Data Science for Economists

Lecture 8: Intro to Simulation and Monte Carlo

Alex Marsh
University of North Carolina | ECON 390

Table of contents

Introduction

Agenda

Today we will cover one of the best uses of R: simulations.

- There will be many appliactions and you should be programming along with me.

As well, we will be covering Monte Carlo, a simulation technique popular in statistics and econometrics.

- While we will keep this simple here, more complicated models aren't any more difficult conceptually.

Introduction

(Some important R concepts)

Review

(Some important math and stat concepts)

Probability Theory

- All probabilities must exist in $[0, 1]$ and the sum of all possible outcomes equals 1.

Probability Theory

- All probabilities must exist in $[0, 1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers

Probability Theory

- All probabilities must exist in $[0, 1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers
- RVs are notated with capital letters (e.g. X) whereas "realized" (i.e. nonrandom) outcomes are lowercase (e.g. x)

Probability Theory

- All probabilities must exist in $[0, 1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers
- RVs are notated with capital letters (e.g. X) whereas "realized" (i.e. nonrandom) outcomes are lowercase (e.g. x)
- The distribution of an RV has a probability function $f_X(x)$
  - If X is discrete, $f_X(x)$ is known as a probability mass function (pmf).
  - If X is continuous, $f_X(x)$ is known as a probability density function (pdf).

Probability Theory

- All probabilities must exist in $[0, 1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers
- RVs are notated with capital letters (e.g. X) whereas "realized" (i.e. nonrandom) outcomes are lowercase (e.g. x)
- The distribution of an RV has a probability function $f_X(x)$
  - If X is discrete, $f_X(x)$ is known as a probability mass function (pmf).
  - If X is continuous, $f_X(x)$ is known as a probability density function (pdf).
- The distribution of an RV also has a Cumulative Distribution Function (CDF) $F_X(x)$
  - $F_X(x) = \Pr(X \leq x)$

Probability Theory

- All probabilities must exist in $[0, 1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers
- RVs are notated with capital letters (e.g. X) whereas "realized" (i.e. nonrandom) outcomes are lowercase (e.g. x)
- The distribution of an RV has a probability function $f_X(x)$
  - If X is discrete, $f_X(x)$ is known as a probability mass function (pmf).
  - If X is continuous, $f_X(x)$ is known as a probability density function (pdf).
- The distribution of an RV also has a Cumulative Distribution Function (CDF) $F_X(x)$
  - $F_X(x) = \Pr(X \leq x)$
- If X is continuous, $\frac{d}{dx}F_X(x) = f_X(x)$ and $F_X(x) = \int_{-\infty}^{x} f_X(t)dt$

Probability Theory

- All probabilities must exist in $[0, 1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers
- RVs are notated with capital letters (e.g. X) whereas "realized" (i.e. nonrandom) outcomes are lowercase (e.g. x)
- The distribution of an RV has a probability function $f_X(x)$
  - If X is discrete, $f_X(x)$ is known as a probability mass function (pmf).
  - If X is continuous, $f_X(x)$ is known as a probability density function (pdf).
- The distribution of an RV also has a Cumulative Distribution Function (CDF) $F_X(x)$
  - $F_X(x) = \Pr(X \leq x)$
- If X is continuous, $\frac{d}{dx}F_X(x) = f_X(x)$ and $F_X(x) = \int_{-\infty}^{x} f_X(t)dt$
- The "average value" of an RV is known as it's expectation:
  - Discrete: $E[X] = \sum_i x_i f_X(x_i)$
  - Continuous: $E[X] = \int_{-\infty}^{\infty} x f_X(x)dx$

Probability Theory

- All probabilities must exist in $[0,1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers
- RVs are notated with capital letters (e.g. X) whereas "realized" (i.e. nonrandom) outcomes are lowercase (e.g. x)
- The distribution of an RV has a probability function $f_X(x)$
    - If X is discrete, $f_X(x)$ is known as a probability mass function (pmf).
    - If X is continuous, $f_X(x)$ is known as a probability density function (pdf).
- The distribution of an RV also has a Cumulative Distribution Function (CDF) $F_X(x)$
    - $F_X(x) = \Pr(X \leq x)$
- If X is continuous, $\frac{d}{dx}F_X(x) = f_X(x)$ and $F_X(x) = \int_{-\infty}^{x} f_X(t)dt$
- The "average value" of an RV is known as it's expectation:
    - Discrete: $E[X] = \sum_i x_i f_X(x_i)$
    - Continuous: $E[X] = \int_{-\infty}^{\infty} x f_X(x)dx$
- Higher order expectations (known as *moments*) are defined as:
    - The n^{th} moment is $E[X^n] = \int_{-\infty}^{\infty} x^n f_X(x)dx$

Probability Theory

- All probabilities must exist in $[0, 1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers
- RVs are notated with capital letters (e.g. X) whereas "realized" (i.e. nonrandom) outcomes are lowercase (e.g. x)
- The distribution of an RV has a probability function $f_X(x)$
  - If X is discrete, $f_X(x)$ is known as a probability mass function (pmf).
  - If X is continuous, $f_X(x)$ is known as a probability density function (pdf).
- The distribution of an RV also has a Cumulative Distribution Function (CDF) $F_X(x)$
  - $F_X(x) = \Pr(X \leq x)$
- If X is continuous, $\frac{d}{dx} F_X(x) = f_X(x)$ and $F_X(x) = \int_{-\infty}^{x} f_X(t)dt$
- The "average value" of an RV is known as it's expectation:
  - Discrete: $E[X] = \sum_i x_i f_X(x_i)$
  - Continuous: $E[X] = \int_{-\infty}^{\infty} x f_X(x)dx$
- Higher order expectations (known as *moments*) are defined as:
  - The n^{th} moment is $E[X^n] = \int_{-\infty}^{\infty} x^n f_X(x)dx$
- We also have *central moments* which are defined as $E[(X - E[X])^n]$

Probability Theory

- All probabilities must exist in $[0,1]$ and the sum of all possible outcomes equals 1.
- Random variables map outcomes in a sample space to real numbers
- RVs are notated with capital letters (e.g. X) whereas "realized" (i.e. nonrandom) outcomes are lowercase (e.g. x)
- The distribution of an RV has a probability function $f_X(x)$
  - If X is discrete, $f_X(x)$ is known as a probability mass function (pmf).
  - If X is continuous, $f_X(x)$ is known as a probability density function (pdf).
- The distribution of an RV also has a Cumulative Distribution Function (CDF) $F_X(x)$
  - $F_X(x) = \Pr(X \leq x)$
- If X is continuous, $\frac{d}{dx}F_X(x) = f_X(x)$ and $F_X(x) = \int_{-\infty}^{x} f_X(t)dt$
- The "average value" of an RV is known as it's expectation:
  - Discrete: $E[X] = \sum_i x_i f_X(x_i)$
  - Continuous: $E[X] = \int_{-\infty}^{\infty} x f_X(x)dx$
- Higher order expectations (known as *moments*) are defined as:
  - The n^{th} moment is $E[X^n] = \int_{-\infty}^{\infty} x^n f_X(x)dx$
- We also have *central moments* which are defined as $E[(X - E[X])^n]$
- The variance of a distribution is it's *second central moment*: $\text{Var}(X) = E[(X - E[X])^2] = E[X^2] - E[X]^2$

Probability Theory (Cont.)

- We say RVs X and Y have a *joint distribution* with pmf/pdf $f_{XY}(x, y)$

Probability Theory (Cont.)

- We say RVs X and Y have a *joint distribution* with pmf/pdf $f_{XY}(x, y)$
- We say that X and Y are *independent* iff $f_{XY}(x, y) = f_X(x)f_Y(y)$

Probability Theory (Cont.)

- We say RVs X and Y have a *joint distribution* with pmf/pdf $f_{XY}(x, y)$
- We say that X and Y are *independent* iff $f_{XY}(x, y) = f_X(x)f_Y(y)$
- We say that RVs $(X_1, \ldots, X_n)$ are independent and identically distributed (iid) if they are mutually independent and each X_i comes from the same distribution.
  - This implies all their moments are the same. So $E[X_k^n] = E[X_j^n]$ for all j, k and n .

Properties of Expectations & Variances

- $E[aX + b] = aE[X] + b$

Properties of Expectations & Variances

- $E[aX + b] = aE[X] + b$
- $E[X + Y] = E[X] + E[Y]$

Properties of Expectations & Variances

- $E[aX + b] = aE[X] + b$
- $E[X + Y] = E[X] + E[Y]$
- If X and Y are independent, $E[XY] = E[X]E[Y]$
  - *The converse is not true.*

Properties of Expectations & Variances

- $E[aX + b] = aE[X] + b$
- $E[X + Y] = E[X] + E[Y]$
- If X and Y are independent, $E[XY] = E[X]E[Y]$
    - *The converse is not true.*
- $\mathrm{Var}[aX + b] = a^2\mathrm{Var}[X]$

Properties of Expectations & Variances

- $E[aX+b] = aE[X]+b$
- $E[X+Y] = E[X]+E[Y]$
- If X and Y are independent, $E[XY] = E[X]E[Y]$
  - *The converse is not true.*
- $\mathrm{Var}[aX+b] = a^2\mathrm{Var}[X]$
- $\mathrm{Var}[X+Y] = \mathrm{Var}[X] + \mathrm{Var}[Y] + 2\mathrm{Cov}(X,Y)$
  - $\mathrm{Cov}(X,Y) = E[XY] - E[X]E[Y]$
  - What is $\mathrm{Cov}(X,X)$?

