# Stochastic Simulation

**A running collection of notes**

## APPM 7400, Section 004

**Meeting :** *MW 3-4:15PM in ECCR 257*

**Department of Applied Mathematics**

**J.N. Corcoran**

**Updated often, constantly changing and definitely tying to get away from you!**

*A good simulation, be it a religious myth or scientific theory, gives us a sense of mastery over experience. To represent something symbolically, as we do when we speak or write, is somehow to capture it, thus making it one's own. But with this appropriation comes the realization that we have denied the immediacy of reality and that in creating a substitute we have but spun another thread in the web of our grand illusion.*

*Heinz Pagels*

# About Coding and Visualization

You are welcome to use any software with which you are comfortable for this class. For visualization of results, however, I'd highly recommend that you use R which can be downloaded for free at www.r-project.org. You are welcome to use R for your actual simulations as well. I personally do not use R for my simulations as it can be slow for some of the more intense models and algorithms. Throughout these notes, I will present some algorithms generically using pseudocode. However, I will generate all Figures in R and include the code for figures in Appendix A.

# Contents

# List of Acronyms and Abbreviations

| | |
|---|---|
| **cdf** | *cumulative distribution function* |
| **CLT** | *Central Limit Theorem* |
| **iid** | *independent and identically distributed* |
| **pdf** | *probability density function* |
| **RNG** | *random number generator* |

Chapter 1

# Random Numbers

This is where it starts.

While it's totally fair to cringe at someone's misuse of the word "literally", I have to admit that I'm an unreasonable snob when it comes to my cringing reaction at the use of the word "random" by the general populace. In this case it's not misuse at all but simply context. In particular, the concept of a *random number*, in colloquial terms, is typically the first number that "pops" into one's head. I've been collecting these from people on the street for many years and my personal opinion is that the responses are far from random. As an experiment, please join me in collecting responses from people this semester and we'll see what kind of distribution we get at the end of the semester.

In mathematics, a **random number** is a realization (observation) of a uniformly distributed random variable. While it may be from a discrete or continuous uniform distribution on any given support set, the default is typically that it is a realization of a continuous uniform random variable on the interval from 0 to 1. A sequence of random numbers is, well, a sequence of such things, but the unspoken assumption is that they are independent.

| A sequence of random numbers is a realization of iid[1] uniform $(0, 1)$ random variables. |
|---|

In this course we are going to be simulating (producing realizations of) random variables and random (stochastic) processes in order to solve problems that would otherwise be intractable. Simulation is also a valuable tool for solving problems analytically. It can give you a conjecture to chase down theoretically, and, in the case where you already have one, it can give you an encouraging boost to keep trying for a proof. Random numbers are the basic building blocks for all stochastic simulation. They are produced by **random number generators (RNGs)** that are built in to most software packages. While we will briefly discuss RNGs in Section 1.1, we will not go into very much detail. This course assumes that you can produce a steady supply of random numbers with your favorite software and basically starts from there.

## 1.1 Random Number Generators (Brief!)

A random number generator is an algorithm used by a computer to generate a stream of independent realizations of a uniform(0,1) random variable.

[1] iid = independent and identically distributed

The CPU will do“this”...

... and a little bit of “that"...

...and “this other thing"...

... with some iteration.

While there have recently been some interesting quantum approaches to random number generation, most RNGs are completely deterministic recipes. No, I'm not kidding. Each number in the sequence is computed deterministically from the last. One might(!) be able to believe that a carefully constructed algorithm can produce things that “appear" uniform, they are as dependent as can be! The truth is that some of these algorithms are actually not so bad on both counts. However, produced random numbers are usually, and justifiably, referred to as **pseudorandom** numbers.

> *Anyone who considers arithmetical methods of producing random digits is of course in a state of sin.”*
>
> *- John von Neumann*

That said, we will still say “random numbers" throughout this course even if we are really using the “pseudo" variety. We will now give a very brief overview of some well known algorithms. In Sections 1.2 and 1.3 we will discuss statistical tests for uniformity and independence that can be used to evaluate these and other RNGs.

## Linear Congruential RNGs

One of the oldest and best known of modern techniques for generating random numbers relies on modular arithmetic and was first proposed by D.H. Lehmer in 1949.

Recall that, for a positive integer $m$, the real numbers $a$ and $b$ are **congruent modulo m** if the difference $a - b$ is exactly divisible by $m$. We write

$$a \equiv b \,(\mathrm{mod}\, m) \tag{1.1}$$

Strict evaluation of the right-hand side of (1.1) is the remainder in the division of $b$ by $m$. For example,

$$\begin{aligned} 7 \,(\mathrm{mod}\, 2) &= 1, \\ 7.2 \,(\mathrm{mod}\, 2) &= 1.2, \\ 7 \,(\mathrm{mod}\, 9) &= 7, \\ \text{and} \quad 8 \,(\mathrm{mod}\, 4) &= 0. \end{aligned}$$

(Notice the use of the equals sign instead of the equivalence in (1.1). We have, for example, $27 \equiv 5 \,(\mathrm{mod}\, 2)$, but $5 \,(\mathrm{mod}\, 2) := 1$.)

For a **linear congruential RNG**, one begins by choosing the following parameters.

- an integer *modulus* $m \geq 1$
- a *multiplier* $a$ such that $0 < a < m$
- an *increment* $b$ such that $0 \leq b < m$
- a *seed* (starting value) $x_0$ such that $0 \leq x_0 < m$

We then produce a sequence of numbers $x_1, x_2, \ldots$ using the relation

$$x_n := (ax_{n-1} + b)(\text{mod}\, m) \tag{1.2}$$

It is easy to see that $0 \leq x_n < m$ for all $n$. Thus, we can scale the numbers to live between 0 and 1 by divinding by $m$. That is, we output as "random numbers" the sequence $\{u_n\}$ where

$$u_n := x_n/m.$$

Aw hell no! Seriously?!? This is supposed to give us a sequence of numbers that appear to be iid uniforms over $(0, 1)$? It's actually not that bad for "good" choices of $a$, $b$, $m$, and $x_0$. Note that the sequence will start repeating since the function

$$f(x) := x(\text{mod}\, m)$$

is periodic with period $m$.

(Proof: For any integer $k$, we have $f(x + km) := (x + km)(\text{mod}\, m) = x(\text{mod}\, m) =: f(x)$.)

So, the first order of business is to choose $m$ to be as large as possible. (This is a machine dependent task.) Still, we must be careful about the entire combination of parameters. There is a famous flop (no pun intended) of an example, called RANDU, that was infamously put forth and used by IBM in the late 60's. The parameters were

$$a = 2^{16} + 3, \;\; b = 0, \;\; m = 2^{31},$$

and $x_0$ is taken to be any odd number.

Figure 1.1 shows a histogram of $100,000$ uniforms generated using RANDU. That looks pretty nice and uniform. (Note: I tend to use ridiculously large samples, if possible, when trying to show an algorithm working or not working in order to squash sampling variability. If there was, for example, a spike in this histogram we would know that the algorithm is clearly wrong– whereas a spike in a histogram of only 100 values could just be sampling variability.)

In Figure 1.2(a), we group our draws into triples and plot these points in $\mathbb{R}^3$. As expected, it looks like a "random" cloud of points uniformly dispersed in the cube. However, viewed from a certain precise angle, as in Figure 1.2(b), we see some serious structure– sixteen planes to be exact! So much for our random cloud... IBM's RANDU was in wide use in computing for more than a decade.

Many extensive PhD theses have been written about choosing parameters for the linear congruential random number generator. There are rules such as "Make sure that $a-1$ is divisible

Figure 1.1: 100,000 Uniform Draws Using RANDU

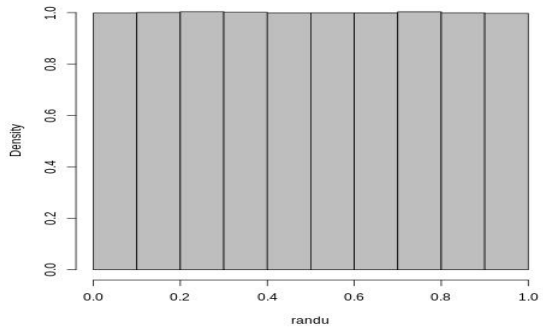
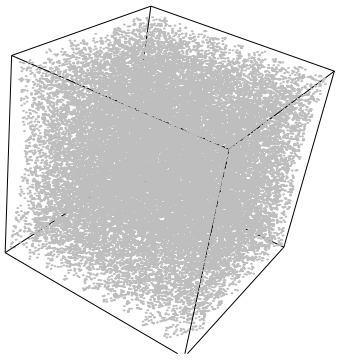
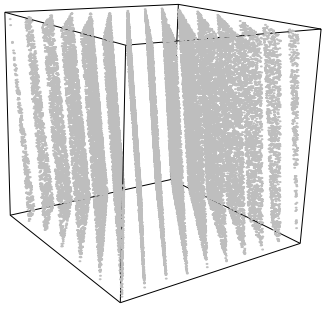



Figure 1.2: RANDU Draws Plotted in Triples



(a) Most views



(b) An interesting view!

by all prime factors of $m$." We will not being going into any of this in this course.

## Other RNGs

Warning: This Section is not even close to being exhaustive. It is barely a footnote!

Liner congruential RNGs with $b = 0$ (such as RANDU) are also known as **multiplicative random number generators**. Multiplicative random number generators with extra recurrence such as

$$x_n := (a_1 x_{n-1} + a_2 x_{n-2}) (\mathrm{mod}\, m)$$

are known as **multiple recursive generators**. Some RNGs rely on *shift registers* and *twisting* (permuting) bits from existing algorithms. Perhaps the mostly widely used algorithms today is called the **Mersenne Twister**. (I believe that it is the default in common software such as R, Matlab, Mathematica, etc...)

There is a nice 2018 summary of popular algorithms given by Debraj Bose from the Indian Statistical Institute in New Delhi. In case I forget to link it for you on our course website, it can be found (as of the writing of these notes) at

`www.isid.ac.in/~deepayan/ICP2017/projects/Debraj_Bose/report.pdf`

As I've said, this course starts with the assumption that you already have a good random number generator. In the next two Sections, we will discuss some tests that you can put it through.

