{aligned}
E(g(X)) &= \int_{-\infty}^{\infty} g(x) f(x)\,dx \\
&= \int_A 1 \times f(x)\,dx + \int_{A^c} 0 \times f(x)\,dx \\
&= \int_A f(x)\,dx \\
&= P(A)
\end{aligned}
$$

Estimating a Cumulative Distribution Function

Again, the CDF of a (continuous) random variable is defined as:

$$\begin{aligned} F(t) &= P(X \leq t) \\ &= \int_{-\infty}^{t} f(x)\,dx \\ &= \int_{-\infty}^{t} 1 \times f(x)\,dx + \int_{t}^{\infty} 0 \times f(x)\,dx \\ &= \int_{-\infty}^{\infty} I(x \leq t)\, f(x)\,dx \\ &= E(I(X \leq t)) \end{aligned}$$

The **empirical CDF** is then the sample mean of $I(X_i \leq t)$:

$$\hat{F}_n(t) = \frac{1}{n} \sum_{i=1}^{n} I(X_i \leq t)$$

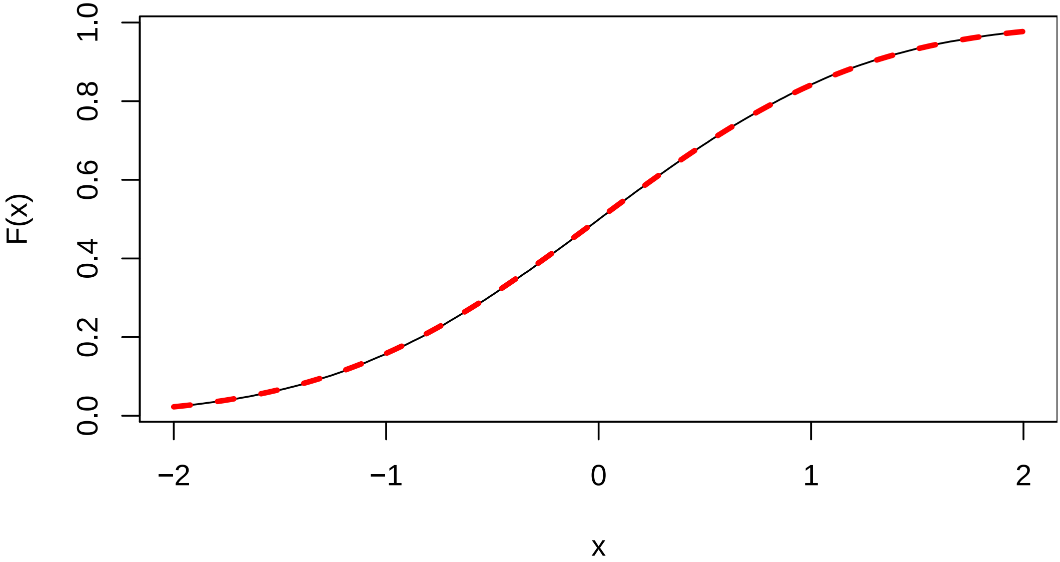
Example: Normal CDF

The CDF of the Normal distribution **does not have a closed form**.

For $X \sim N(0, 1)$, let's estimate $P(X \leq 1.96)$.

```
> xs <- rnorm(1e5)
> mean(xs <= 1.96)

[1] 0.9751
```


1.0
0.8
0.6
0.4
0.2
0.0
F(x)
−2
−1
0
1
2
x

Better Confidence Intervals for Estimated Probabilities

The **t-distribution intervals** we used earlier would work for when estimating a probability, but **we can do slightly better**.

Consider the random variable $W = I(X \leq t)$. **What is this?**

- W can take values of 0 or 1
- $P(W = 1) = P(X \leq t) = \theta$, $P(W = 0) = 1 - \theta$.
- W is **Bernoulli**!

We have to estimate θ (our goal anyway), but this means we can use **confidence intervals for proportions** (i.e., `binom.test` instead of `t.test`).

Example: Log-Normal revisited

Recall, we estimated $P(X \leq 1.25)$, for $X = \exp(Z), Z \sim N(0,1)$ using uniform random variables $W \sim U(0, 1.25)$.

As an alternative method, we can sample from X directly:

```
> xs <- exp(rnorm(10e6)) ## rnorm gives random N(0,1)
> mean(xs <= 1.25)

[1] 0.5885

> binom.test(sum(xs <= 1.25), n = length(xs))$conf.int

[1] 0.5882 0.5888
attr(,"conf.level")
[1] 0.95
```

Picking n to achieve a width

After we find a way to estimate, we can pick n to **achieve given precision**.