Properties of Expectations & Variances

- $E[aX + b] = aE[X] + b$
- $E[X + Y] = E[X] + E[Y]$
- If X and Y are independent, $E[XY] = E[X]E[Y]$
    - *The converse is not true.*
- $\text{Var}[aX + b] = a^2\text{Var}[X]$
- $\text{Var}[X + Y] = \text{Var}[X] + \text{Var}[Y] + 2\text{Cov}(X, Y)$
    - $\text{Cov}(X, Y) = E[XY] - E[X]E[Y]$
    - What is $\text{Cov}(X, X)$?
- $\text{Cov}(aX + b, cY + d) = ac\text{Cov}(X, Y)$

Properties of Expectations & Variances

- $E[aX+b] = aE[X]+b$
- $E[X+Y] = E[X]+E[Y]$
- If X and Y are independent, $E[XY] = E[X]E[Y]$
    - *The converse is not true.*
- $\mathrm{Var}[aX+b] = a^2\mathrm{Var}[X]$
- $\mathrm{Var}[X+Y] = \mathrm{Var}[X]+\mathrm{Var}[Y]+2\mathrm{Cov}(X,Y)$
    - $\mathrm{Cov}(X,Y) = E[XY]-E[X]E[Y]$
    - What is $\mathrm{Cov}(X,X)$?
- $\mathrm{Cov}(aX+b, cY+d) = ac\mathrm{Cov}(X,Y)$
- If X and Y are independent, $\mathrm{Cov}(X,Y) = 0$

Properties of Expectations & Variances

- $E[aX + b] = aE[X] + b$
- $E[X + Y] = E[X] + E[Y]$
- If X and Y are independent, $E[XY] = E[X]E[Y]$
  - *The converse is not true.*
- $\mathrm{Var}[aX + b] = a^2\mathrm{Var}[X]$
- $\mathrm{Var}[X + Y] = \mathrm{Var}[X] + \mathrm{Var}[Y] + 2\mathrm{Cov}(X, Y)$
  - $\mathrm{Cov}(X, Y) = E[XY] - E[X]E[Y]$
  - What is $\mathrm{Cov}(X, X)$?
- $\mathrm{Cov}(aX + b, cY + d) = ac\mathrm{Cov}(X, Y)$
- If X and Y are independent, $\mathrm{Cov}(X, Y) = 0$
- $\mathrm{Cov}(X + Y, V + W) = \mathrm{Cov}(X, V) + \mathrm{Cov}(X, W) + \mathrm{Cov}(Y, V) + \mathrm{Cov}(Y, W)$

Properties of Expectations & Variances

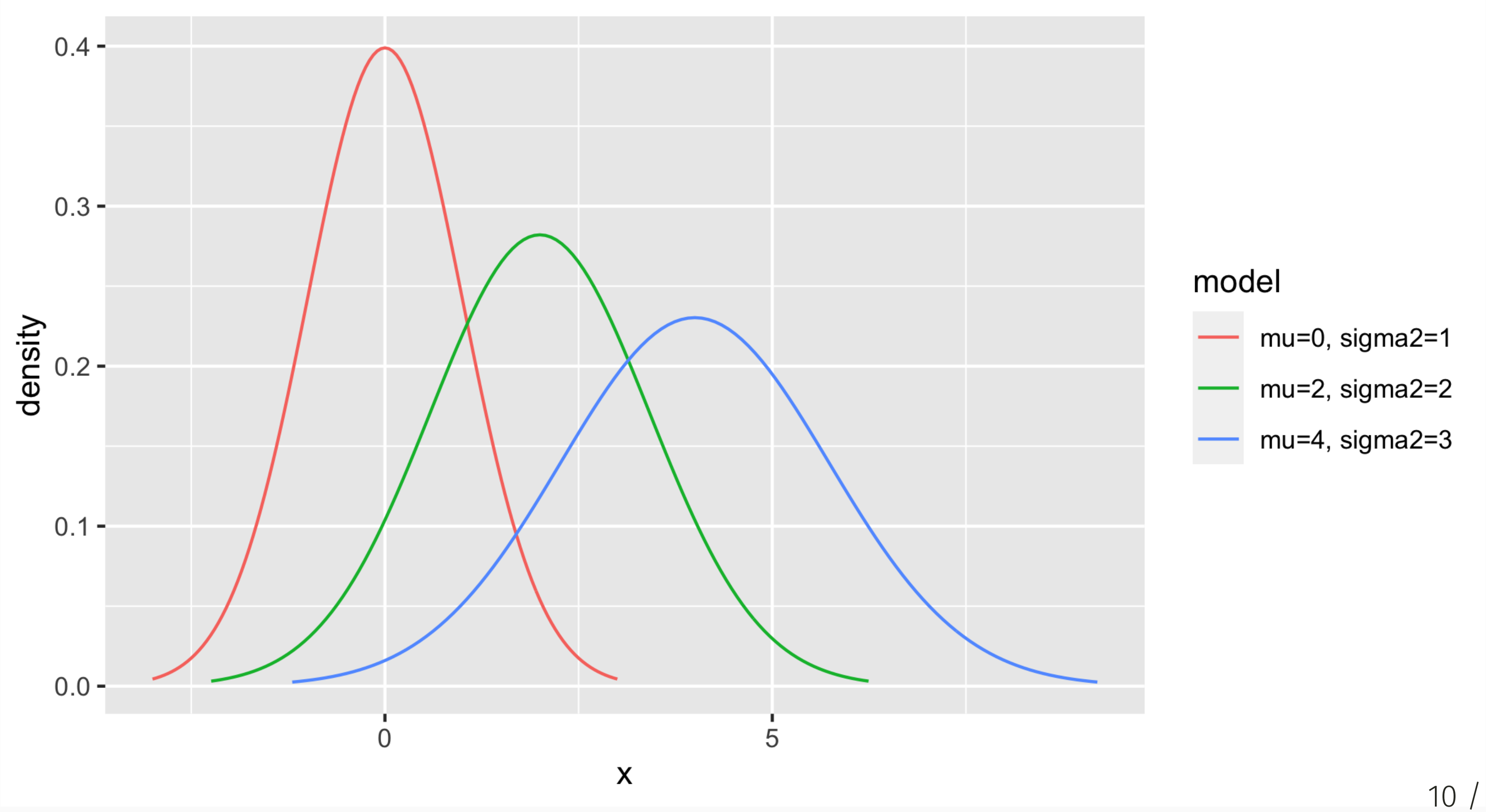
- $E[aX + b] = aE[X] + b$
- $E[X + Y] = E[X] + E[Y]$
- If X and Y are independent, $E[XY] = E[X]E[Y]$
  - *The converse is not true.*
- $\mathrm{Var}[aX + b] = a^2\mathrm{Var}[X]$
- $\mathrm{Var}[X + Y] = \mathrm{Var}[X] + \mathrm{Var}[Y] + 2\mathrm{Cov}(X, Y)$
  - $\mathrm{Cov}(X, Y) = E[XY] - E[X]E[Y]$
  - What is $\mathrm{Cov}(X, X)$?
- $\mathrm{Cov}(aX + b, cY + d) = ac\mathrm{Cov}(X, Y)$
- If X and Y are independent, $\mathrm{Cov}(X, Y) = 0$
- $\mathrm{Cov}(X + Y, V + W) = \mathrm{Cov}(X, V) + \mathrm{Cov}(X, W) + \mathrm{Cov}(Y, V) + \mathrm{Cov}(Y, W)$
- If $(X_1, \ldots, X_n)$ are independent $\mathrm{Var}(\sum_i a_i X_i) = \sum_i a_i^2 \mathrm{Var}(X_i)$
  - What if $(X_1, \ldots, X_n)$ are iid?

Properties of Expectations & Variances

- $E[aX + b] = aE[X] + b$
- $E[X + Y] = E[X] + E[Y]$
- If X and Y are independent, $E[XY] = E[X]E[Y]$
  - *The converse is not true.*
- $\mathrm{Var}[aX + b] = a^2\mathrm{Var}[X]$
- $\mathrm{Var}[X + Y] = \mathrm{Var}[X] + \mathrm{Var}[Y] + 2\mathrm{Cov}(X, Y)$
  - $\mathrm{Cov}(X, Y) = E[XY] - E[X]E[Y]$
  - What is $\mathrm{Cov}(X, X)$?
- $\mathrm{Cov}(aX + b, cY + d) = ac\mathrm{Cov}(X, Y)$
- If X and Y are independent, $\mathrm{Cov}(X, Y) = 0$
- $\mathrm{Cov}(X + Y, V + W) = \mathrm{Cov}(X, V) + \mathrm{Cov}(X, W) + \mathrm{Cov}(Y, V) + \mathrm{Cov}(Y, W)$
- If $(X_1, \ldots, X_n)$ are independent $\mathrm{Var}(\sum_i a_i X_i) = \sum_i a_i^2 \mathrm{Var}(X_i)$
  - What if $(X_1, \ldots, X_n)$ are iid?
- If $(X_1, \ldots, X_n)$ are *not* independent: $\mathrm{Var}(\sum_i a_i X_i) = \sum_j \sum_i a_i a_j \mathrm{Cov}(X_i, X_j)$

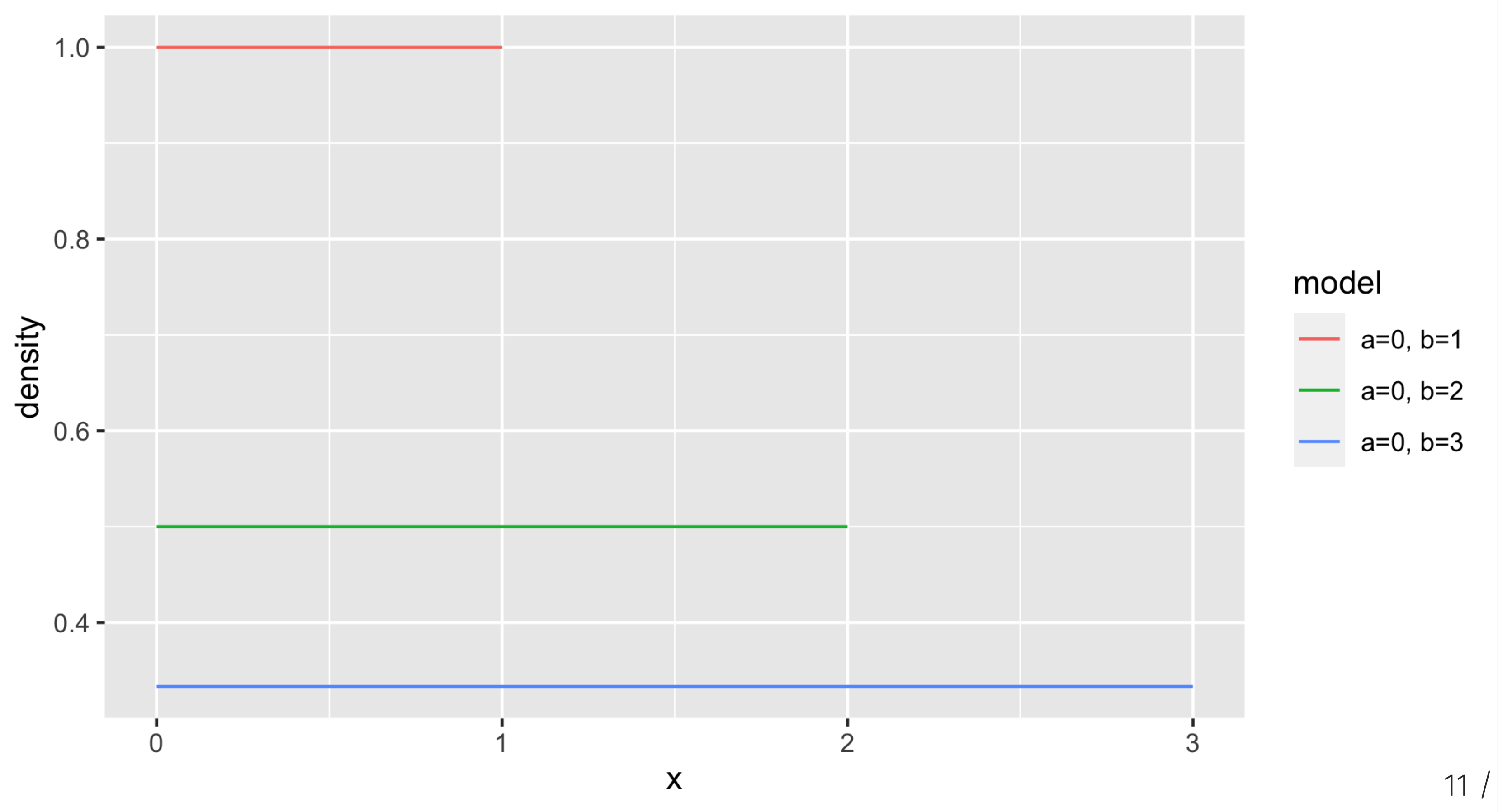
Review of Common Distributions

- The Normal Distribution
  - Two parameters: μ and σ^2; notated $N(\mu, \sigma^2)$
  - Mean and variance: mean is μ and variance is σ^2