## 1.2 Test for Independence

### 1.2.1 Runs Up and Down

I generated $100,000$ random numbers using a mysterious random number generator. The first 35 numbers were as follows. (Read from left to right across rows. Order is important!)

| | | | | | | |
|---|---|---|---|---|---|---|
| 0.536839 | 0.501837 | 0.945124 | 0.066075 | 0.926468 | 0.220971 | 0.975456 |
| 0.332541 | 0.594702 | 0.025603 | 0.644262 | 0.713022 | 0.858687 | 0.211142 |
| 0.683887 | 0.158859 | 0.555520 | 0.343992 | 0.539392 | 0.955368 | 0.162900 |
| 0.925698 | 0.723009 | 0.894404 | 0.373177 | 0.323461 | 0.117826 | 0.945908 |
| 0.097929 | 0.931847 | 0.644134 | 0.890236 | 0.968813 | 0.452754 | 0.225053 |

Since the second number in the table (0.501837) is smaller than the first (0.536839), the first number is the beginning of a "run down". Since the third number is larger than the second, the run down is short lived and has length 1. For this test, we are not concerned about the

length of runs. We just want to count the total number of runs up and down. In this sample of size 35 there are 27 total up/down runs. (Make sure you can count them.)

In the total sample of size $100,000$, there were $66,832$ runs. Does this seem right to you? How many do you expect? We will answer this question for a generic **random sample** (iid values) that is not necessarily from a uniform distribution.

**Theorem 1.2.1**
*Let $X_1, X_2, \ldots, X_n$ be a random sample of size $n$ from any distribution with probability density function (pdf) $f$ and cumulative distribution function (cdf) $F$.*

*Let $R_n$ be the total number of up/down runs in the sample.*

*Then*

$$\mathsf{E}[R_n] = \frac{2n-1}{3} \qquad \textit{and} \qquad Var[R_n] = \frac{16n-29}{90}.$$

PROOF First, note that $X_1$ will always start a run and $X_n$ will never start a run.

For $i = 2, 3, \ldots, n-1$, define the indicator

$$I_i = \begin{cases} 1 & , \text{ if } X_i \text{ starts a run} \\ 0 & , \text{ if } X_i \text{ does not start a run.} \end{cases}$$

Then

$$R_n = 1 + \sum_{i=2}^{n-1} I_i. \tag{1.3}$$

Note that

$$\begin{aligned} \mathsf{E}[I_i] &= P(I_i = 1) = P(X_i \text{ starts a run}) \\ &= P(X_i > X_{i-1}, X_i > X_{i+1}) + P(X_i < X_{i-1}, X_i < X_{i+1}) \\ &= 2 \cdot P(X_i > X_{i-1}, X_i > X_{i+1}) \end{aligned}$$

by symmetry. (If you are not convinced of the symmetry, compute both terms!)

Now, conditioning on the value of $X_i$ (and using independence), we have

$$\begin{aligned} \mathsf{E}[I_i] &= 2 \cdot P(X_i > X_{i-1}, X_i > X_{i+1}) \\ &= 2 \textstyle\int_{-\infty}^{\infty} P(X_{i-1} < x, X_{i+1} < x)\, f(x)\, dx \\ &\overset{indep}{=} 2 \textstyle\int_{-\infty}^{\infty} P(X_{i-1} < x) \cdot P(X_{i+1} < x)\, f(x)\, dx \\ &= 2 \textstyle\int_{-\infty}^{\infty} F^2(x)\, f(x)\, dx. \end{aligned}$$

Making the $u$-substitution $u = F(x)$, and noting that $du = F'(x)\,dx = f(x)\,dx$, we get

$$\mathsf{E}[I_i] = 2\,\frac{F^3(x)}{3}\bigg|_{-\infty}^{\infty} = \frac{2}{3}(1^3 - 0^3) = \frac{2}{3}.$$

Thus, we have that

$$\mathsf{E}[R_n] = 1 + \sum_{i=2}^{n-1} \mathsf{E}[I_i] = 1 + \sum_{i=2}^{n-1} \frac{2}{3} = 1 + \frac{2}{3}\cdot(n-2) = \frac{2n-1}{3}. \qquad \surd$$

For the variance, note that

$$Var[R_n] = Var\left[1 + \sum_{i=2}^{n-1} I_i\right] = Var\left[\sum_{i=2}^{n-1} I_i\right]$$

but that we can only pull the sum out of the variance if the $I_i$ are independent. (They aren't! In particular, for the $unif(0,1)$ distribution, not seeing the start of an up/down run for a long time would push us towards 0 or 1, eventually forcing a turnaround and the start of a new run.) So, we will have to consider some covariances here. Note that

$$Var\left[\sum_{i=2}^{n-1} I_i\right] = Cov\left(\sum_{i=2}^{n-1} I_i, \sum_{j=2}^{n-1} I_j\right) = \sum_{i=2}^{n-1}\sum_{j=2}^{n-1} Cov(I_i, I_j).$$

Also,

$$Cov(I_i, I_j) = \mathsf{E}[I_i I_j] - \mathsf{E}[I_i]\mathsf{E}[I_j] = \mathsf{E}[I_i I_j] - \left(\frac{2}{3}\right)^2.$$

Now

$$\begin{aligned}
\mathsf{E}[I_i I_j] &= 0 \cdot 0 \cdot P(I_i = 0, I_j = 0) + \cdots \\
&= P(I_i = 1, I_j = 1) \\
&= P(X_i \text{ starts a run and } X_j \text{ starts a run})
\end{aligned}$$

Case: $j = i$ (More generally $|i - j| = 0$)

$$\mathsf{E}[I_i^2] = \mathsf{E}[I_i] = \frac{2}{3}$$

So,

$$Cov(I_i, I_i) = Var[I_i] = \mathsf{E}[I_i^2] - (\mathsf{E}[I_i])^2 = \frac{2}{3} - \left(\frac{2}{3}\right)^2 = \frac{2}{9}.$$

If you imagine the terms in the covariance double sum arranged in an $(n-2)\times(n-2)$ matrix, these covariances correspond to the $n-2$ diagonal terms.

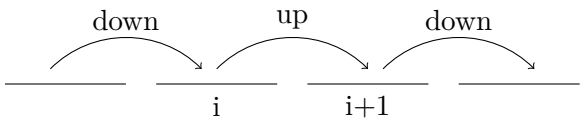
Case: $j = i + 1$ (More generally $|i - j| = 1$)

$$\begin{aligned}
\mathsf{E}[I_i I_{i+1}] &= P(X_i \text{ starts a run and } X_{i+1} \text{ starts a run}) \\
&= P(X_i \text{ starts a run up and } X_j \text{ starts a run up}) \\
&\quad + P(X_i \text{ starts a run up and } X_j \text{ starts a run down}) \\
&\quad + P(X_i \text{ starts a run down and } X_j \text{ starts a run up}) \\
&\quad + P(X_i \text{ starts a run down and } X_j \text{ starts a run down})
\end{aligned}$$

Since the indices ($i$ and $i+1$) are so close together, it is not possible for $X_i$ and $X_{i+1}$ to either both start a run up or both start a run down. I claim that both other cases (up/down and down/up) are symmetric. (However, if you are unconvinced, you should compute both!) Thus,

$$\mathsf{E}[I_i I_{i+1}] = 2 \cdot P(X_i \text{ starts a run up and } X_j \text{ starts a run down}).$$

Envisioning "slots" for $X_{i-1}$, $X_i$, $X_{i+1}$, and $X_{i+2}$, we have




(Note that we will always have "slots" to work with on the left and right since our covariance sums run over indices in $\{2, 3, \ldots, n-1\}$.)

So, we see that

$$\begin{aligned}
\mathsf{E}[I_i I_{i+1}] &= 2 \cdot P(X_i \text{ starts a run up and } X_j \text{ starts a run down}) \\
&= 2 \cdot P(X_i < X_{i-1}, X_{i+1} > X_i, X_{i+2} < X_{i+1}).
\end{aligned}$$

♢ Conditioning on the values of $X_i$ and $X_{i+1}$ gives

$$\begin{aligned}
\mathsf{E}[I_i I_{i+1}] &= 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(X_i < X_{i-1}, X_{i+1} > X_i, X_{i+2} < X_{i+1} | X_i = x, X_{i+1} = y)\, f(x)\, f(y)\, dx\, dy \\
&= 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(x < X_{i-1}, y > x, X_{i+2} < y | X_i = x, X_{i+1} = y)\, f(x)\, f(y)\, dx\, dy
\end{aligned}$$

At this point, the stuff after the conditional line can be dropped because all random variables of the left side of the line are independent of those on the right.

$$\mathsf{E}[I_i I_{i+1}] = 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} P(x < X_{i-1}, y > x, X_{i+2} < y)\, f(x)\, f(y)\, dx\, dy$$

The non-random thing ($y > x$) in the probability will cause the probability to be zero if it is ♢ not true and equal to the probability of the remaining things if it is. This ends up causing a change in the limits of integration. Thus,we have

$$\mathsf{E}[I_i I_{i+1}] = 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{y} P(x < X_{i-1}, X_{i+2} < y)\, f(x)\, f(y)\, dx\, dy$$

♢ = extra detail is being shown here that will be omitted for the rest of the proof/explanation

By independence of $X_{i-1}$ and $X_{i+2}$, that joint probability will factor into a product of probabilities, each of which can be written with a cdf.

$$\mathsf{E}[I_i I_{i+1}] \;=\; 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{y} [1 - F(x)]\, F(y)\, f(x)\, f(y)\, dx\, dy \tag{1.4}$$

For the inner integral, make the substitution $u = F(x)$ with $du = F'(x)\, dx = f(x)dx$. We $\diamondsuit$ then get the inner integral to be

$$\left[F(x) - \frac{1}{2}F^2(x)\right]_{x=-\infty}^{x=y} = F(y) - \frac{1}{2}F^2(y)$$

since the cdf evaluated at $-\infty$ is zero. Plugging this back into (1.4) we have

$$\begin{aligned}
\mathsf{E}[I_i I_{i+1}] &= 2 \cdot \int_{-\infty}^{\infty} \left[F(y) - \tfrac{1}{2}F^2(y)\right] F(y)\, f(y)\, dy \\
&= 2 \cdot \int_{-\infty}^{\infty} \left[F^2(y) - \tfrac{1}{2}F^3(y)\right] f(y)\, dy \\
&\overset{u=F(y)}{=} 2 \cdot \left[\tfrac{1}{3}F^3(y) - \tfrac{1}{8}F^4(y)\right]_{y=-\infty}^{y=\infty} \\
&= 2 \cdot \left[\tfrac{1}{3} - \tfrac{1}{8}\right] = \tfrac{5}{12}.
\end{aligned}$$

So,

$$Cov(I_i, I_{i+1}) = \mathsf{E}[I_i I_{i+1}] - \mathsf{E}[I_i]\mathsf{E}[I_{i+1}] = \frac{5}{12} - \left(\frac{2}{3}\right)^2 = -\frac{1}{36}.$$

If you again imagine the terms in the covariance double sum arranged in an $(n-2) \times (n-2)$ matrix, these covariances correspond to the $2(n-3)$ off diagonal terms.