The CI width will be approximately:

$$w = 2z_{\alpha/2}\tau/\sqrt{n} \Rightarrow n = 4z_{\alpha_2}^2\tau^2/w^2$$

where $z_{\alpha/2} = P(Z \leq \alpha/2)$, $Z \sim N(0,1)$ and $\tau^2 = \mathrm{Var}(g(X))$.

We need to **make a good guess** for τ^2.

- Estimate using a small sample.
- Find an upper bound (e.g., bounded $g(X_i)$).

Example: Finding an upper bound

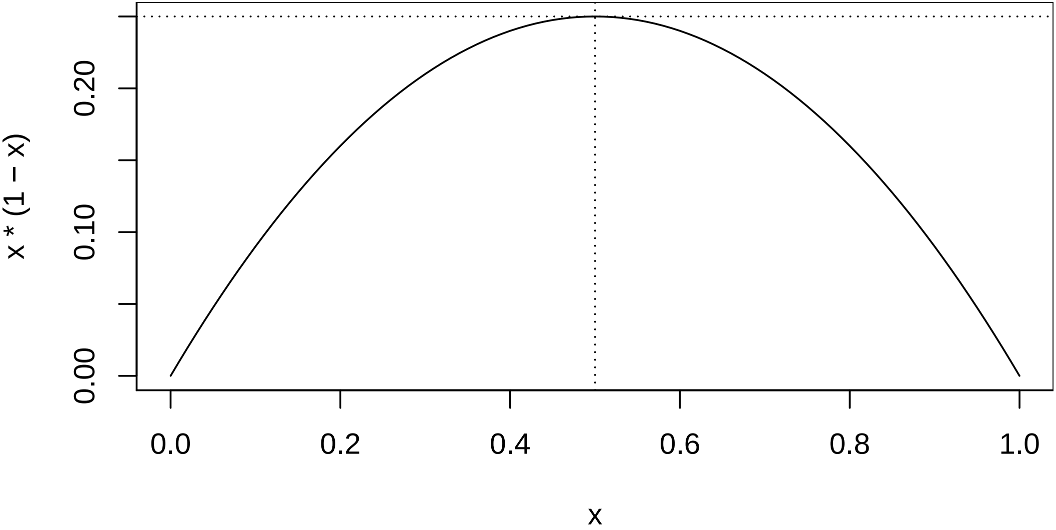
Let's estimate a probability for $X \sim N(0, 1)$ again:

$$\theta = P(X \leq 1.96)$$

Here, $X_i \sim N(0, 1)$, $g(x) = I(x \leq 1.96)$. Recall,

$$g(X_i) \sim \text{Bernoulli}(\theta) \Rightarrow \text{Var}(g(X_i)) = \theta(1 - \theta)$$

Where is the max of $\theta(1 - \theta)$?



95% CI with $w = 0.001$

```
> (targetN <- 4 * qnorm(0.975)^2 * 0.25 / 0.001^2)

[1] 3841459

> gxs <- rnorm(targetN) <= 1.96
> (ci <- t.test(gxs, conf.level = 0.95)$conf.int)

[1] 0.9749 0.9752
attr(,"conf.level")
[1] 0.95

> diff(ci)

[1] 0.0003121
```

More about precision

Often, we can pick from two RVs (X, Y): $\int f(x)\,dx = \mathrm{E}(g(X)) = \mathrm{E}(h(Y))$.

We say $g(X)$ is **more efficient** than $h(Y)$, if $\mathrm{Var}\,(g(X)) < \mathrm{Var}\,(h(Y))$.

Comparing methods for log-Normal estimation $P(X \leq 1.25)$:

```
> c(mean(gW), var(gW)) # based on f(runif(10e6, max = 1.25))
[1] 0.58820 0.03732
> c(mean(hY), var(hY)) # based on exp(rnorm(10e6)) <= 1.25
[1] 0.5883 0.2422
```

We discuss efficiency much more in a few weeks.

Discrete Example

Suppose $X \sim Poisson(2)$. What is the probability that X **is odd**?

Again, we can use an **indicator function**:

$$P(X \text{is odd}) = \mathrm{E}(I(X \mod 2 == 1)$$

```
> x <- rpois(10e6, lambda = 2) %% 2 == 0 # Vectorized computation
> mean(x)

[1] 0.5091

> t.test(x)$conf.int

[1] 0.5088 0.5094
attr(,"conf.level")
[1] 0.95
```

Monte Carlo Integration: Summary

- Write down the integral $\int g(x)\,dx$ you want to solve.
- Find a random variable X **with density** f with the **same domain** as the **bounds of integration** (see `?Distributions`.
- Write down $h(x) = g(x)/f(x)$.
- **Sample from** X and compute $n^{-1}\sum_{i=1}^{n} h(X_i)$
- The **central limit theorem** provides **confidence intervals** (`t.test`) in general, binomial CIs for indicator functions