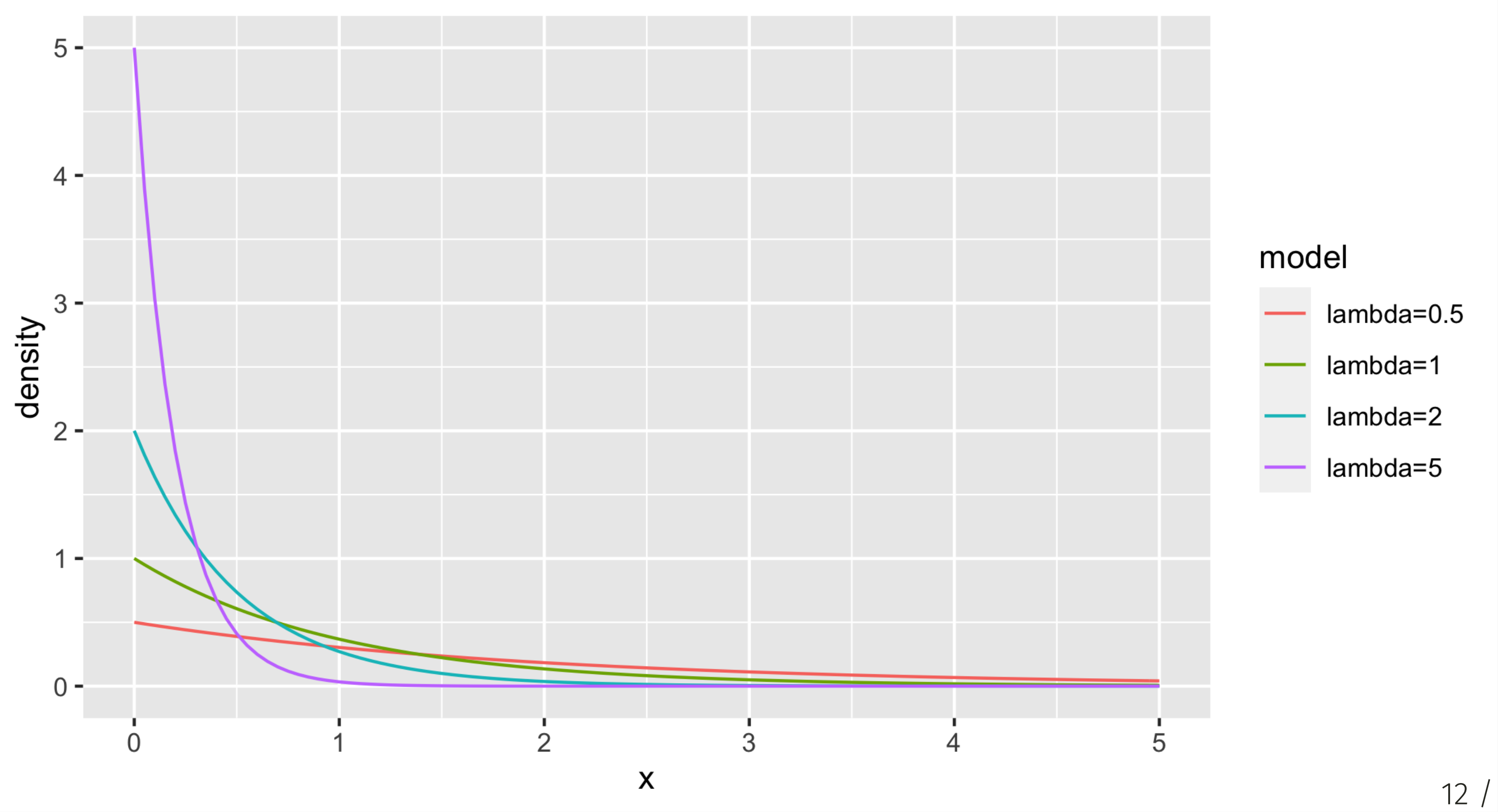
Review of Common Distributions

- The (Continuous) Uniform Distribution
  - Two parameters: a and b where $a < b$; notated $\textbf{Uniform}(a, b)$
  - Mean and variance: mean is $\frac{a+b}{2}$ and variance is $\frac{(b-a)^2}{12}$



Review of Common Distributions

- The Exponential Distribution
  - One parameter: λ; notated $\mathbf{Exp}(\lambda)$
  - Mean and variance: mean is $\frac{1}{\lambda}$ and variance is $\frac{1}{\lambda^2}$



What is simulation?

Simulation

While it sounds fancy, simulation is nothing but running "data generating processes" in `R` with randomly generated data.

- What is a data generating process (DGP)?

Simulation

While it sounds fancy, simulation is nothing but running "data generating processes" in `R` with randomly generated data.

- What is a data generating process (DGP)?

> "...a data generating process is a process in the real world that "generates" the data one is interested in."

Simulation

While it sounds fancy, simulation is nothing but running "data generating processes" in `R` with randomly generated data.

- What is a data generating process (DGP)?

> "...a data generating process is a process in the real world that "generates" the data one is interested in."

For actual research questions, we can never know the true DGP. However, we can see if our techniques work when the true data generating process is specified correctly.

Examples of a DGP

- Let's start with a basic example from ECON 101.

Examples of a DGP

- Let's start with a basic example from ECON 101.
- Suppose that we have a demand equation $\boldsymbol{q^d(p) = 10 - 2p}$.

Examples of a DGP

- Let's start with a basic example from ECON 101.
- Suppose that we have a demand equation $q^d(p) = 10 - 2p$.
- As well, suppose we have a supply equation $q^s(p) = 3p$

Examples of a DGP

- Let's start with a basic example from ECON 101.
- Suppose that we have a demand equation $q^d(p) = 10 - 2p$.
- As well, suppose we have a supply equation $q^s(p) = 3p$
- In equilibrium, $q^s = q^d = q$, so $10 - 2p = 3p$, or $p^* = 2$ and $q^* = 5$

Examples of a DGP

- Let's start with a basic example from ECON 101.
- Suppose that we have a demand equation $q^d(p) = 10 - 2p$.
- As well, suppose we have a supply equation $q^s(p) = 3p$
- In equilibrium, $q^s = q^d = q$, so $10 - 2p = 3p$, or $p^* = 2$ and $q^* = 5$
- What is the DGP and what are the data?

Examples of a DGP

- Let's start with a basic example from ECON 101.
- Suppose that we have a demand equation $q^d(p) = 10 - 2p$.
- As well, suppose we have a supply equation $q^s(p) = 3p$
- In equilibrium, $q^s = q^d = q$, so $10 - 2p = 3p$, or $p^* = 2$ and $q^* = 5$
- What is the DGP and what are the data?
- The DGP is the demand and supply equations.
    - Sometimes called the structural equations.

Examples of a DGP

- Let's start with a basic example from ECON 101.
- Suppose that we have a demand equation $q^d(p) = 10 - 2p$.
- As well, suppose we have a supply equation $q^s(p) = 3p$
- In equilibrium, $q^s = q^d = q$, so $10 - 2p = 3p$, or $p^* = 2$ and $q^* = 5$
- What is the DGP and what are the data?
- The DGP is the demand and supply equations.
  - Sometimes called the structural equations.
- The data are the equilibrium outcomes p^* and q^*.

Examples of a DGP

- Let's start with a basic example from ECON 101.
- Suppose that we have a demand equation $q^d(p) = 10 - 2p$.
- As well, suppose we have a supply equation $q^s(p) = 3p$
- In equilibrium, $q^s = q^d = q$, so $10 - 2p = 3p$, or $p^* = 2$ and $q^* = 5$
- What is the DGP and what are the data?
- The DGP is the demand and supply equations.
  - Sometimes called the structural equations.
- The data are the equilibrium outcomes p^* and q^*.
- While this example is very simple (almost too simple), the conceptual idea of what a DGP is stays the same.
  - DGP is your economic/theoretical model and the data are the outcomes of your model.

Our First Simulation

Let's start with a basic example: flipping a coin.

Our First Simulation

Let's start with a basic example: flipping a coin.

- Virtually every distribution can be simulated from a $\mathbf{Uniform}(0, 1)$ distribution.

Our First Simulation

Let's start with a basic example: flipping a coin.

- Virtually every distribution can be simulated from a $\textbf{Uniform}(0, 1)$ distribution.
- With some creative thinking, we can use "random" draws from a $\textbf{Uniform}(0, 1)$ distribution to simulate flipping a coin.

Our First Simulation

Let's start with a basic example: flipping a coin.

- Virtually every distribution can be simulated from a $\textbf{Uniform}(0, 1)$ distribution.
- With some creative thinking, we can use "random" draws from a $\textbf{Uniform}(0, 1)$ distribution to simulate flipping a coin.
  - Why $\textbf{Uniform}(0, 1)$?

Our First Simulation

Let's start with a basic example: flipping a coin.

- Virtually every distribution can be simulated from a $\mathbf{Uniform}(0, 1)$ distribution.
- With some creative thinking, we can use "random" draws from a $\mathbf{Uniform}(0, 1)$ distribution to simulate flipping a coin.
  - Why $\mathbf{Uniform}(0, 1)$?

```
set.seed(123)                         #reproducibility
one_draw = runif(1,min=0,max=1) #runif(1) also works
```

Now that we have the draw, how can we simulate flipping a (fair) coin?

Our First Simulation

Let's start with a basic example: flipping a coin.

- Virtually every distribution can be simulated from a $\textbf{Uniform}(0,1)$ distribution.
- With some creative thinking, we can use "random" draws from a $\textbf{Uniform}(0,1)$ distribution to simulate flipping a coin.
  - Why $\textbf{Uniform}(0,1)$?

```
set.seed(123)                     #reproducibility
one_draw = runif(1,min=0,max=1) #runif(1) also works
```

Now that we have the draw, how can we simulate flipping a (fair) coin?

Say, if the draw is below 0.5, it is heads, otherwise it is tail.

Our First Simulation

Let's start with a basic example: flipping a coin.

- Virtually every distribution can be simulated from a $\text{Uniform}(0,1)$ distribution.
- With some creative thinking, we can use "random" draws from a $\text{Uniform}(0,1)$ distribution to simulate flipping a coin.
  - Why $\text{Uniform}(0,1)$?

```
set.seed(123)                    #reproducibility
one_draw = runif(1,min=0,max=1) #runif(1) also works
```

Now that we have the draw, how can we simulate flipping a (fair) coin?