Case: $j = i + 2$ (More generally $|i - j| = 2$)

Again,

$$\mathsf{E}[I_i I_{i+2}] = P(X_i \text{ starts a run and } X_j \text{ starts a run})$$

and this time (draw the “slots" picture) all four (up/up, up/down, down/up, down/down) events are possible. There is symmetry in the up/down and down/up cases and also in the up/up and down/down cases, but all four cases are not the same. Thus,

$$\begin{aligned}
\mathsf{E}[I_i I_{i+2}] \;=\; & 2 \cdot P(X_i \text{ starts a run up and } X_j \text{ starts a run up}) \\
& +2 \cdot P(X_i \text{ starts a run up and } X_j \text{ starts a run down}).
\end{aligned}$$

Again, if you are not convinced of this you should compute all 4 terms!

If you draw the “slots" picture, you will see that

$$\begin{gathered}
P(X_i \text{ starts a run up and } X_j \text{ starts a run up}) = \\
P(X_i < X_{i-1}, X_{i+1} > X_i, X_{i+2} < X_{i+1}, X_{i+3} > X_{i+2})
\end{gathered}$$

and

$$P(X_i \text{ starts a run up and } X_j \text{ starts a run down}) = \\ P(X_i < X_{i-1}, X_{i+1} > X_i, X_{i+2} > X_{i+1}, X_{i+3} < X_{i+2}).$$

After conditioning on the values of $X_i$, $X_{i+1}$, and $X_{i+2}$, to compute these, you will get $11/120$ and $2/15$, respectively.

In all,

$$\mathsf{E}[I_i I_{i+2}] = \frac{9}{20}$$

and so

$$Cov(I_i, I_{i+2}) = \frac{9}{20} - \left(\frac{2}{3}\right)^2 = \frac{1}{180}.$$

There are $2(n-4)$ ("second off of the diagonal") such terms.

Case: $|i-j| \geq 3$

If you draw the "slots" picture, you will probably suspect that whether or not $X_i$ starts a run is independent of whether or not $X_j$ starts a run at this distance. Indeed, if you go through the same integral machinations, you will get $\mathsf{E}[I_i I_j] = 4/9$ which will leave you with

$$Cov(I_i, I_j) = \frac{4}{9} - \left(\frac{2}{3}\right)^2 = 0.$$

In conclusion,

$$\begin{aligned} Var\left[\sum_{i=2}^{n-1} I_i\right] &= \sum_{i=2}^{n-1} \sum_{j=2}^{n-1} Cov(I_i, I_j) \\ &= \tfrac{2}{9}(n-2) - \tfrac{1}{36} \cdot 2(n-3) + \tfrac{1}{180} \cdot 2(n-4) = \tfrac{16n-29}{90} \end{aligned}$$

as desired. □

So, what can we do with this mean and variance? In our sample of size $n = 100,000$, we had $66,832$ runs. If the values sampled were really independent, we would expect

$$\mathsf{E}[R_{100,000}] = \frac{2(100000) - 1}{3} = 66666.3\overline{3}$$

runs. Since

$$Var[R_{100000}] \approx 17777.46$$

we see that our observed number of runs is about

$$\frac{66832 - 66666.33}{\sqrt{17777.46}} \approx 1.24$$

standard deviations above the expected value. Is this far away? A distribution would be helpful. From (1.5) we see that the number of runs is a sum of random variables (plus a constant). Although the indicators in the sum are identically distributed, they are not

independent. Thus, the classical Central Limit Theorem (CLT) does not apply. However, there are versions of the CLT that apply to sequences with various forms of weak dependence. In particular, there is a CLT for sequences with **finite range dependence** such as is the case here. Following the proof of Theorem 1.2.1 a bit further, we can easily show our claim that $Cov(I_i, I_j) = 0$ if $|i - j| \geq 3$. This is not enough to say that the indicators at this distance are independent but at least it is promising. It is not hard to show actual lag three (or higher) independence by looking at joint distributions of the indicators as opposed to only expectations of their products. The basic idea behind showing the finite range independence version of the CLT would be to break up the sum of the indicators into sub-sums of the independent ones. We can then get asymptotic normality for each of the sub-sums using the classical CLT and then put them all back together as a sum of normals.

So, for a "large" sample size $n$, if the values in the sample are independent, $R_n$ is roughly normally distributed. We can standardize $R_n$ into something that has roughly a standard normal distribution. A standard normal random variable will be between $-1.96$ and $1.96$ for 95% of the time. Thus, a rough hypothesis test for independence, with level of significance $\alpha = 0.05$, is given by the following.

**Runs Up/Down Test for Independence**
For a sample of some "large" size $n$, proceed as follows.

- Count the total number of runs in the sample. Call it $R_n$.
- Compute $\mu_n = \mathsf{E}[R_n]$ and $\sigma_n^2 = Var[R_n]$ as defined by Theorem 1.2.1.
- Compute
$$Z_n = \frac{R_n - \mu_n}{\sigma_n}.$$
- If $-1.96 \leq Z_n < 1.96$ we say that this sample has "passed" the runs up and down test for independence!

**Example:**

For our sample of size $n = 100,000$, we observed $R_{100,000} = 66,832$. We have already computed that

$$\mu_{100,000} = 66,666.33 \qquad \text{and} \qquad \sigma_n^2 = 17,777.46.$$

Thus we have the test statistic

$$Z_n = \frac{R_n - \mu_n}{\sigma_n} = \frac{66832 - 66666.33}{\sqrt{17777.46}} \approx 1.24.$$

Since this is in the interval $(-1.96, 1.96)$, our RNG has passed the runs up/down test for independence. Equivalently, plugging $-1.96$ and $1.96$ in for $Z_n$, we would pass for any $R_n$ in the interval $(66,405, 66,928)$. □

With so much approximation going on here, combined with the fact that a dependent sample could possibly pass, I would never advise using this test alone to establish independence of values from a RNG. It should be one of a battery of tests. Read on!

### 1.2.2 Runs Above and Below the Mean

The test in the last section had nothing to do with uniformity in particular. In this Section, we consider runs of numbers in the sample that are above or below the uniform $(0, 1)$ mean of 0.5. As before, consider the first 35 values of our random sample of size $100,000$. (Read across rows, order is important!)

| | | | | | | |
|---|---|---|---|---|---|---|
| 0.536839 | 0.501837 | 0.945124 | 0.066075 | 0.926468 | 0.220971 | 0.975456 |
| 0.332541 | 0.594702 | 0.025603 | 0.644262 | 0.713022 | 0.858687 | 0.211142 |
| 0.683887 | 0.158859 | 0.555520 | 0.343992 | 0.539392 | 0.955368 | 0.162900 |
| 0.925698 | 0.723009 | 0.894404 | 0.373177 | 0.323461 | 0.117826 | 0.945908 |
| 0.097929 | 0.931847 | 0.644134 | 0.890236 | 0.968813 | 0.452754 | 0.225053 |

This sample starts out with a run of length 3 of values that are above the mean of 0.5. Then there is a run of length 1 below the mean, followed by a run of length 1 above the mean. For this test, we are not concerned with the length of each run. Instead, we only want to count the total number of runs in the sample. In this subsample of size 35, there are 22 runs above and below the mean. (Make sure you can count them!) In the full sample of size $100,000$, there were $49,908$ such runs. For a true random sample (iid) from the uniform distribution on $(0, 1)$, what do we expect to see?

**Theorem 1.2.2**
*Let $X_1, X_2, \ldots, X_n$ be a sample of size $n$ from the $unif(0, 1)$ distribution.*

*Let $R_n$ be the total number of above/below mean runs in the sample.*

*Then, if $X_1, X_2, \ldots, X_n$ are independent,*

$$\mathsf{E}[R_n] = \frac{n+1}{2} \qquad \textit{and} \qquad Var[R_n] = \frac{n-1}{4}.$$

PROOF As in the previous Section, note that $X_1$ will always start a run. Unlike the previous Section, it is possible for $X_n$ to start a run. Overall, this proof will be easier than the previous proof since you can tell whether $X_i$ is part of a run above or below the mean just by looking at it whereas you need to look at neighboring values to determine whether it is a part of an up or down run. (That said, you do need to look at neighboring values to determine whether of not $X_i$ is the start of an above/below run.)

For $i = 2, 3, \ldots, n$, define the indicator

$$I_i = \begin{cases} 1 & , \text{ if } X_i \text{ starts a run} \\ 0 & , \text{ if } X_i \text{ does not start a run.} \end{cases}$$

Then

$$R_n = 1 + \sum_{i=2}^{n} I_i. \tag{1.5}$$

Note that, for $i = 2, 3, \ldots, n$,

$$\begin{aligned}
\mathsf{E}[I_i] &= P(X_i \text{ starts a run}) \\
&= P(X_i \text{ starts a run above the mean}) + P(X_i \text{ starts a run below the mean}) \\
&\overset{symm}{=} 2 \cdot P(X_i \text{ starts a run above the mean}) \\
&= 2 \cdot P(X_i > 0.5, X_{i-1} < 0.5) \\
&\overset{indep}{=} 2 \cdot P(X_i > 0.5) \cdot P(X_{i-1} < 0.5) \\
&\overset{unif}{=} 2\left(\tfrac{1}{2}\right)\left(\tfrac{1}{2}\right) = \tfrac{1}{2}.
\end{aligned}$$

Thus,

$$\mathsf{E}[R_n] = 1 + \sum_{i=2}^{n} \mathsf{E}[I_i] = 1 + \sum_{i=2}^{n} \frac{1}{2} = 1 + \frac{1}{2}(n-1) = \frac{n+1}{2},$$

as desired.

For the variance, note that

$$\begin{aligned}
Var[R_n] &= Var\left[1 + \textstyle\sum_{i=2}^{n} I_i\right] = Var\left[\textstyle\sum_{i=2}^{n} I_i\right] \\
&= Cov\left(\textstyle\sum_{i=2}^{n} I_i, \sum_{j=2}^{n} I_j\right) = \textstyle\sum_{i=2}^{n} \sum_{j=2}^{n} Cov(I_i, I_j).
\end{aligned}$$

Case: $j = i$ (More generally $|i - j| = 0$)

$$\mathsf{E}[I_i^2] = \mathsf{E}[I_i] = \frac{1}{2}$$

So,

$$Cov(I_i, I_i) = Var[I_i] = \mathsf{E}[I_i^2] - (\mathsf{E}[I_i])^2 = \frac{1}{2} - \left(\frac{1}{2}\right)^2 = \frac{1}{4}.$$

If you imagine the terms in the covariance double sum arranged in an $(n-1) \times (n-1)$ matrix, these covariances correspond to the $n - 1$ diagonal terms.

Since $(1/4)(n-1)$ is the ultimate variance we want, the higher lag covariances should all be zero. Let's check them out to be sure.