Say, if the draw is below 0.5, it is heads, otherwise it is tail.

```
one_draw
```

```
## [1] 0.2875775
```

```
ifelse(one_draw<0.5,"heads","tails")
```

```
## [1] "heads"
```

Our First Simulation

- What if we didn't want it to be fair? Say heads with 0.25 probability?
  - Hint remember the code we used for the fair coin.

```
one_draw
ifelse(one_draw<0.5,"heads","tails")
```

Our First Simulation

- What if we didn't want it to be fair? Say heads with 0.25 probability?
    - Hint remember the code we used for the fair coin.

```
one_draw
ifelse(one_draw<0.5,"heads","tails")
```

- If we change the cut-off (i.e. 0.5), we can adjust the probability mass accordingly.

Our First Simulation

- What if we didn't want it to be fair? Say heads with 0.25 probability?
  - Hint remember the code we used for the fair coin.

```
one_draw
ifelse(one_draw<0.5,"heads","tails")
```

- If we change the cut-off (i.e. 0.5), we can adjust the probability mass accordingly.

```
one_draw
```

```
## [1] 0.2875775
```

```
ifelse(one_draw<0.25,"heads","tails")
```

```
## [1] "tails"
```

Our Next Simulation

What if we wanted to simulate rolling a six-sided die? How would we simulate this?

Our Next Simulation

What if we wanted to simulate rolling a six-sided die? How would we simulate this?

- Hint: flipping a coin is like rolling a two-sided die.

Our Next Simulation

What if we wanted to simulate rolling a six-sided die? How would we simulate this?

- Hint: flipping a coin is like rolling a two-sided die.

```
another_draw = runif(1)
ubs          = seq(1/6,1,1/6)
lbs          = seq(0,1-1/6,1/6)

which(ubs >= another_draw & lbs < another_draw)
```

```
## [1] 5
```

Our Next Simulation

What if we wanted to simulate rolling a six-sided die? How would we simulate this?

- Hint: flipping a coin is like rolling a two-sided die.

```
another_draw = runif(1)
ubs          = seq(1/6,1,1/6)
lbs          = seq(0,1-1/6,1/6)

which(ubs >= another_draw & lbs < another_draw)
```

```
## [1] 5
```

Notice the structure of this code. This could be easily generalized.

Our Next Simulation

What if we wanted to simulate rolling a six-sided die? How would we simulate this?

- Hint: flipping a coin is like rolling a two-sided die.

```
another_draw = runif(1)
ubs          = seq(1/6,1,1/6)
lbs          = seq(0,1-1/6,1/6)

which(ubs >= another_draw & lbs < another_draw)
```

```
## [1] 5
```

Notice the structure of this code. This could be easily generalized.

Let's write a function generalizing this to an n-sided die.

Rolling Die Function

Rolling Die Function

```
roll_die = function(n){
  draw = runif(1)
  ubs  = seq(1/n,1,1/n)
  lbs  = seq(0,1-1/n,1/n)
  which(ubs >= draw & lbs < draw)
}
```

Rolling Die Function

```r
roll_die = function(n){
  draw = runif(1)
  ubs  = seq(1/n,1,1/n)
  lbs  = seq(0,1-1/n,1/n)
  which(ubs >= draw & lbs < draw)
}
```

```r
roll_die(6)
```

```
## [1] 3
```

```r
roll_die(20)
```

```
## [1] 18
```

```r
roll_die(12)
```

```
## [1] 12
```

Rolling Die Function

```r
roll_die = function(n){
  draw = runif(1)
  ubs  = seq(1/n,1,1/n)
  lbs  = seq(0,1-1/n,1/n)
  which(ubs >= draw & lbs < draw)
}
```

```r
roll_die(6)
```

```
## [1] 3
```

```r
roll_die(20)
```

```
## [1] 18
```

```r
roll_die(12)
```

```
## [1] 12
```

What about multiple rolls?

Rolling Dice Function: Multiple Rolls

```
roll_dice = function(k,n){
  draws  = runif(k)         #draw simulations
  output = rep(0,k)         #initialize output vector
  ubs    = seq(1/n,1,1/n)   #upper-bounds for roll intervals
  lbs    = seq(0,1-1/n,1/n) #lower-bounds for roll intervals

  for(i in 1:n){
    # if draw is in the ith interval, the roll was i
    output[draws>lbs[i] & draws<= ubs[i]] = i
  }
  output[draws<= lbs[1]] = 1 #weird edge case
  output
}
```

Rolling Dice Function: Multiple Rolls

```
roll_dice = function(k,n){
  draws  = runif(k)         #draw simulations
  output = rep(0,k)         #initialize output vector
  ubs    = seq(1/n,1,1/n)   #upper-bounds for roll intervals
  lbs    = seq(0,1-1/n,1/n) #lower-bounds for roll intervals

  for(i in 1:n){
    # if draw is in the ith interval, the roll was i
    output[draws>lbs[i] & draws<= ubs[i]] = i
  }
  output[draws<= lbs[1]] = 1 #weird edge case
  output
}
```

```
roll_dice(8,6)
```

```
## [1] 1 4 6 4 3 6 3 5
```

```
roll_dice(8,20)
```

```
## [1] 12  3 18  5  1  7 20 18
```

Testing Our Function

Does our function actually roll a fair n-sided dice?

Let's look at a histogram of many rolls.

Testing Our Function

Does our function actually roll a fair n-sided dice?

Let's look at a histogram of many rolls.

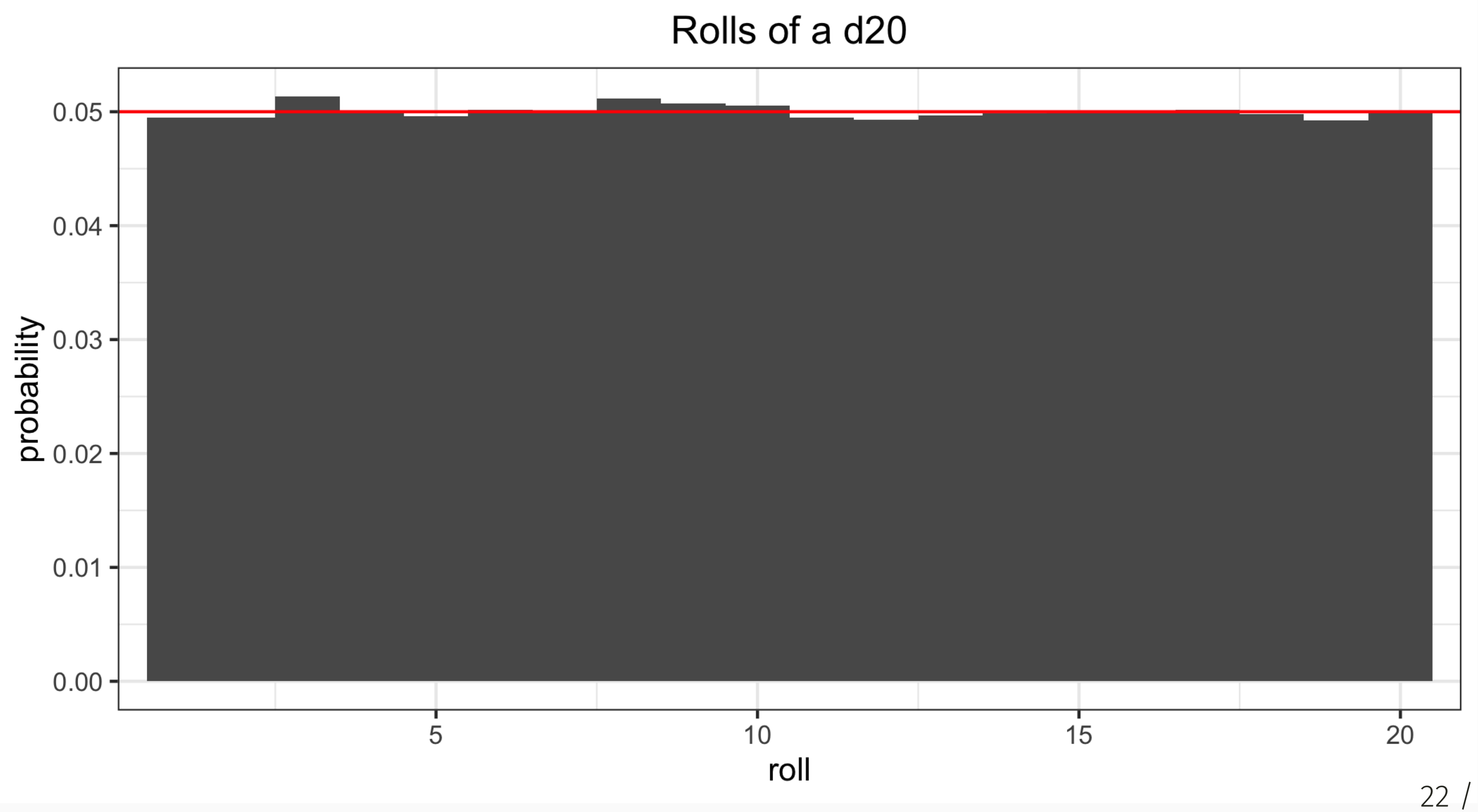
```
Nrolls            = 100000                  #set number of rolls
Nsides            = 20                      #set number of sides
rolls             = roll_dice(Nrolls,Nsides) #simulate rolls
roll_data         = data.frame(x=rolls)      #store rolls in data.table

# estimate probabilities
roll_probs        = sapply(1:Nsides,function(x){mean(roll_data[,1]==x)})
names(roll_probs) = 1:Nsides #set names for each estimated probability
roll_probs
```

```
##       1       2       3       4       5       6       7       8       9      10
## 0.04946 0.04951 0.05134 0.05004 0.04958 0.05019 0.05005 0.05118 0.05071 0.05055
##      11      12      13      14      15      16      17      18      19      20
## 0.04946 0.04928 0.04969 0.04985 0.04989 0.04992 0.05015 0.04978 0.04927 0.05010
```

```
roll_plot = ggplot(roll_data) + geom_histogram(aes(x=x, y=..density..),bins=Nsides)
```

Testing Our Function

```
roll_plot + coord_cartesian(xlim=c(min(rolls),max(rolls))) +
  ggtitle("Rolls of a d20") + theme_bw() + ylab("probability") + xlab("roll") +
  theme(plot.title = element_text(hjust = 0.5)) + geom_hline(yintercept=1/20, col = 'r
```



Simulating a Regression

Suppose we have the following regression equation that is the *true* DGP

$$y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \varepsilon_i$$

where $x_{1i} \sim N(2,1)$, $x_{2i} \sim \text{Exp}(2)$, $\varepsilon_i \sim U(-1,1)$, $\beta_0 = 2$, $\beta_1 = -5$, $\beta_2 = 4$.

```
Nsim    = 1000            #set number of simulations
beta0   = 2               #set intercept
beta1   = -5              #set coefficient for x1
beta2   = 4               #set coefficient for x2
x1      = rnorm(Nsim,2)   #draw x1
x2      = rexp(Nsim, 2)   #draw x2
y       = beta0 + beta1*x1 + beta2*x2 + runif(Nsim,-1) #form y
reg_fit = lm(y ~ x1 + x2) #run regression
```

Simulating a Regression

```
summary(reg_fit)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.10273 -0.47979 -0.01751  0.47996  1.08616
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.11034    0.04569   46.19   <2e-16 ***
## x1          -5.06384    0.01921 -263.63   <2e-16 ***
## x2           4.05518    0.03791  106.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5755 on 997 degrees of freedom
## Multiple R-squared:  0.9878,    Adjusted R-squared:  0.9878
## F-statistic: 4.029e+04 on 2 and 997 DF,  p-value: < 2.2e-16
```

What Else Can We Use Simulation For?

- As seen above, simulation can be used to… well, simulate a DGP.
- We can also use simulation to calculate a complicated probability.
- How can we do this?

What Else Can We Use Simulation For?

- As seen above, simulation can be used to... well, simulate a DGP.
- We can also use simulation to calculate a complicated probability.
- How can we do this?
- Well, we actually already have!

What Else Can We Use Simulation For?

- As seen above, simulation can be used to... well, simulate a DGP.
- We can also use simulation to calculate a complicated probability.
- How can we do this?
- Well, we actually already have!
- The (naive) definition of a probability is the number of ways an event can occur divided by the number of possible outcomes.

What Else Can We Use Simulation For?

- As seen above, simulation can be used to... well, simulate a DGP.
- We can also use simulation to calculate a complicated probability.
- How can we do this?
- Well, we actually already have!
- The (naive) definition of a probability is the number of ways an event can occur divided by the number of possible outcomes.

```
Nrolls      = 200                        #set number of rolls
Nsides      = 20                         #set number of sides
rolls       = roll_dice(Nrolls,Nsides)   #simulate rolls
sim_prob    = mean(rolls==20)            #calculate simulated probability
theory_prob = 1/20                       #store theoretical probability
c(sim_prob,theory_prob)
```

```
## [1] 0.04 0.05
```

Estimating Probabilities

Estimating Probabilities

- The last example was pretty simple.
    - You probably already knew what the probability was!
- Let's try a slightly more complicated example.

Estimating Probabilities

- The last example was pretty simple.
  - You probably already knew what the probability was!
- Let's try a slightly more complicated example.
- Suppose $v_i \sim N(1, 4)$ and we want to know $\mathbf{Pr}(-1 \leq v_i \leq 3)$.

Estimating Probabilities

- The last example was pretty simple.
  - You probably already knew what the probability was!
- Let's try a slightly more complicated example.
- Suppose $v_i \sim N(1, 4)$ and we want to know $\mathbf{Pr}(-1 \leq v_i \leq 3)$.
- While this probability is simple to calculate if you've taken a probability theory class, maybe you haven't but need to know it. How could we use simulation?

Estimating Probabilities

- The last example was pretty simple.
  - You probably already knew what the probability was!
- Let's try a slightly more complicated example.
- Suppose $v_i \sim N(1,4)$ and we want to know $\mathbf{Pr}(-1 \leq v_i \leq 3)$.
- While this probability is simple to calculate if you've taken a probability theory class, maybe you haven't but need to know it. How could we use simulation?

```
Ndraw       = 500                             #set number of draws (sims)
ub          = 3                               #set upper bound of the interval
lb          = -1                              #set lower bound of the interval
mu          = 1                               #set mean of the dist
sigma       = sqrt(4)                         #set st dev of the dist
vi          = rnorm(Ndraw,mu,sigma)           #draw simulations
sim_prob    = mean(vi < ub &  vi > lb)        #estimate sim prob
theory_prob = pnorm(ub,mu,sigma)-pnorm(lb,mu,sigma) #calc theory prob

c(sim_prob,theory_prob)
```

```
## [1] 0.6660000 0.6826895
```

Estimating Probabilities

- Let's try one more example that is slightly harder
- Assume that v_i and w_i are distributed *jointly* normal.
    - $\mu_v = 2$, $\mu_w = 3$, $\sigma_v^2 = 4$, $\sigma_w^2 = 1$, and $\sigma_{vw} = 0.5$
- What is $\Pr(-1 \leq v_i \leq 3 \;\&\; -1 \leq w_i \leq 3)$

```r
library(mvtnorm)
Nsim       = 1000                             #set number of sims
Mu         = c(2,3)                           #store means
Sigma      = matrix(c(4,0.5,0.5,1),ncol=2) #store covariance matrix
vws        = rmvnorm(Nsim,Mu,Sigma)           #draw vs and ws
test1      = vws[,1] >= -1 & vws[,1] <= 3   #test if vs are in test range
names(vws) = c("v_i","w_i")                   #set names of draws
test2      = vws[,2] >= -1 & vws[,2] <= 3   #test if ws are in test range
sim_prob   = mean(test1 & test2)              #calc simulated probability
sim_prob
```

```
## [1] 0.343
```

Reflection: Pros and Cons

- What are some of the pros of this approach?

Reflection: Pros and Cons

- What are some of the pros of this approach?
    - Very simple to calculate these probabilities.
- What are some of the cons of this approach?

Reflection: Pros and Cons

- What are some of the pros of this approach?
    - Very simple to calculate these probabilities.
- What are some of the cons of this approach?
    - Must be able to draw from these distributions.
    - This can be done as long as there as a way to evaluate the CDF (ask me if you're curious).
    - Can be computationally expensive as the complexity of the problem increases.
    - Does not have the best empirical properties if used in some optimization problems.

An Economic Example: Product Selection

- Suppose that a firm (labeled i) can release four products $j = \{1, 2, 3, 4\}$.
  - Suppose also that there is an option 0, to not release anything.

An Economic Example: Product Selection

- Suppose that a firm (labeled i) can release four products $j = \{1, 2, 3, 4\}$.
  - Suppose also that there is an option 0, to not release anything.
- The profit of releasing each product has the following compenets:
  - Revenue: r_{ij}
  - A non-negative cost shock with a known mean: c_{ij}
  - Some unobserved part of profit: ε_{ij}

An Economic Example: Product Selection

- Suppose that a firm (labeled i) can release four products $j = \{1, 2, 3, 4\}$.
  - Suppose also that there is an option 0, to not release anything.
- The profit of releasing each product has the following compenets:
  - Revenue: r_{ij}
  - A non-negative cost shock with a known mean: c_{ij}
  - Some unobserved part of profit: ε_{ij}
- So profit for each product j is $\pi_{ij} = r_{ij} - c_{ij} + \varepsilon_{ij}$
  - The profit from not releasing anything is $\pi_{i0} = \varepsilon_{i0}$

An Economic Example: Product Selection

- Suppose that a firm (labeled i) can release four products $j = \{1, 2, 3, 4\}$.
  - Suppose also that there is an option 0, to not release anything.
- The profit of releasing each product has the following compenets:
  - Revenue: r_{ij}
  - A non-negative cost shock with a known mean: c_{ij}
  - Some unobserved part of profit: ε_{ij}
- So profit for each product j is $\pi_{ij} = r_{ij} - c_{ij} + \varepsilon_{ij}$
  - The profit from not releasing anything is $\pi_{i0} = \varepsilon_{i0}$
- Suppose we observe revenues but *do not* observe costs.

An Economic Example: Product Selection

- Suppose that a firm (labeled i) can release four products $j = \{1, 2, 3, 4\}$.
  - Suppose also that there is an option 0, to not release anything.
- The profit of releasing each product has the following compenets:
  - Revenue: r_{ij}
  - A non-negative cost shock with a known mean: c_{ij}
  - Some unobserved part of profit: ε_{ij}
- So profit for each product j is $\pi_{ij} = r_{ij} - c_{ij} + \varepsilon_{ij}$
  - The profit from not releasing anything is $\pi_{i0} = \varepsilon_{i0}$
- Suppose we observe revenues but *do not* observe costs.
- A firm releases product j if $\pi_{ij} > \pi_{ik}$ for all $k \neq j$.

An Economic Example: Product Selection

- Suppose that a firm (labeled i) can release four products $j = \{1, 2, 3, 4\}$.
  - Suppose also that there is an option 0, to not release anything.
- The profit of releasing each product has the following compenets:
  - Revenue: r_{ij}
  - A non-negative cost shock with a known mean: c_{ij}
  - Some unobserved part of profit: ε_{ij}
- So profit for each product j is $\pi_{ij} = r_{ij} - c_{ij} + \varepsilon_{ij}$
  - The profit from not releasing anything is $\pi_{i0} = \varepsilon_{i0}$
- Suppose we observe revenues but *do not* observe costs.
- A firm releases product j if $\pi_{ij} > \pi_{ik}$ for all $k \neq j$.
- Lastly, suppose
  - $\varepsilon_{ij} \sim \text{Gumble}$
  - $c_{ij} \sim \text{Exp}(\lambda)$ with $\lambda = 1/2$
  - $r_{ij} = j$

An Economic Example: Product Selection

- Suppose that a firm (labeled i) can release four products $j = \{1, 2, 3, 4\}$.
  - Suppose also that there is an option 0, to not release anything.
- The profit of releasing each product has the following compenets:
  - Revenue: r_{ij}
  - A non-negative cost shock with a known mean: c_{ij}
  - Some unobserved part of profit: ε_{ij}
- So profit for each product j is $\pi_{ij} = r_{ij} - c_{ij} + \varepsilon_{ij}$
  - The profit from not releasing anything is $\pi_{i0} = \varepsilon_{i0}$
- Suppose we observe revenues but *do not* observe costs.
- A firm releases product j if $\pi_{ij} > \pi_{ik}$ for all $k \neq j$.
- Lastly, suppose
  - $\varepsilon_{ij} \sim \text{Gumble}$
  - $c_{ij} \sim \text{Exp}(\lambda)$ with $\lambda = 1/2$
  - $r_{ij} = j$
- What is the probability that firm i releases product j?

An Economic Example: Product Selection

- Suppose that a firm (labeled i) can release four products $j = \{1, 2, 3, 4\}$.
  - Suppose also that there is an option 0, to not release anything.
- The profit of releasing each product has the following compenets:
  - Revenue: r_{ij}
  - A non-negative cost shock with a known mean: c_{ij}
  - Some unobserved part of profit: ε_{ij}
- So profit for each product j is $\pi_{ij} = r_{ij} - c_{ij} + \varepsilon_{ij}$
  - The profit from not releasing anything is $\pi_{i0} = \varepsilon_{i0}$
- Suppose we observe revenues but *do not* observe costs.
- A firm releases product j if $\pi_{ij} > \pi_{ik}$ for all $k \neq j$.
- Lastly, suppose
  - $\varepsilon_{ij} \sim \text{Gumble}$
  - $c_{ij} \sim \text{Exp}(\lambda)$ with $\lambda = 1/2$
  - $r_{ij} = j$
- What is the probability that firm i releases product j?
- It is $\Pr(\pi_{ij} > \pi_{ik} \text{ for all } k \neq j)$

An Economic Example: Product Selection

- $\Pr(\pi_{ij} > \pi_{ik} \text{ for all } k \neq j) = \Pr(j - c_{ij} + \varepsilon_{ij} > k - c_{ik} + \varepsilon_{ik} \text{ for all } k \neq j)$
- We need to know the distribution of $-c_{ij} + \varepsilon_{ij}$ to know the probability j is selected
  - This is hard!