Case: $j = i+1$ (More generally $|i-j| = 1$)

$$\begin{aligned}
\mathsf{E}[I_i I_{i+1}] &= P(X_i \text{ starts a run and } X_{i+1} \text{ starts a run}) \\
&= P(X_i \text{ starts a run above the mean and } X_j \text{ starts a run above the mean}) \\
&\quad + P(X_i \text{ starts a run above the mean and } X_j \text{ starts a run below the mean}) \\
&\quad + P(X_i \text{ starts a run below the mean and } X_j \text{ starts a run above the mean}) \\
&\quad + P(X_i \text{ starts a run below the mean and } X_j \text{ starts a run above the mean})
\end{aligned}$$

The first and fourth probabilities here are zero and the other two are the same. (Again, if you are not convinced, compute each one separately. Thus, we have

$$\begin{aligned}
\mathsf{E}[I_i I_{i+1}] &= 2 \cdot P(X_i \text{ starts a run above and } X_j \text{ starts a run below}) \\
&= 2 \cdot P(X_{i-1} < 0.5, X_i > 0.5, X_{i+1} < 0.5) \\
&\overset{indep}{=} 2 \cdot P(X_{i-1} < 0.5) \cdot P(X_i > 0.5) \cdot P(X_{i+1} < 0.5) \\
&\overset{unif}{=} 2 \cdot \left(\tfrac{1}{2}\right) \cdot \left(\tfrac{1}{2}\right) \cdot \left(\tfrac{1}{2}\right) = \tfrac{1}{4}.
\end{aligned}$$

So,

$$Cov(I_i, I_{i+1}) = \mathsf{E}[I_i I_{i+1}] - \mathsf{E}[I_i]\mathsf{E}[I_{i+1}] = \frac{1}{4} - \left(\frac{1}{2}\right)^2 = 0.$$

Equivalently, we have shown that

$$P(X_i \text{ starts a run and } X_{i+1} \text{ starts a run}) = P(X_i \text{ starts a run}) \cdot P(X_{i+1} \text{ starts a run})$$

which tells us that the events here are independent.

It is not difficult to verify that this independence in run starting continues at higher lags. □

When considering runs above and below the mean, the total number of runs $R_n$ is 1 plus a sum of $n-1$ iid Bernoulli random variables. So, by the standard CLT, $R_n$ is approximately normally distributed for large $n$. Thus, a rough 95% test for independence in the case of uniformity is given by the following.

**Runs Above/Below the Mean Test for Independence**
For a sample of some "large" size $n$, from the $unif(0,1)$ distribution proceed as follows.

- Count the total number of runs in the sample. Call it $R_n$.
- Compute $\mu_n = \mathsf{E}[R_n]$ and $\sigma_n^2 = Var[R_n]$ as defined by Theorem 1.2.2.
- Compute
$$Z_n = \frac{R_n - \mu_n}{\sigma_n}.$$
- If $-1.96 \leq Z_n < 1.96$ we say that this sample has "passed" the runs up and down test for independence!

**Example:**

For our sample of size $n = 100,000$, we observed $R_{100,000} = 49,908$. We have that

$$\mu_{100,000} = \frac{100,000+1}{2} = 50,000.5 \qquad \text{and} \qquad \sigma_n^2 = \frac{100,000-1}{4} = 24999.75.$$

Thus we have the test statistic

$$Z_n = \frac{R_n - \mu_n}{\sigma_n} = \frac{49908 - 50000.5}{\sqrt{24999.75}} \approx -0.594.$$

Since this is in the interval $(-1.96, 1.96)$, our RNG has passed the runs above/below the mean test for independence. Equivalently, plugging $-1.96$ and $1.96$ in for $Z_n$, we would pass for any $R_n$ in the interval $(49,691, 50,310)$. □

### 1.2.3 Length of Runs

We have talked about two types of runs so far while ignoring the length of these runs. It is finally time to consider the length of these runs. We can look at both types of runs (up/down or above/below the mean) but it is most common to consider the length of up/down runs. This is what we will do here.

Once again, the first 35 values produced by our RNG are as follows.

| | | | | | | |
|---|---|---|---|---|---|---|
| 0.536839 | 0.501837 | 0.945124 | 0.066075 | 0.926468 | 0.220971 | 0.975456 |
| 0.332541 | 0.594702 | 0.025603 | 0.644262 | 0.713022 | 0.858687 | 0.211142 |
| 0.683887 | 0.158859 | 0.555520 | 0.343992 | 0.539392 | 0.955368 | 0.162900 |
| 0.925698 | 0.723009 | 0.894404 | 0.373177 | 0.323461 | 0.117826 | 0.945908 |
| 0.097929 | 0.931847 | 0.644134 | 0.890236 | 0.968813 | 0.452754 | 0.225053 |

The first value in the sample always starts a run. In this case it is starting a short-lived run down of length 1. (Since this run is comprised of two sampled values, you might want to call it a run of length 2. Everything that follows would need to be adjusted. It is more common to call this a run of length 1. Imagine drawing arcs from number to number and counting the number of arcs.)

So, we start with a run down of length 1, followed by a run up of length 1, a run down of length 1, et cetera. For this test, we actually don't care whether the runs go up or down Within these 35 values there are

| | |
|---|---|
| 22 | runs of length 1 |
| 3 | runs of length 2 |
| 2 | runs of length 3 |
| 0 | runs of higher lengths |

(Make sure you can count them!)

In the full sample, we observed up/down runs up to length 7. In a sample of size $100,000$, we could observe a run of length $99,999$ at most. (That would not bode well for independence though!) What should we expect to observe?

**Theorem 1.2.3**
*Let $X_1, X_2, \ldots, X_n$ be a random sample of size $n$ from any distribution.*

*For $i = 1, 2, \ldots, n-2$, the expected number of runs (under the assumption of independence) of length $i$ is*

$$E_i := \frac{2}{(i+3)!}\left[n(i^2+3i+1) - (i^3+3i^2-i-4))\right]$$

*and*

$$E_{n-1} = \frac{2}{n!}.$$

Proof Let

$$I_j^{(i)} = \begin{cases} 1 & , \text{ if } X_j \text{ starts a run of length } i \\ 0 & , \text{ otherwise.} \end{cases}$$

Then the total number of runs of length $i$ is

$$R_n^{(i)} := \sum_{j=1}^{n-1} I_j^{(i)}$$

and

$$E_i = \mathsf{E}[R_n^{(i)}] = \sum_{j=1}^{n-1} \mathsf{E}[I_j^{(i)}].$$

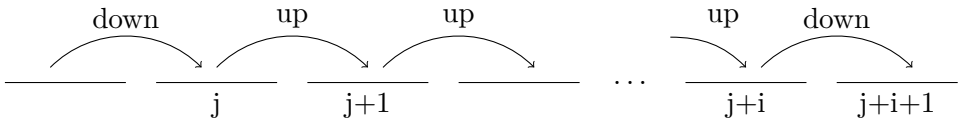
Since $X_1$ acts a little differently than the other $X_j$, we will first consider $I_j$ for $j = 2, 3, \ldots, n-1$. Also, note that $X_{n-1}$, for example, can not start a run of length 2 or more. In general, $X_j$ can not start a run of length greater than $n - j$.

So, for $j = 2, 3, \ldots, n-1$,

$$I_j^{(i)} = 0 \qquad \text{for } i = n-j+1, n-j+2, \ldots, n-1.$$

For $i = 1, 2, \ldots, n-j$ (and $j > 1$) we have

$$\begin{aligned} \mathsf{E}[I_j^{(i)}] &= P(X_j \text{ starts a run of length } i) \\ &= 2 \cdot P(X_j \text{ starts a run } \mathbf{up} \text{ of length } i) \end{aligned}$$




So,

$$\mathsf{E}[I_j^{(i)}] = 2 \cdot P(X_j < X_{j-1}, X_{j+1} > X_j, \ldots, X_{j+i} > X_{j+i-1}, X_{j+i+1} < X_{j+i})$$

Conditioning on the values of $X_j, X_{j+1}, \ldots, X_{j+i}$ gives

$$\begin{aligned} &\mathsf{E}[I_j^{(i)}] \\ &= 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{x_{j+i-1}} \cdots \int_{-\infty}^{x_{j+2}} \int_{-\infty}^{x_{j+1}} P(x_j < X_{j-1}, X_{j+i+1} < x_{j+i}) f(x_j) f(x_{j+1}) \cdots f(x_{j+i})\, dx_j\, dx_{j+1} \cdots dx_{j+i} \\ &= 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{x_{j+i-1}} \cdots \int_{-\infty}^{x_{j+2}} \int_{-\infty}^{x_{j+1}} [1 - F(x_j)]\, F(x_{j+i})\, f(x_j) f(x_{j+1}) \cdots f(x_{j+i})\, dx_j\, dx_{j+1} \cdots dx_{j+i} \\ &= 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{x_{j+i-1}} \cdots \int_{-\infty}^{x_{j+3}} \int_{-\infty}^{x_{j+2}} \left[ F(x_{j+1}) - \frac{1}{2} F^2(x_{j+1}) \right] F(x_{j+i}) f(x_{j+1}) \cdots f(x_{j+i})\; dx_{j+1} \cdots dx_{j+i} \\ &= 2 \cdot \int_{-\infty}^{\infty} \int_{-\infty}^{x_{j+i-1}} \cdots \int_{-\infty}^{x_{j+4}} \int_{-\infty}^{x_{j+3}} \left[ \frac{1}{2} F^2(x_{j+2}) - \frac{1}{2 \cdot 3} F^3(x_{j+2}) \right] F(x_{j+i}) f(x_{j+2}) \cdots f(x_{j+i})\; dx_{j+2} \cdots dx_{j+i} \\ &\vdots \\ &= 2 \cdot \textstyle\int_{-\infty}^{\infty} \left[ \frac{1}{i!} F^i(x_{j+i}) - \frac{1}{(i+1)!} F^{i+1}(x_{j+i}) \right] F(x_{j+i}) f(x_{j+i})\, dx_{j+i} \\ &= 2 \cdot \textstyle\int_{-\infty}^{\infty} \left[ \frac{1}{i!} F^{i+1}(x_{j+i}) - \frac{1}{(i+1)!} F^{i+2}(x_{j+i}) \right] f(x_{j+i})\, dx_{j+i} \\ &= 2 \cdot \left[ \frac{1}{(i+2)i!} - \frac{1}{(i+3)(i+1)!} \right] = \frac{2(i^2+3i+1)}{(i+3)!} \end{aligned}$$

Look familiar? This is a major piece of the result stated in the Theorem. Finishing it will require us to compute $\mathsf{E}[I_1^{(i)}]$ and to add up the appropriate pieces keeping in mind that $I_j^{(i)} = 0$ for $j = 2, 3, \ldots, n-1$ and $i = n-j+1, n-j+2, \ldots, n-1$. Also, I'll leave $E_{n-1}$ for

an exercise as well! □

So, what will we do with this result? In the sample, we will observe a certain number of runs of length 1, of length 2, et cetera. We now know what to expect for each category. Are our observations and expectations "close enough" for each category? This sounds like a classic $\chi^2$ goodness of fit test. (See Appendix B, Section B.1 for the MathStat discussion.)