An Economic Example: Product Selection

- $\Pr(\pi_{ij} > \pi_{ik} \text{ for all } k \neq j) = \Pr(j - c_{ij} + \varepsilon_{ij} > k - c_{ik} + \varepsilon_{ik} \text{ for all } k \neq j)$
- We need to know the distribution of $-c_{ij} + \varepsilon_{ij}$ to know the probability j is selected
  - This is hard!
- But if we actually observed c_{ij}, we would know the probability j is selected

$$\frac{e^{r_j - c_{ij}}}{1 + \sum_{k=1}^{4} e^{r_k - c_{ik}}}$$

  - Don't worry about where this comes from.
  - The 1 comes from the option to not release anything.

An Economic Example: Product Selection

- $\Pr(\pi_{ij} > \pi_{ik} \text{ for all } k \neq j) = \Pr(j - c_{ij} + \varepsilon_{ij} > k - c_{ik} + \varepsilon_{ik} \text{ for all } k \neq j)$
- We need to know the distribution of $-c_{ij} + \varepsilon_{ij}$ to know the probability j is selected
  - This is hard!
- But if we actually observed c_{ij}, we would know the probability j is selected

$$\frac{e^{r_j - c_{ij}}}{1 + \sum_{k=1}^{4} e^{r_k - c_{ik}}}$$

  - Don't worry about where this comes from.
  - The 1 comes from the option to not release anything.
- Since we assumed $c_{ij} \sim \mathrm{Exp}(1/2)$, we can simulate this probability.

An Economic Example: Product Selection

- $\Pr(\pi_{ij} > \pi_{ik} \text{ for all } k \neq j) = \Pr(j - c_{ij} + \varepsilon_{ij} > k - c_{ik} + \varepsilon_{ik} \text{ for all } k \neq j)$
- We need to know the distribution of $-c_{ij} + \varepsilon_{ij}$ to know the probability j is selected
  - This is hard!
- But if we actually observed c_{ij}, we would know the probability j is selected

$$\frac{e^{r_j - c_{ij}}}{1 + \sum_{k=1}^{4} e^{r_k - c_{ik}}}$$

  - Don't worry about where this comes from.
  - The 1 comes from the option to not release anything.
- Since we assumed $c_{ij} \sim \text{Exp}(1/2)$, we can simulate this probability.
- We draw a bunch of c_{ij}'s from $\text{Exp}(1/2)$, evaluate the expression, and then take the mean.

An Economic Example: Product Selection

```
Nsim     = 100000
lambda   = 1/2
cij      = rexp(4*Nsim,rate=lambda)
cij      = matrix(cij,ncol=4)
rij      = matrix(rep(c(1,2,3,4),each=Nsim),ncol=4)
numer    = exp(rij-cij)
denoms   = apply(numer,1,sum) + 1
cprob_ij = numer/denoms
cprob_j  = apply(cprob_ij,2,mean)
cprob_j
```

```
## [1] 0.04893091 0.11920240 0.26264964 0.51100375
```

An Economic Example: Product Selection

- To check out derivations, we will also simulate these choices *not using* the expression for the choice probability conditional on $\boldsymbol{c}_{ij}$
  - This is to check out work, but also in case y'all didn't follow that derivation!

An Economic Example: Product Selection

- To check out derivations, we will also simulate these choices *not using* the expression for the choice probability conditional on c_{ij}
  - This is to check out work, but also in case y'all didn't follow that derivation!

```
library(evd)
Nsim      = 100000
lambda    = 1/2
cij       = rexp(4*Nsim,rate=lambda)
epsij     = rgumbel(5*Nsim)
cij       = matrix(c(rep(0,Nsim),cij),ncol=5)
epsij     = matrix(epsij,ncol=5)
rs        = matrix(rep(0:4,each=Nsim),ncol=5)
profits   = rs - cij + epsij
choice    = apply(profits,1,which.max)-1
sim_cprob = sapply(1:4,function(x){mean(choice==x)})
sim_cprob
```

```
## [1] 0.04836 0.11866 0.26247 0.51249
```

Expectation Simulation

Expectation (Average) Simulation

- Review: The expectation (or average) of a discrete random variable X is

$$E[X] = \sum_{x_i} x f_X(x),$$

and if X is continuous

$$E[X] = \int_{-\infty}^{\infty} x f_X(x) dx.$$

- Assume X is continuous. Then if X has probability density function f_X and $g(x)$ is any function, then it is true that $E[g(X)] = \int g(x) f_X(x) dx$.
    - Note: $E[g(X)] \neq g(E[X])$ unless $g(x) = ax + b$.
- For those who have taken a lot of calculus, you know that some integrals don't have closed form solutions.
- So while we might know that $E[g(X)] = \int g(x) f_X(x) dx$, that doesn't mean we can always calculate it.
- Simulation!

Example: The Uniform Distribution

- Suppose $X \sim \mathbf{Uniform}(0, 1)$. Then

$$E[X] = \int_0^1 x dx = \frac{1}{2} x^2 \Big|_0^1 = \frac{1}{2}$$

  - Note $f_X(x) = 1$ for $\mathbf{Uniform}(0, 1)$, so I did not forget about it.
- Now, suppose $f(x) = x^2$. Then

$$E[X^2] = \int_0^1 x^2 dx = \frac{1}{3} x^3 \Big|_0^1 = \frac{1}{3}$$

- Let's check these results with simulation.

Example: The Uniform Distribution

- Suppose $X \sim \mathbf{Uniform}(0,1)$. Then

$$E[X] = \int_0^1 x dx = \frac{1}{2} x^2 \Big|_0^1 = \frac{1}{2}$$

  - Note $f_X(x) = 1$ for $\mathbf{Uniform}(0,1)$, so I did not forget about it.
- Now, suppose $f(x) = x^2$. Then

$$E[X^2] = \int_0^1 x^2 dx = \frac{1}{3} x^3 \Big|_0^1 = \frac{1}{3}$$

- Let's check these results with simulation.

```r
Nsim = 100
xs   = runif(Nsim) #store draws from uniform(0,1)
Ex   = mean(xs)    #estimate E[X] = 1/2
Ex2  = mean(xs^2)  #estimate E[X^2] = 1/3
c(Ex,Ex2)
```

```
## [1] 0.4753781 0.3049598
```

Example: The Uniform Distribution

- Suppose $X \sim \textbf{Uniform}(0,1)$. Then

$$E[X] = \int_0^1 x dx = \frac{1}{2}x^2\Big|_0^1 = \frac{1}{2}$$

  - Note $f_X(x) = 1$ for $\textbf{Uniform}(0,1)$, so I did not forget about it.
- Now, suppose $f(x) = x^2$. Then

$$E[X^2] = \int_0^1 x^2 dx = \frac{1}{3}x^3\Big|_0^1 = \frac{1}{3}$$

- Let's check these results with simulation.

```
Nsim = 100
xs   = runif(Nsim) #store draws from uniform(0,1)
Ex   = mean(xs)    #estimate E[X] = 1/2
Ex2  = mean(xs^2)  #estimate E[X^2] = 1/3
c(Ex,Ex2)
```

```
## [1] 0.4753781 0.3049598
```

This is as complicated as it gets! Would y'all like to see more examples?

Monte Carlo Simulation

Monte Carlo Simulation

- Sometimes we would like to examine if a statistical estimation procedure we come up with actually works the way we hope it should.
- To do this, we can simulate the DGP, perform the estimation procedure multiple times, and see if it's right "on average."
- This is called Monte Carlo simulation.

Review of LLN

- The Law of Large Numbers (LLN) states that if we have iid data drawn from "well behaved" distributions, as the sample size gets bigger, the population mean converges to the actual mean.
- In math,

$$\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} X_i = E[X]$$

(Sorta, this is actually not exactly correct; the limit should actually be a probability limit, but this is the right intuition)

Review of LLN

- The Law of Large Numbers (LLN) states that if we have iid data drawn from "well behaved" distributions, as the sample size gets bigger, the population mean converges to the actual mean.

- In math,

$$\lim_{n \to \infty} \frac{1}{n} \sum_{i=1}^{n} X_i = E[X]$$

  (Sorta, this is actually not exactly correct; the limit should actually be a probability limit, but this is the right intuition)

- Note: we've actually already used this result when using simulations to calculate expectations!

- Let's see an example.

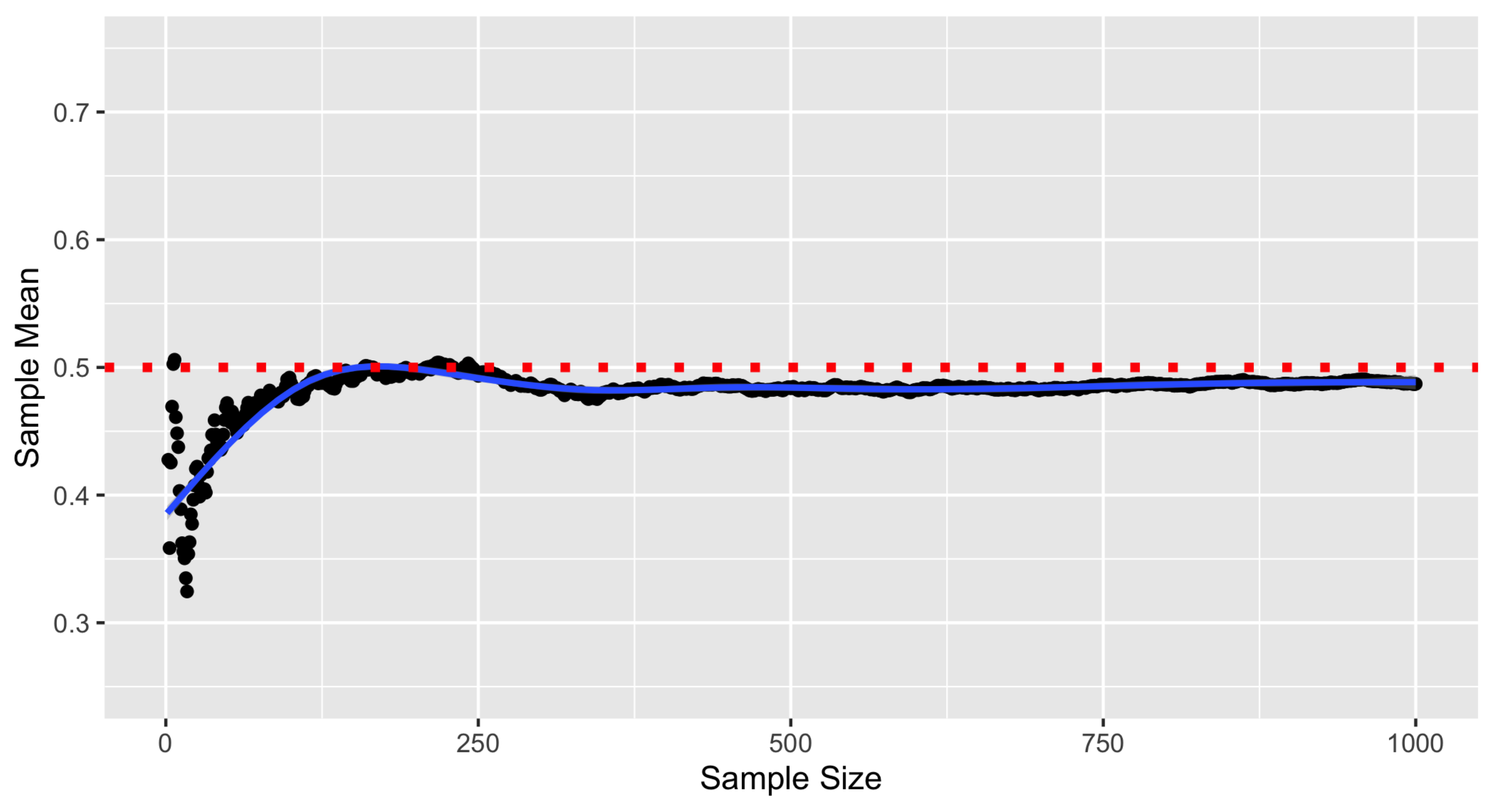
LLN Example

- Suppose we have $1,000$ draws from $\mathrm{Uniform}(0,1)$.
- Take cumulative sum to simulate getting an extra draw each time.

```
Nsim       = 1000
samp_means = cumsum(runif(Nsim))/(1:Nsim)
plot_data = data.table(x=1:Nsim,y=samp_means)
LLN_plot = ggplot(plot_data,aes(x=x,y=y)) +
  geom_point() + geom_smooth() + coord_cartesian(ylim=c(0.25,0.75)) +
  geom_hline(yintercept=1/2, linetype='dotted', col = 'red',size=1.5) +
  ylab("Sample Mean") + xlab("Sample Size")
```