The $\chi^2$ goodness of fit test is usually used to test whether observations that can fall into one of $k$ categories are falling into those categories in certain hypothesized proportions. It might be stated as a null and alternate hypothesis such as

$$\begin{array}{lcl} H_0 & : & p_1 = p_{1,0},\ p_2 = p_{2,0},\ \ldots,\ p_k = p_{k,0} \\ H_1 & : & \text{not } H_0 \end{array}$$

for some fixed and specified probabilities $p_{1,0}, p_{2,0}, \ldots, p_{k,0}$ that add up to 1.

To perform the test, one computes the observed number of outcomes in each category ($O_i$) and the expected number in each category under the assumption that $H_0$ is true ($E_i = np_{i,0}$), and then the test statistic

$$W := \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}.$$

In Appendix B, Section B.1 we see that, if $H_0$ is true, this statistic should behave, for large enough $n$, roughly like a $\chi^2(k-1)$ random variable. In particular, $n$ should be taken **"large enough"** so that we have a minimum expected number in each of the $k$ categories. A **rule of thumb** is to require that $E_i = np_{i,0} \geq 5$ for $i = 1, 2, \ldots, k$.

It makes sense that if are observations are far from what we expect, this test statistic will be large. That is, we should reject the null hypothesis if $W$ is larger than a critical value from the $\chi^2(k-1)$ distribution.

**Length of Runs (Up/Down) Test for Independence**
For a sample of some "large" size $n$, proceed as follows.

- Count the total number of observed runs of each length: $O_1, O_2, \ldots$
- Compute the expected number of runs of each length: $E_1 E_2, \ldots$
- The expected numbers will be shrinking. Ensure at least 5 expected in each run length group by putting all runs longer than some length into one category for a total of $k$ categories. (See example.)
- Compute the test statistic
$$W := \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}.$$
- If the test statistic is greater than an upper tail $\chi^2(k-1)$ critical value at a desired level of significance, reject the null hypothesis of independence which is reflected in the expected number in each category computed under the assumption of independence.

**Example:**

For our sample of size $n = 100,000$, we observed up/down runs of lengths 1 through 7. In particular, we observed the following.

| Run Length | Observed | Expected |
|---|---|---|
| 1 | 41,721 | 41666.75 |
| 2 | 18,386 | 18,333.10 |
| 3 | 5,178 | 5,277.65 |
| 4 | 1,145 | 1,150.75 |
| 5 | 220 | 203.36 |
| 6 | 45 | 30.31 |
| 7 | 3 | 3.91 |
| 8 | 0 | 0.45 |
| ⋮ | ⋮ | ⋮ |

In order to ensure an expected value of at least 5 in each category, we will consider six categories

| Run Length | Observed | Expected |
|---|---|---|
| 1 | 41,721 | 41666.75 |
| 2 | 18,386 | 18,333.10 |
| 3 | 5,178 | 5,277.65 |
| 4 | 1,145 | 1,150.75 |
| 5 | 220 | 203.36 |
| $\geq$ 6 | 45 | 34.72 |

Recall that, in Section 1.2.1, we determined that the expected number of up/down runs was

$$\mathsf{E}[R_{100,000}] = \frac{2n-1}{3} = \frac{2(100000)-1}{3} = 66,666.33.$$

Thus, the expected number of runs with length greater than or equal to 6 is given by

$$66,666.33 - (41,666.75 + 18,333.10 + 5.277.65 + 1,150.75 + 203.36) = 34.72.$$

The test statistic is

$$W = \sum_{i=1}^{6} \frac{(O_i - E_i)^2}{E_i} = \frac{(41721 - 41666.75)^2}{41666.75} + \cdots + \frac{(45 - 34.72)^{22}}{34.72} \approx 6.54.$$

For a test of independence with $\alpha = 0.05$ level of significance, the upper tail 5% capturing critical value for the $\chi^2(5)$ distribution is given by 11.0705. Since $W$ is less than this value, we are not up in the rejection region and so we fail to reject the null hypothesis of independence. In other words, our RNG has passed this test. □

### 1.2.4 Autocorrelation Test

Uncorrelated random variables may or may not be independent. However, independent random variables are always uncorrelated. So, if we estimate the correlation between values in our sample and decide that it is significantly far away from zero we may conclude that the values are not independent. (On the other hand, if it is close to zero it doesn't tell us anything.)

We will consider testing the hypotheses

$$\begin{aligned} H_0 &: \text{the values are independent} \\ H_1 &: \text{the values are not independent.} \end{aligned}$$

As with any hypothesis test, we won't be able to prove that $H_0$ is true. We will assume it is true and look for strong evidence to reject it.

Recall that the covariance between random variables $X$ and $Y$ is defined as

$$Cov(X, Y) = \mathsf{E}[(X - \mu_x)(Y - \mu_y)]$$

where $\mu_X = \mathsf{E}[X]$ and $\mu_y = \mathsf{E}[Y]$. It is easy to show that we can write

$$Cov(X,Y) = \mathsf{E}[XY] - \mathsf{E}[X]\mathsf{E}[Y],$$

and therefore easy to see that if $X$ and $Y$ are independent, the covariance is zero.

The correlation between $X$ and $Y$ is a standardized version of covariance defined as

$$Corr(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var[X]\,Var[Y]}}$$

which will also be zero if $X$ and $Y$ are independent.

For our sampled values $X_1, X_2, \ldots, X_n$, presumably from the $unif(0,1)$ distribution, there are several correlations we can look at. We can look at the correlation between $X_1$ and $X_2$, for example. While we will only have one observation of each, under the assumption that we have identically distributed values, we can consider $X_2$ and $X_3$ to be another observation of a neighboring pair, for example. Alternatively, we might want to look at the "lag 5" correlation between $X_1$ and $X_6$. In this case, under the assumption that we have a random sample, $X_2$ and $X_7$ would be another "copy" of this pair.

In general, we denote/define the **lag j (auto)correlation** to be

$$\rho_j := Corr(X_i, X_{i+j})$$

for any $i$. Note that

$$\rho_j = \frac{Cov(X_i,X_{i+j})}{\sqrt{Var[X_i]\,Var[X_{i+j}]}} = \frac{Cov(X_1,X_{1+j})}{\sqrt{Var[X_1]\,Var[X_1]}} = \frac{Cov(X_1,X_{1+j})}{Var[X_1]}$$

under the assumption that we are looking at identically distributed values.

So,

$$\rho_j = \frac{\mathsf{E}[X_1X_{1+j}] - \mathsf{E}[X_1]\mathsf{E}[X_{1+j}]}{Var[X_1]}$$

For the $unif(0,1)$ distribution, $\mathsf{E}[X_i] = 1/2$ and $Var[X_i] = 1/12$. Thus, we have

$$\rho_j = 12\mathsf{E}[X_1X_{1+j}] - 3.$$

Note that we are so far working under the assumption that all of the $X_i$ are uniform and, in particular, identically distributed. We have not used independence. If the $X_i$ are independent, this lag $j$ (auto)correlation is zero. Let's estimate it from the sample by estimating the expectation with a sample mean.

$$\widehat{\rho_j} = 12\left[\frac{\sum_{i=1}^{h} X_iX_{i+j}}{h}\right] - 3.$$

Here, $h$ is taken to be the largest integer such that we don't go past the end of the sample.

We will reject the null hypothesis of independence if this value is “far" (too large or too small) away from zero. Under the assumption that $H_0$ is true and the $X_i$ are independent, it is easy to verify that

$$\mathsf{E}[\widehat{\rho_j}] = 0.$$

In order to figure out what “far" means, we also need to compute the variance.

$$\begin{aligned} Var[\widehat{\rho_j}] &= Var\left[12\left[\frac{\sum_{i=1}^{h} X_i X_{i+j}}{h}\right] - 3\right] = Var\left[12\left[\frac{\sum_{i=1}^{h} X_i X_{i+j}}{h}\right]\right] \\ &= \frac{144}{h^2} Var\left[\sum_{i=1}^{h} X_i X_{i+j}\right]. \end{aligned}$$

Although we are now assuming independence of the $X_i$, the terms in the sum are not independent. For example, if $j = 1$, the first two terms are $X_1X_2$ and $X_2X_3$ which both contain $X_2$. So, we can not pull the sum out of the variance. Instead, we again use the fact that

$$Var\left[\sum_{i=1}^{h} X_i X_{i+j}\right] = Cov\left(\sum_{i=1}^{h} X_i X_{i+j}, \sum_{k=1}^{h} X_k X_{k+j}\right) = \sum_{i=1}^{h}\sum_{k=1}^{h} Cov(X_i X_{i+j}, X_k X_{k+j}).$$

Now,

$$Cov(X_i X_{i+j}, X_k X_{k+j}) = \mathsf{E}[X_i X_{i+j} X_k X_{k+j}] - \mathsf{E}\left[X_i X_{i+j}\right] \cdot \mathsf{E}[X_k X_{k+j}].$$

Since $j \geq 1$, $X_i$ and $X_{i+j}$ are independent. We can factor those final two expectations. For the first expectation $\mathsf{E}[X_i X_{i+j} X_k X_{k+j}]$, we have to consider several different cases like

$$\mathsf{E}[X1X_2X_3X_4], \qquad \mathsf{E}[X_1^2 X_2^2], \qquad \text{and } \mathsf{E}[X_1^3 X_2].$$

In the end, we will get that

$$Var[\widehat{\rho_j}] = \frac{13n + 7}{(n+1)^2}.$$

The finite range dependence version of the CLT will give us that, for large $h$, $\widehat{\rho_h}$ is approximately normally distributed. We can use its mean and variance to standardize it to an approximate $N(0, 1)$ and use normal critical values for our hypothesis test.

**Autocorrelation Test for Independence**
For a sample of some "large" size $n$, proceed as follows. For some lag $j \geq 1$,

- Compute $\widehat{p_j}$.
- Compute $\sigma_n^2 = Var[\widehat{\rho_j}]$.
- Compute
$$Z_n = \frac{\widehat{\rho_j}}{\sigma_n}.$$
- If $-1.96 \leq Z_n \leq 1.96$, we say that the sample has "passed" the autocorrelation test for independence at lag $j$.

People usually perform this test for some selection of small lags from $j = 1$ up to $j =$ "something". Note that for a very large lag (relative to $n$) the sample correlation will consist of few terms and the CLT will not hold.

## 1.3 Tests for Uniformity

We have covered several tests for independence which are used to evaluate RNGs. Some of them relied on the sample being uniform and some did not. The tests that did rely on uniformity can be adjusted to test for independence for a sample from non-uniform distributions as well. In this Section, we will discuss three tests for uniformity that can also be adjusted to test for different distributions. I have found the Kolmogorov-Smirnov test, given in Section 1.3.3 particularly useful in several circumstances.