LLN Example

- Notice how the line gets closer to the true mean, $1/2$.
- However, improvement is not monotonic; also, after a bit, improvement is slow.



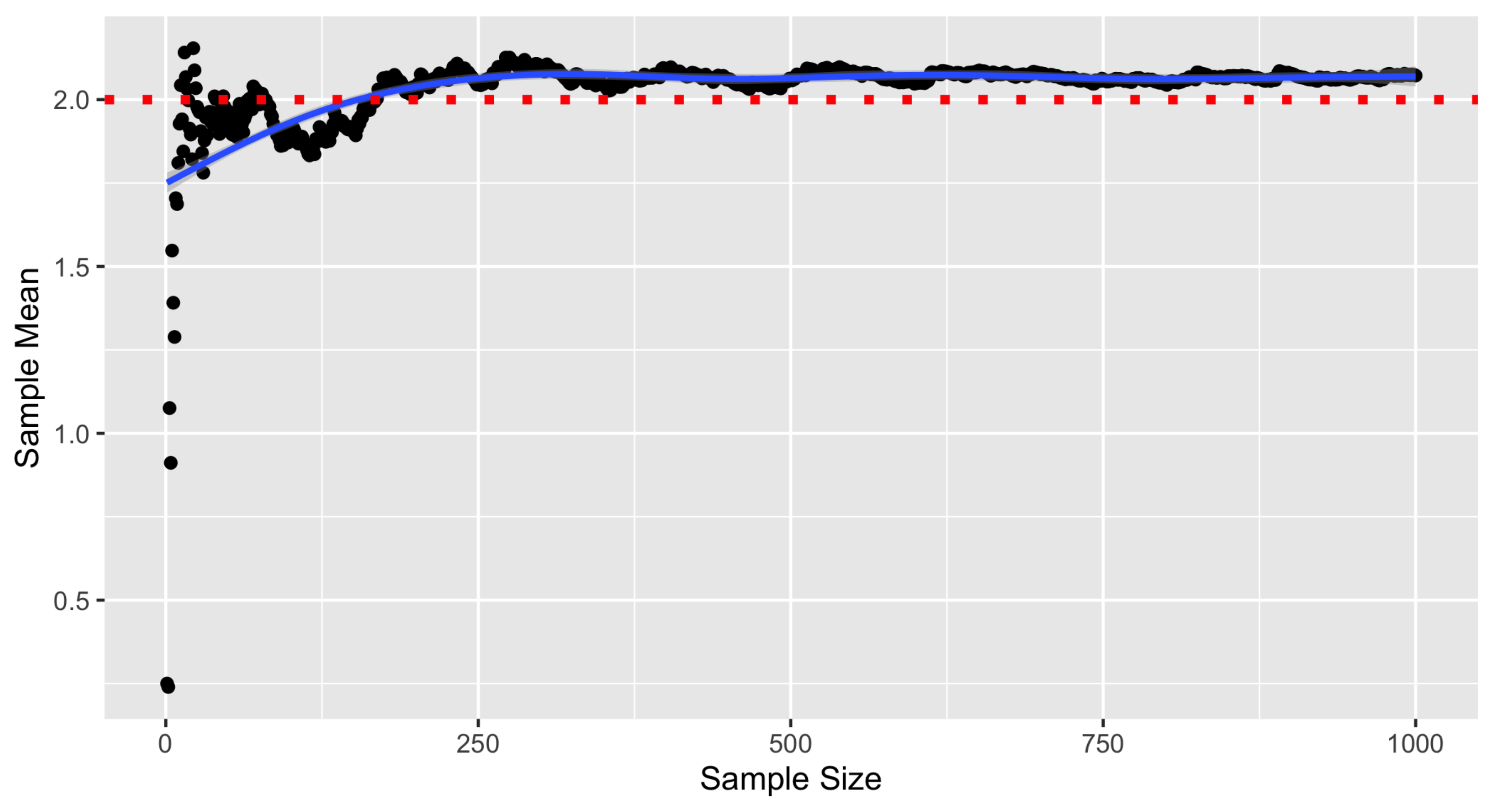
LLN Example 2

- Suppose we have $1,000$ draws from $\mathrm{Exp}(1/2)$.
- Take cumulative sum to simulate getting an extra draw each time.

```
Nsim       = 1000
lambda     = 1/2
samp_means = cumsum(rexp(Nsim,lambda))/(1:Nsim)
plot_data = data.table(x=1:Nsim,y=samp_means)
LLN_plot2 = ggplot(plot_data,aes(x=x,y=y)) +
  geom_point() + geom_smooth()  +
  geom_hline(yintercept=2, linetype='dotted', col = 'red',size=1.5) +
  ylab("Sample Mean") + xlab("Sample Size")
```

LLN Example 2

- Notice how the line gets closer to the true mean, 2.
- Notice that this one appears to converge faster than the uniform example.



Review of CLT

- Before finally examining a Monte Carlo simulation, we need to review the Central Limit Theorem (CLT).
- While the LLN tells us what the sample mean converges to, the CLT tells us the distribution of the sample mean converges to.
- An estimator is a function that takes in random variables and spits out an estimate.
- As such, estimators are random variables too!
- The idea is that the sample mean is a function of random data and is thus random itself.
  - If you took another sample, the sample mean would be slightly different.
  - Picture doing this many times; the CLT tells us what this distribution will be.
- If $\hat{\mu}$ is the sample mean, the CLT tells us that

$$\hat{\mu} \sim^{A} N(\mu_X, \sigma_X^2/n)$$

- Note:
  - $E[\hat{\mu}] = E[\frac{1}{n}\sum_{i=1}^{n} X_i] = \frac{1}{n}\sum_{i=1}^{n} E[X_i] = \mu_X$
  - $Var[\hat{\mu}] = Var[\frac{1}{n}\sum_{i=1}^{n} X_i] = \frac{1}{n^2}\sum_{i=1}^{n} Var[X_i] = \frac{\sigma_X^2}{n}$

Monte Carlo Simulation

- The idea behind Monte Carlo (MC) simulation is that you can generate data from a DGP, estimate the parameters you're interested in, and then repeat a bunch of times.
- If your estimator "works," you should get a nice, normal distribution that matches the CLT.
- With MC, important to keep two different n's separate in your head:
  1. The sample size which is the n that corresponds to the CLT previously.
  2. The number of simulations which is how many times the MC simulation is repeated.
- To reiterate: the MC simulation algorithm, at a broad level is as follows:
  1. Generate data by simulating the DGP with a sample size of N_{samp},
  2. Estimate the parameters of your model,
  3. Store the estimates,
  4. Go back to step 1 until you've repeated it N_{sim} times.

Monte Carlo for the Sample Mean

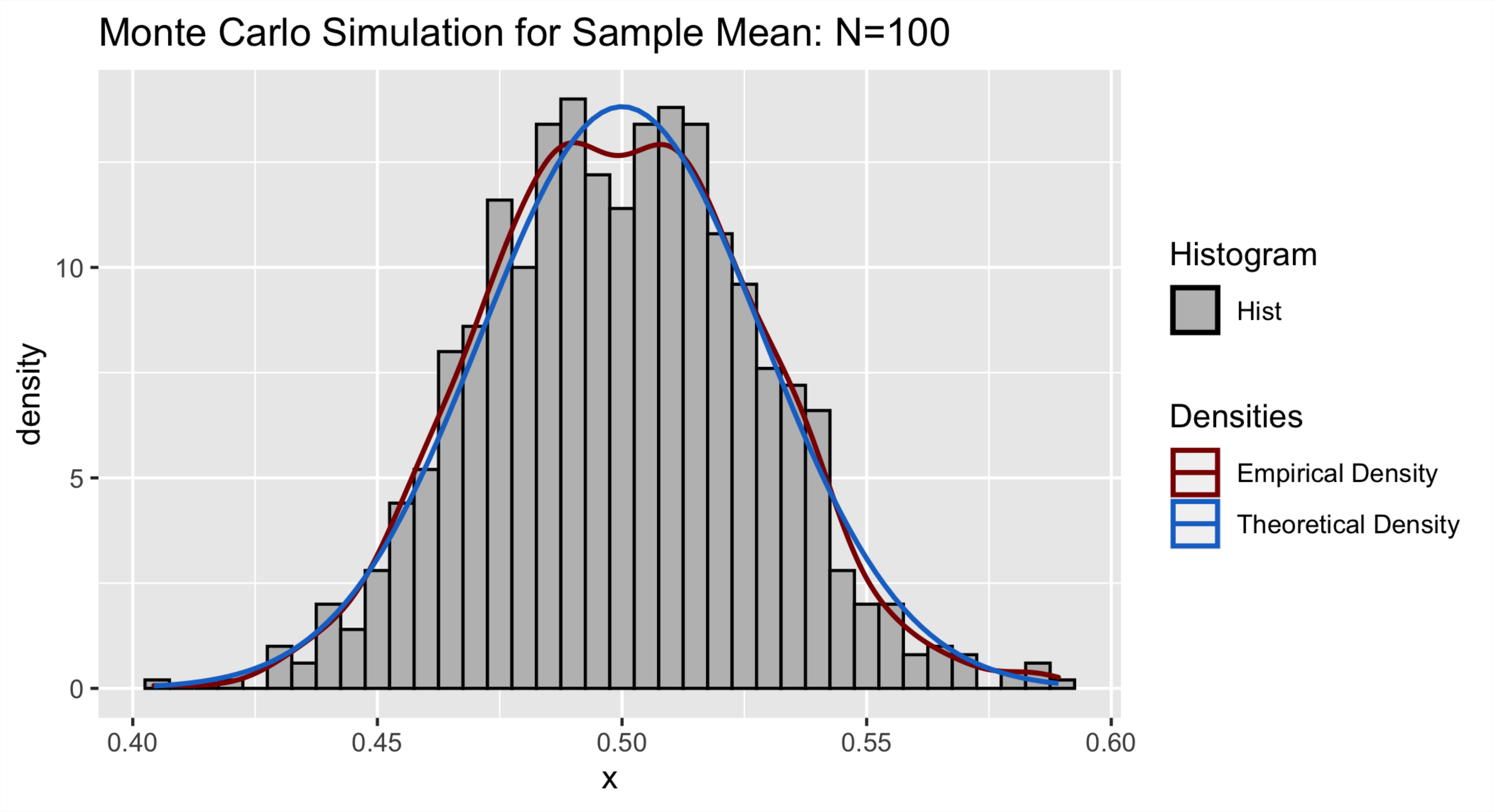
Let's run an MC simulation for the sample mean for data drawn idd from $\text{Uniform}(0, 1)$

```
Nsim         = 1000        #set number of simulations
Nsamp        = 100         #set sample size
sample_means = rep(0,Nsim) #preallocate vector to store sample means

for(sim in 1:Nsim){
  draws = runif(Nsamp)
  sample_means[sim] = mean(draws)
}

xs       = seq(min(sample_means),max(sample_means),length.out = 100)
ys       = dnorm(xs,mean=1/2,sd=sqrt(1/12/Nsamp))
den_data = data.table(x=xs,y=ys)
MC_data  = data.table(x = sample_means)
```