### 1.3.1 $\chi^2$ Test

Suppose that we have a sample $X_1, X_2, \ldots, X_n$ of values in $(0, 1)$ that we would like to test for uniformity. If we break up the unit interval into, say 5 "bins" from 0 to 0.20, from 0.20 to 0.40, et cetera, then under uniformity we would expect $n/5$ values in each bin. We can run a standard $\chi^2$ goodness of fit test to compare our observed values in each bin with these expected values just as we did in Section 1.2.3. The major problem with this test is in choosing the number of bins to use. The conclusion (the data come from a given distribution versus not) is going to depend on the number of bins used and there is, as far as I know, no good method for choosing this number. If there were, it would likely be distribution dependent. (Note that, in order to use the $\chi^2$ goodness of fit test you should at least expect 5 values to fall in each bin!)

**$\chi^2$ Test for Uniformity**
For a sample of some "large" size $n$, of numbers between 0 and 1, proceed as follows.

- Choose an integer $k \geq 1$ for a number of bins.
- For $i = 1, 2, \ldots, k$, let $O_i$ be the number of values in the sample that fall in bin $i$.
- For $i = 1, 2, \ldots, k$, the number expected in bin $k$, under the assumption of uniformity is $E_i = n/k$.
- Compute the test statistic

$$W = \sum_{i=1}^{6} \frac{(O_i - E_i)^2}{E_i}$$

.

- If $W$ is larger than a $\chi^2$ critical value, reject the null hypothesis of independence.

**Example:** For our sample of size $100,000$, we broke up the interval from 0 to 1 into $k = 20$ equal width bins. We observed $4,962$ values in the first bin where we expected $5,000$ values. We observed $4,989$ in the second bin where we also expect $5,000$ values. Moving along, we observed $5,038$ in the last bin where, you guessed it, we expect $5,000$ values. The test statistic is

$$W = \frac{(4962 - 5000)^2}{5000} + \frac{(4989 - 5000)^2}{5000} + \cdots + \frac{(5038 - 5000)^2}{5000} \approx 5.77.$$

We compare this to the 5% upper tail critical value for the $\chi^2(19)$ distribution which is

$$\chi^2 0.05(19) = 30.144.$$

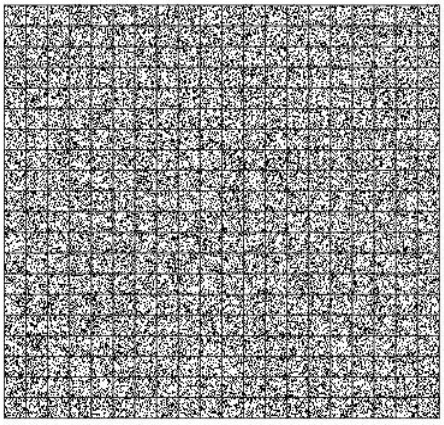
Since $W < 30.144$, we fail to reject the null hypothesis for uniformity. In other words, our RNG has passed our first test for uniformity. □

### 1.3.2 Serial Test

This test actually serves as a test for **both** uniformity and a "local" independence. Here, we bunch up the data into $m$-dimensional vectors and consider their distribution in $(0, 1)^m$. We do this by breaking up $(0, 1)^m$ into $k$ (user chosen) bins on each axis and counting the number of points that fall in each bin. $k$ is usually chosen to be quite small. Indeed, using $k$ $m$-dimensional bins will give us a total of $k^m$ bins, and we should at least have $(n/m)/k^m \geq 5$.

This test, with $k = 3$, may have caught the problem with the IBM RANDU generator if the

Figure 1.3: Serial Test Bins



bins were “small enough but not too small". As with the previous test, results are dependent on user chosen parameters. So, while I wouldn't advise this being your sole test for uniformity, it can lend support as one of a battery of tests that give the same conclusion.

**Example:**

For our sample, I used a $20 \times 20$ grid on $\mathbb{R}^2$. The $100,000$ values are shown in their bins in Figure 1.3.

Of the $50,000$ two-dimensional points, we expect 125 in each cell under the assumption of uniformity. The test statistic is

$$W = \sum_{i=1}^{400} \frac{(O_i - E_i)^2}{E_i} \approx 422.4.$$

The $\chi^2$ critical value is

$$\chi^2_{0.05}(399) = 446.5742.$$

Since our test statistic is not above the critical value, we fail to reject the null hypothesis of independent uniforms. Again, our RNG passed!

Here are a few more test statistics, critical values, and conclusions based on different binnings.

| k | W | $\chi^2_{0.05}(k-1)$ | Independence? |
|---|---|---|---|
| 2 | 10.87 | 7.8147 | Failed |
| 6 | 42.26 | 49.8019 | Passed |
| 10 | 122.51 | 123.2252 | Passed |

### 1.3.3 Kolmogorov-Smirnov Test

The Kolmogorov-Smirnov test is something I use quite often (unlike the other tests discussed so far) when I want to establish whether or not a random sample is coming from a given univariate distribution. The idea is to compare the cumulative distribution function (cdf) of the desired target distribution with the *empirical cdf* produced from the sample.

**The Empirical CDF**

The cdf for a random variable $X$, usually denoted by a captial $F$, is defined as the cumulative probability

$$F(x) := P(X \leq x).$$

Suppose we simulate a random sample of size $n = 6$ from the exponential distribution with rate $\lambda = 0.6$ and get the values

$$1.46, \quad 4.26, \quad 0.83, \quad 1.67, \quad 0.18, \quad 3.30$$

As far as we can tell, based on this sample, the probability that $X$ is less than or equal to 2, for example, is $2/3$ since 4 out of 6 of our observations are less then or equal to 2. Continuing along these lines, we can estimate the cdf at any $x$ using the step function

$$\widehat{F}(x) = \begin{cases} 0 & , \quad x < 0.18 \\ 1/6 & , \quad 0.18 \leq x < 0.83 \\ 2/6 & , \quad 0.83 \leq x < 1.46 \\ 3/6 & , \quad 1.46 \leq x < 1.67 \\ 4/6 & , \quad 1.67 \leq x < 3.30 \\ 5/6 & , \quad 3.30 \leq x < 4.26. \\ 1 & , \quad x \geq 4.26 \end{cases}$$

This empirical cdf is plotted along with the true cdf ($F(x) = 1 - e^{-0.6x}$ for $x > 0$) in Figure 1.4.

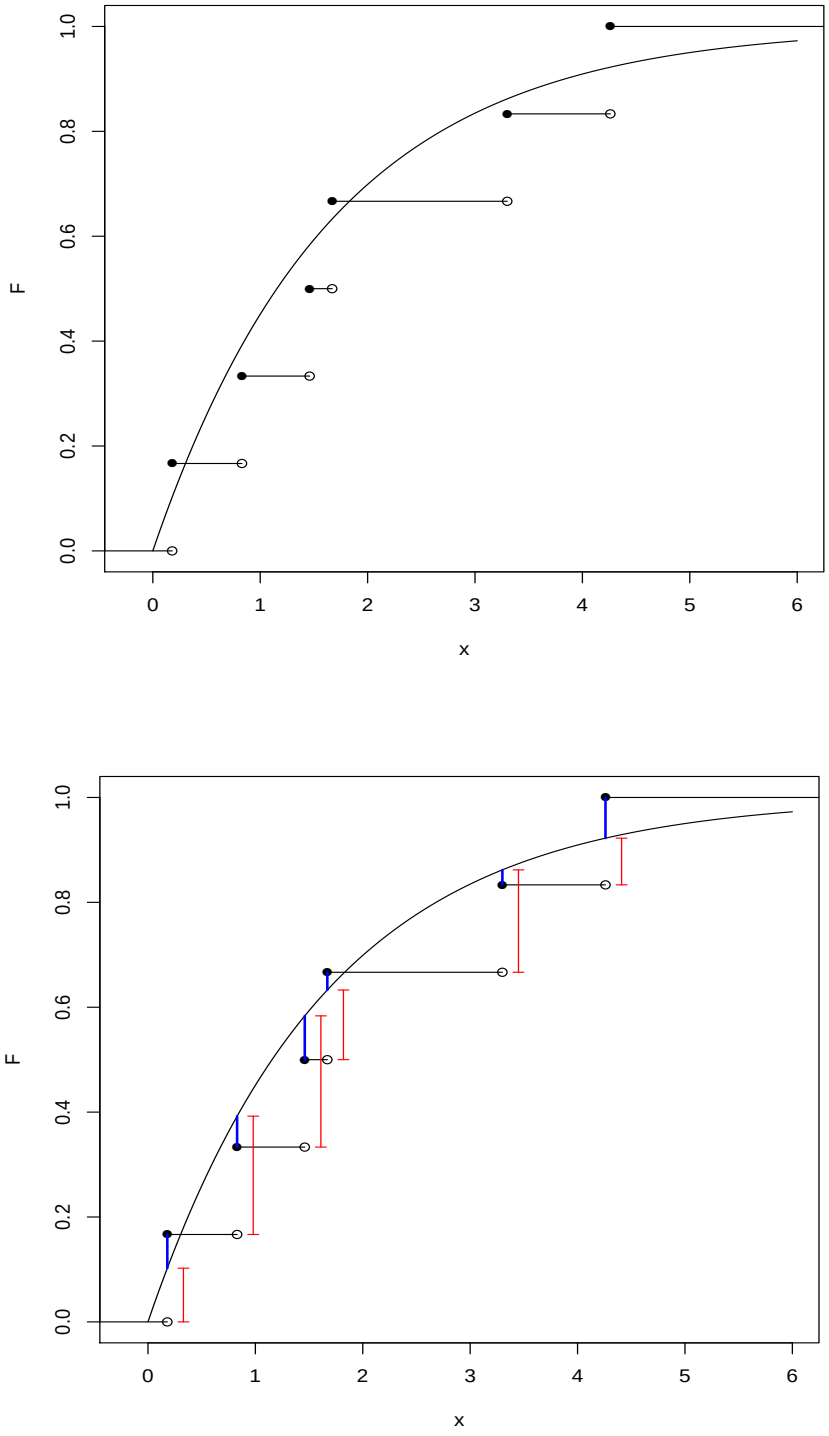
For a general sample, $X_1, X_2, \ldots, X_n$ of size $n$, we will denote the emprical cdf by $\widehat{F}_n$ and define it by

$$\widehat{F}_n(x) = \frac{\#\text{ of } X_i \text{ in the sample} \leq x}{n}.$$

If $X_1, X_2, \ldots, X_n$ are coming from a distribution with true cdf $F$, we would expect the distance

$$D = D_n := \sup_x |\widehat{F}_n(x) - F(x)|$$

Figure 1.4: Empirical CDF, True CDF and Distances



to be “small" and to get “smaller" as we make $n$ larger.

For our particular six point example, we have

$$D_6 \approx 0.2502.$$

In general we have the following.

**The Kolmogorov-Smirnov Statistic**
For an ordered sample $X_{(1)}, X_{(2)}, \ldots, X_{(n)}$, define

$$D_n = \max\{D_n^+, D_n^-\}$$

where

- $D_n^+ = \max\limits_{1 \le i \le n} \left| \dfrac{i}{n} - F(X_{(i)}) \right|$
- $D_n^- = \max\limits_{1 \le i \le n} \left| F(X_{(i)}) - \dfrac{i-1}{n} \right|$

(Note: This is assuming non-repeating values in the sample, otherwise you will have to take into account jumps of size $j/n$, for $j > 1$. If you have a large sample and just a few repeating values due to measurement precision, this formula is sometimes used anyway as an approximation to the K-S statistic.)

We will reject the hypothesis that the sample comes from the proposed distribution if the empirical cdf is too far from the true cdf of the proposed distribution. That is, we reject that we have a sample from the proposed distribution if $D_n$ is too “large". How large is large?

In the 1930's, Kolmogorov [2] showed that

$$\lim_{n \to \infty} P(\sqrt{n} D_n \le t) = 1 - 2 \sum_{i=1}^{\infty} (-1)^{i-1} e^{-2i^2 t^2}.$$

So, for large sample sizes, you could assume that

$$P(\sqrt{n} D_n \le t) \approx 1 - 2 \sum_{i=1}^{\infty} (-1)^{i-1} e^{-2i^2 t^2}$$

and try to find the value of $t$ that makes the right-hand side equal to $1 - \alpha$ for an $\alpha$ level test. This is not a nice thing to work with and can not be solved in closed form.

Over the years, many people (summarized nicely in [3]) have worked on numerical approximations to exact values for small samples that hold for certain distributions and approximations for both small and large samples from unknown distributions. In the following table, we give commonly used [1] approximate critical values for large samples ($n > 40$) from general distributions.

| $\alpha$ | 0.20 | 0.10 | 0.05 | 0.02 | 0.01 |
|---|---|---|---|---|---|
| c.v | $1.0730/\sqrt{n}$ | $1.2239/\sqrt{n}$ | $1.3581/\sqrt{n}$ | $1.5174/\sqrt{n}$ | $1.6276/\sqrt{n}$ |

**Small Sample Example:**

Recall our six point sample from the exponential distribution with rate 0.6. We wish to test the hypotheses

$$\begin{array}{rcl} H_0 & : & \text{The sample came from the } exp(rate = 0.6) \text{ distribution} \\ H_1 & : & \text{Not } H_0. \end{array}$$

Our K-S statistic was

$$D_6 \approx 0.2502.$$

Since this represents a maximum absolute distance between the empirical and true cdfs, we will reject $H_0$ in favor of $H_1$ if this number is too large.

From a K-S table (for one-sample tests), the $\alpha = 0.05$ level critical value is 0.51926. Since our statistic did not exceed this value, we fail to reject $H_0$ at a 0.05 level of significance. The data does not suggest that the exponential hypothesis is false. □

**Large Sample Example:**

Returning to our $100,000$ values produced by the "Acme random number generator", we wish to test the hypotheses

$$\begin{array}{rcl} H_0 & : & \text{The sample came from the } unif(0,1) \text{ distribution} \\ H_1 & : & \text{Not } H_0. \end{array}$$

The K-S statistic in this case turns out to be

$$D_{100,000} \approx 0.00152392.$$

An approximate level 0.05 critical value is given by

$$\frac{1.3581}{\sqrt{100000}} \approx 0.004294689.$$

Since our test statistic does not exceed the critical value, we will not reject $H_0$. We have passed the Kolmogorov-Smirnov test for uniformity! □

## Homework Problems

1. Use your favorite software's RNG to produce your own sample of $100,000$ values that are trying to be independent and uniformly distributed on $(0,1)$. Perform all tests from this Chapter on your sample. Are you satisfied with your RNG? (Note: Even with a perfectly iid $unif(0,1)$ sample, when using a 0.05 level of significance, you should

expect to reject the hypotheses of uniformity and independence 5% of the time... so don't freak out if you don't pass a test!)

2. Explore the accuracy of the Kolmogorov-Smirnov critical values given in the table on page 29 via simulation. Feel free to use built-in functions for simulating samples from various distributions for large values of $n$. (In the next Chapter we will talk about simulating from various common distributions "from scratch".)

Chapter 2

# Simulating from Some Common Univariate Distributions

Suppose that $X$ is a random variable that takes on the values 0, 1, and 2, with probabilities 1/8, 1/3, and 13/24, respectively.

To simulate $X$, we want a procedure that will produce a list of numbers that we will call **realizations** of $X$. It is perfectly legitimate to simulate one or two values. However, in order to "see the probabilities" you will need to simulate many values. After producing a long list of numbers you should see

- 0 approximately 1/8 of the time,
- 1 approximately 1/3 of the time, and
- 2 approximately 13/24 of the time.

For a continuous distribution, such as the beloved exponential distribution with rate 0.2, we have (for example)

$$\int_0^1 0.2e^{-0.2x}\,dx = 1 - e^{-0.2} \approx 0.1813$$

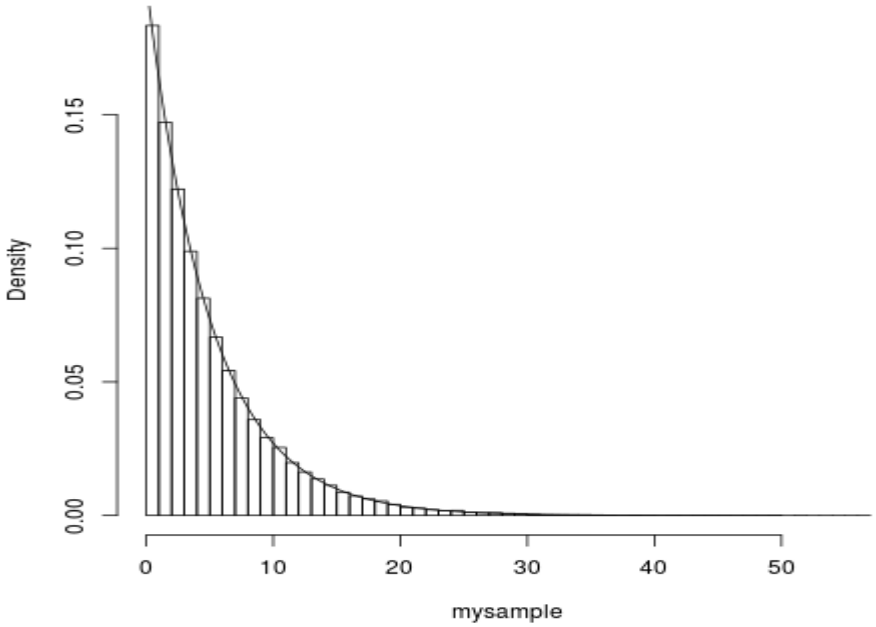
Thus, if we are simulating from this distribution, after producing many realizations the proportion of observed values less than 1 should be approximately 0.1813. In general, we want to match the proportion of values in any interval $(a, b)$ with the value

$$\int_a^b 0.2e^{-0.2x}\,dx = e^{-0.2a} - e^{-0.2b}.$$

The easiest way to do this is visually with a histogram and the true density superimposed as in Figure 2.1.

When making such a histogram, choose your bin widths so that the graph is neither too "chunky" nor too "spikey". My most commonly used bin width for univariate distributions is 0.1. However, since the sample represented in Figure 2.1 had values in the upper 50's, using 0.1 would have given way too many bins, resulting a really spikey histogram that would appear to be made up of vertical lines rather than rectangles. In the end I used a bin width of 1.

Figure 2.1: 100,000 Draws from the Exponential Distribution with Rate $\lambda = 0.2$



## 2.1 Generating Discrete Random Variables

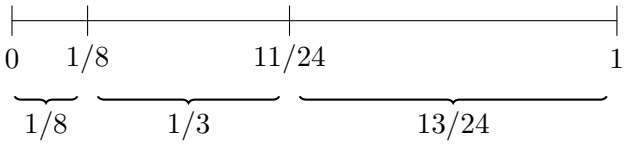
### 2.1.1 The Basic Finite Discrete Case

Let us return to our three point example of a random variable $X$ with probability density (mass) function given by

| x | 0 | 1 | 2 |
|---|---|---|---|
| P(X=x) | 1/8 | 1/3 | 13/24 |

Consider the rather arbitrary interval from 1/2 to 5/8. Since this interval is of length 1/8 a realization of a $unif(0,1)$ random variable will fall in this interval 1/8 of the time. Thus, if we want to simulate a random variable $X$ that takes on the value 0 one-eighth of the time, we could simulate a $unif(0,1)$ and assign $X$ to be 0 every time our uniform falls between 1/2 and 5/8.

There is no reason to be so weird about theses intervals. Let us chop up the unit interval into three consecutive non-overlapping pieces of lengths 1/8, 1/3, and 13/24.




Now grab a $unif(0,1)$ from your RNG. Call it $U$. Output

$$X = \begin{cases} 0 & , \text{ if } U < 1/8 \\ 1 & , \text{ if } 1/8 < U \leq 11/24 \\ 2 & , \text{ if } U \geq 11/24 \end{cases}$$

If you repeat this $n$ times, where $n$ is large, you will roughly get zero 1/8 of the time, one 1/3 of the time and two 13/24 of the time. You could do a quick and dirty (non-simultaneous) check that your proportions are in between $p \pm 2\sqrt{p(1-p)/n}$ for the three different values of $p$. You could also do a $\chi^2$-goodness of fit test to compare the observed number of 0's, 1's, and 2's with the expected numbers for a sample of size $n$.

### 2.1.2 The Basic Infinite Discrete Case

Suppose that $X$ is a discrete random variable, taking values in $\{0, 1, 2, \ldots\}$, with pdf

$$f(x) = (1-p)^x p \, I_{\{0,1,2,\ldots\}}(x) \tag{2.1}$$

for some parameter $0 \leq p \leq 1$.

Here, I am using the **indicator notation** defined, for a set $A$ as

$$I_A(x) = \begin{cases} 1 & , \text{ if } x \in A \\ 0 & , \text{ if } x \notin A. \end{cases}$$

You should recognize (2.1) as the pdf for a geometric random variable. As such, you might want to simulate it using the interpretation of the geometric random variable as the number of trials (or failures in this case) until (before) the first success in a experiment consisting of repeated independent trials of an experiment that can result in either success or failure on any one trial with $p$ being the probability of success on any one trial. We will do this in Section 2.1.3. Here, however, we will consider a generic algorithm which is essentially the algorithm in Section 2.1.1. The only difference is that we will not chop up the interval $[0, 1]$ a priori.