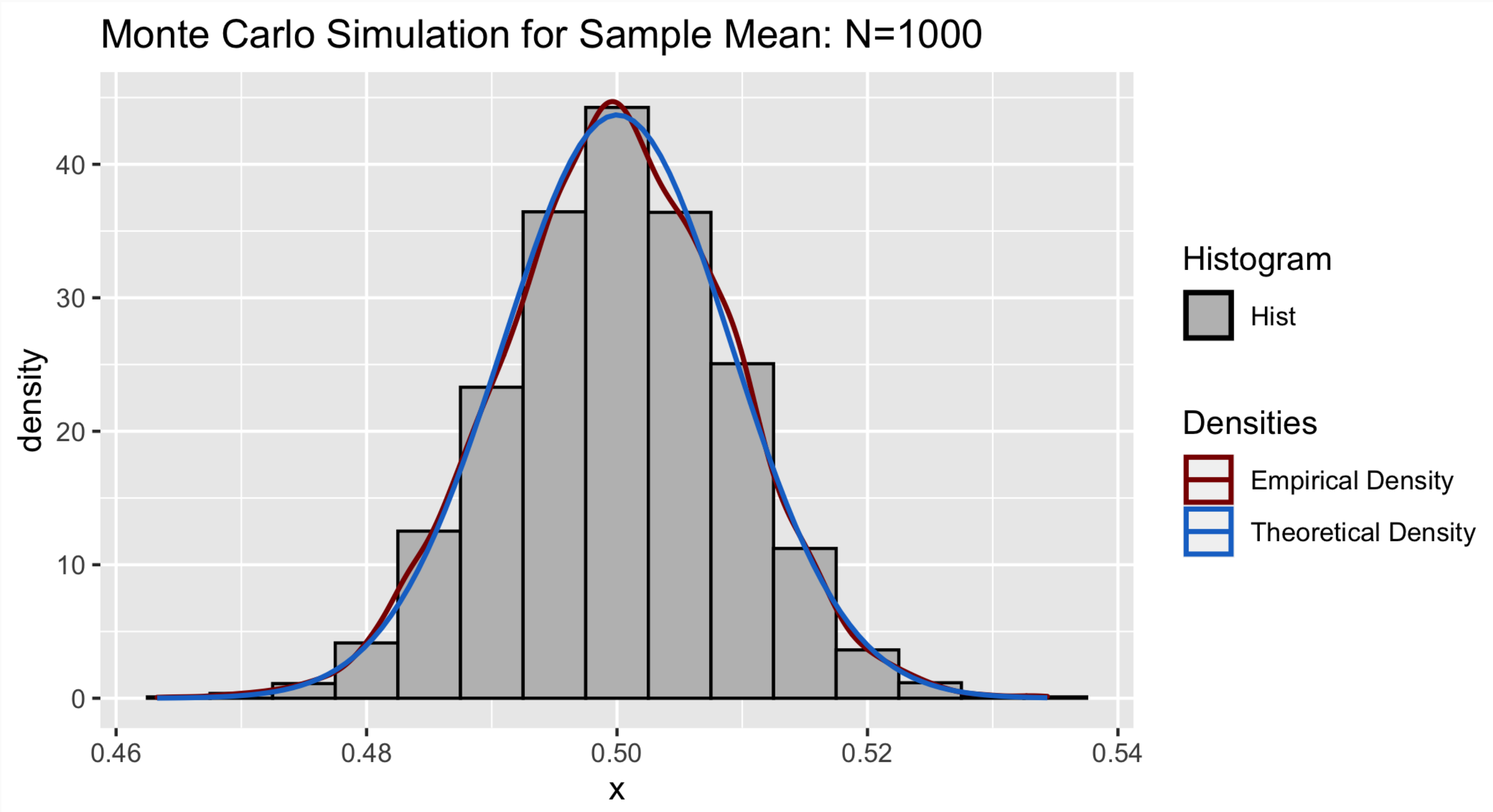
Monte Carlo Sample Mean Plot Code

```
den_cols = c("Empirical Density"="darkred","Theoretical Density"="dodgerblue3")

MC_plot = ggplot(data=MC_data,aes(x=x))+
  geom_histogram(aes(y = ..density..,fill="Hist"),color="black",binwidth = 0.005) +
  geom_density(aes(colour="Empirical Density"),size=0.8)+
  geom_line(aes(x = x, y = y,color = "Theoretical Density"),size=0.8,data=den_data) +
  scale_color_manual(name="Densities",values=den_cols) +
  scale_fill_manual(name="Histogram",values=c("Hist"="grey")) +
  labs(title=paste0("Monte Carlo Simulation for Sample Mean: N=",Nsamp))
```

Monte Carlo for the Sample Mean



Repeat with N=1000

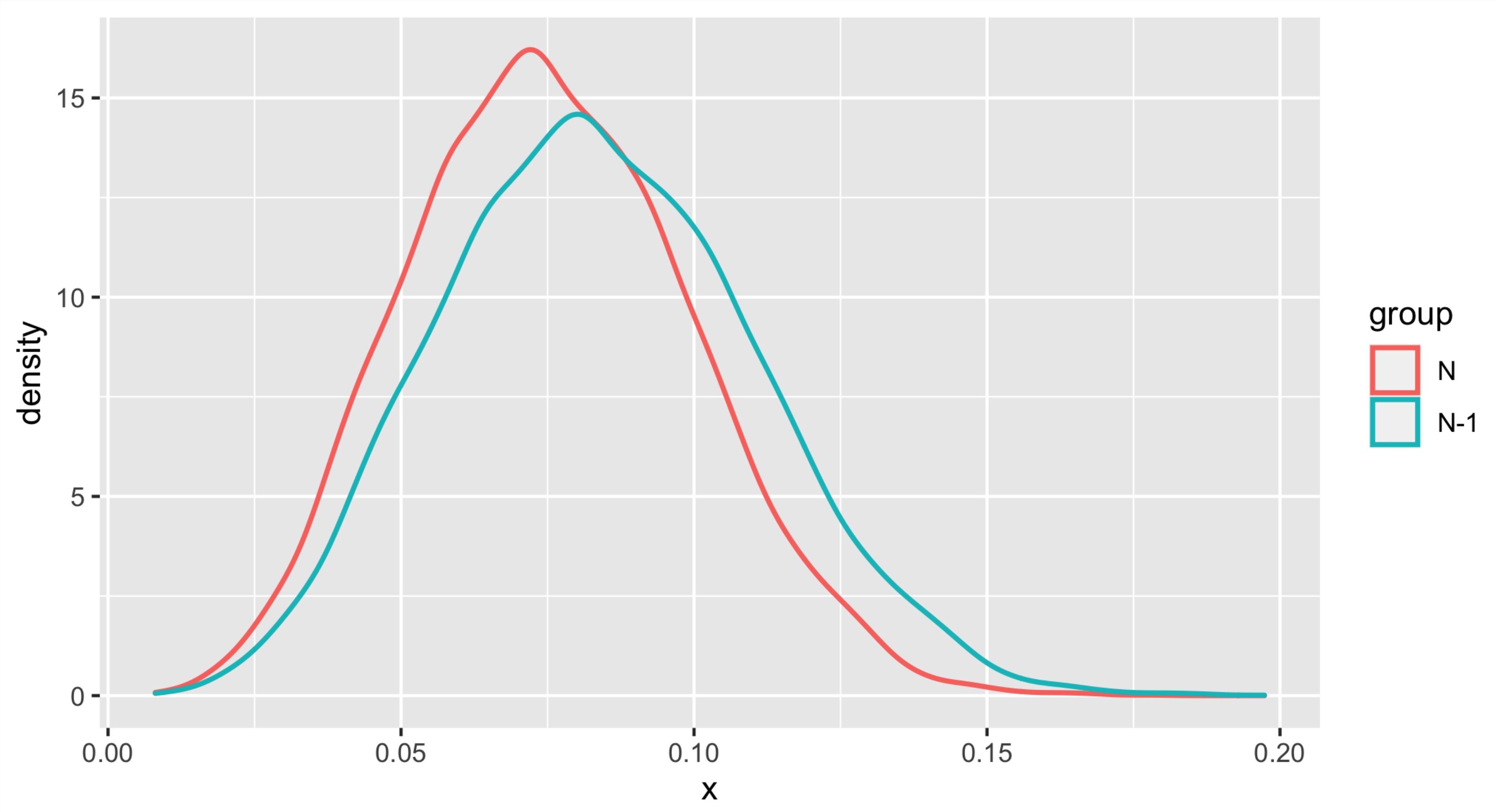


MCs for Other Estimators

- We can use MC simulation with any (consistent) estimator
- Let's try the two estimator's for the sample variance!
- The theoretical distribution is tedious to derive because you need the theoretical variance of the sample variance (confusing!), but we can still use MC to illustrate a point.

```
Nsim = 10000
Nsamp = 10
sample_varN    = rep(0,Nsim)    #preallocate
sample_varN1 = rep(0,Nsim)

for(sim in 1:Nsim){
  draws = runif(Nsamp)
  sample_varN[sim]  = mean((draws-mean(draws))^2)
  sample_varN1[sim] = var(draws)
}

MC_data_var = data.table(x=c(sample_varN,sample_varN1),
                         group=rep(c("N","N-1"),each=Nsim))
```

MC Plot for Sample Variances



First Look at the Bias Variance Trade-Off

```
##         groups       mean     variance
## 1:           N 0.07529201 0.0005836409
## 2:         N-1 0.08365779 0.0007205443
## 3: Theoretical 0.08333333           NA
```

Up Next: Numerical Methods & Optimization