For our previous three point example, we would draw a $unif(0, 1)$. Call it $u$. If $u < 1/8$ we would output 0. Otherwise, if $u < 1/8 + 1/3$, we would output 1. Finally, if $u < 1/8 + 1/3 + 13/24$, we would output 2. (Put the equalities wherever you want!) Generalizing, the algorithm (pseudocode) for simulating one value from $f(x)$, assuming $x$ lives on $\{0, 1, 2, \ldots\}$ is then the following.

**Algorithm 1** Simulating from a discrete $f(x)$ on $\{0, 1, 2, \ldots\}$

```
u = unif(0,1)

x = 0
cdf = f(0)

start loop
    if (u <= cdf) then
      exit loop
    else
      x = x+1
      cdf = cdf + f(x)
    end if
end loop

output x
```

This algorithm can easily be modified for a different discrete domain and, as mentioned, will even work for pdfs with a finite support set.

### 2.1.3 Some Specific Distributions

The algorithm described in Section 2.1.2 can be used for any discrete (univariate) distribution. In this Section, we describe some distribution specific algorithms that you may or may not want to use instead.

#### 2.1.3.1 The Discrete Uniform

The discrete uniform distribution on a set such as $\{1, 2, \ldots, n\}$ assigns equal probability $1/n$ to each value. The pdf is

$$f(x) = \frac{1}{n} I_{\{1,2,\ldots,n\}}(x).$$

While one could chop up the interval $[0, 1]$ into $n$ equal subintervals ans proceed as in Section 2.1.1, one could also procede as follows.

1. Draw a continuous uniform $U \sim unif(0, 1)$ from a RNG.
2. Multiply the draw by $n$ to get a continuous uniform, $nU$, on $(0, n)$.
3. Use the ceiling function to uniformly map $nU$ to an integer in $\{1, 2, \ldots, n\}$.

Clearly this will give the desired result since $U$ is equally likely to be in the intervals $(0, 1)$, $(1, 2)$, $(2, 3)$, $\ldots$, $(n - 1, n)$ and the ceiling function will give the righ endpoints of those intervals.

#### 2.1.3.2 The Geometric, Binomial, and Negative Binomial

Each of these distributions has an interpretation as counting something in a sequence of trials of a "success" and "failure" experiment. In each case, we could perform that experiment and make the counts.

Consider a sequence of independent trials of an experiment of an experiment where each trial can result in either success ($S$) or failure ($F$). Further assume that the probability of success remains the same from trial to trial and is denoted by some $0 \leq p \leq 1$.

1. The **geometric random variable** on $\{0, 1, 2, \ldots\}$ counts the number of failures before the first success. It's pdf is

$$f(x) = P(X = x) = (1 - p)^x \cdot p \cdot I_{\{0,1,2,\ldots\}}(x).$$

2. The **geometric random variable** on $\{1, 2, 3, \ldots\}$ counts the number of trials up to and including the first success. It's pdf is

$$f(x) = P(X = x) = (1 - p)^{x-1} \cdot p \cdot I_{\{1,2,3,\ldots\}}(x).$$

3. The **negative binomial random variable** on $\{0, 1, 2, \ldots\}$ counts the number of failures before the $r$th success for some integer $r \geq 1$. It's pdf is

$$f(x) = P(X = x) = \left( \begin{array}{c} x + r - 1 \\ r \end{array} \right) p^r (1 - p)^x I_{\{0,1,2,\ldots\}}(x).$$

4. The **negative binomial random variable** on $\{r, r+1, r+2, \ldots\}$ counts the number of trials up to and including the $r$th success for some integer $r \geq 1$. It's pdf is

$$f(x) = P(X = x) = \left( \begin{array}{c} x-1 \\ r-1 \end{array} \right) p^r (1-p)^{x-r} \, I_{\{r,r+1,r+2,\ldots\}}(x).$$

5. Consider now a fixed number $n$ of trials. The **binomial random variable** counts the number of successes in $n$ trials. It's pdf is

$$f(x) = P(X = x) = \left( \begin{array}{c} n \\ x \end{array} \right) p^x (1-p)^{n-x} \, I_{\{0,1,2,\ldots,n\}}(x).$$

In all cases, we simulate the outcome of a single trial by drawing $U \text{unif}(0,1)$. Since $P(U \leq p) = p$, if $U \leq p$, we register a success, otherwise, we register a failure. All that is left to do is count what we want and to output that count!

### 2.1.4 Miscellaneous Univariate "Specialty Draws"

#### 2.1.4.1 Geometric

Let $U \sim unif(0,1)$.

**Claim:** For $0 < p < 1$, we have

$$\left\lceil \frac{\ln U}{\ln(1-p)} \right\rceil \sim geom_1(p).$$

Here, $\lceil \cdot \rceil$ and we are using "$geom_1(p)$" to denote the version of the geometric distribution that starts from 1 as opposed to the one that starts from 0. (The base of the logarithm is not important.)

PROOF Let $X = \left\lceil \frac{\ln U}{\ln(1-p)} \right\rceil$. First note that $\ln U$ and $\ln(1-p)$ are both negative, making the ratio positive. For fixed $p$, the ratio goes from $\infty$ to 0 as $U$ ranges from 0 to 1. So, the ceiling function will give us non-negative integers.

Note that

$$\begin{aligned} P(X = 1) &= P\left(0 < \tfrac{\ln U}{\ln(1-p)} \leq 1\right) \\ &= P\left(\ln(1-p) < \ln U \leq 0\right) \\ &= P(1-p < U \leq 1) \\ &\overset{unif}{=} 1 - (1-p) = p. \end{aligned}$$

We also have that

$$\begin{aligned} P(X=2) &= P\left(1 < \tfrac{\ln U}{\ln(1-p)} \leq 2\right) \\ &= P\left(2\ln(1-p) < \ln U \leq \ln(1-p)\right) \\ &= P\left(\ln(1-p)^2 < \ln U \leq \ln(1-p)\right) \\ &= P((1-p)^2 < U \leq (1-p)) \\ &\overset{unif}{=} (1-p) - (1-p)^2 = (1-p)[1-(1-p)] = (1-p)\cdot p. \end{aligned}$$

For a general integer $x \geq 1$, we have

$$\begin{aligned} P(X=x) &= P\left(x-1 < \tfrac{\ln U}{\ln(1-p)} \leq x\right) \\ &= P\left(x\ln(1-p) < \ln U \leq (x-1)\ln(1-p)\right) \\ &= P\left(\ln(1-p)^x < \ln U \leq \ln(1-p)^{x-1}\right) \\ &= P((1-p)^x < U \leq (1-p)^{x-1}) \\ &\overset{unif}{=} (1-p)^{x-1} - (1-p)^x = (1-p)^{x-1}[1-(1-p)] = (1-p)^{x-1}\cdot p. \end{aligned}$$

This is the desired $geom_1(p)$ pdf. □

If you want to simulate from the geometric distribution that starts from 0, you could use the fact that $X \sim geom_1(p)$ implies that $Y := X - 1 \sim geom_0(p)$. you could also adjust the function in the claim to directly simulate from the $geom_0$.

#### 2.1.4.2 Binomial

Let $X$ be a binomial random variable with parameters $n$ and $p$. We write $X \sim bin(n,p)$. Recall that the pdf for $X$ is

$$f(x) = P(X=x) = \left(\begin{array}{c} n \\ x \end{array}\right) p^x (1-p)^{n-x}\, I_{\{0,1,2,\ldots,n\}}(x).$$

If using the algorithms of Sections 2.1.1 and/or 2.1.2, you might want to take advantage of the recursive relationship

$$P(X=x+1) = \frac{n-x}{x+1}\cdot\frac{p}{1-p}\cdot P(X=x).$$

Here are two more relationships that won't gain you much but that I am including "for fun".

**Claim 1:** Let $G_1, G_2, \ldots$ be iid $geom_1(p)$ random variables.

Let $X$ be the smallest integer such that

$$\sum_{i=1}^{X+1} G_i > n.$$

Then $X \sim bin(n, p)$.

PROOF (Partial) First of all, note that, since each $G_i \geq 1$, $X$ will be at most $n$. That is, $X$ lives on $\{0, 1, 2, \ldots, n\}$.

We will now check a few cases. For all, we are using the fact that the cdf for each $G_i$ is $P(G_i \leq x) = 1 - (1-p)^x$.

$$P(X = 0) = P(G_1 > n) = (1-p)^n = \left( \begin{array}{c} n \\ 0 \end{array} \right) p^0 (1-p)^{n-0} \quad \surd$$

$$P(X = 1) = P(G_1 \leq n, G_1 + G_2 > n)$$

Conditioning on the value of $G_1$ gives

$$\begin{aligned} P(X = 1) &= P(G_1 \leq n, G_1 + G_2 > n) \\ &= \textstyle\sum_{x=1}^{\infty} P(G_1 \leq n, G_1 + G_2 > n | G_1 = x)(1-p)^{x-1} p \\ &= \textstyle\sum_{x=1}^{\infty} P(x \leq n, x + G_2 > n | G_1 = x)(1-p)^{x-1} p \\ &\overset{indep}{=} \textstyle\sum_{x=1}^{n} P(G_2 > n - x)(1-p)^{x-1} p \\ &= \textstyle\sum_{x=1}^{n} (1-p)^{n-x}(1-p)^{x-1} p \\ &= p \textstyle\sum_{x=1}^{n} (1-p)^{n-1} \\ &= p(1-p)^{n-1} \textstyle\sum_{x=1}^{n} 1 \\ &= p \cdot n(1-p)^{n-1} = \left( \begin{array}{c} n \\ 1 \end{array} \right) p^1 (1-p)^{n-1} \quad \surd \end{aligned}$$

Let's do one more...

$$P(X = 2) = P(G_1 \leq n, G_1 + G_2 \leq n, G_1 + G_2 + G_3 > n)$$

Conditioning on $G_1$ and $G_2$ gives

$$\begin{aligned}
P(X=2) &= P(G_1 \le n, G_1+G_2 \le n, G_1+G_2+G_3 > n) \\
&= \textstyle\sum_{x=1}^{\infty}\sum_{y=1}^{\infty} P(x \le n, x+y \le n, G_3 > n-x-y)\cdot(1-p)^{x-1}p(1-p)^{y-1}p \\
&= \textstyle\sum_{x=1}^{n}\sum_{y=1}^{n-x} P(G_3 > n-x-y)\cdot(1-p)^{x+y-2}p^2 \\
&= \textstyle\sum_{x=1}^{n}\sum_{y=1}^{n-x}(1-p)^{n-x-y}\cdot(1-p)^{x+y-2}p^2 \\
&= p^2(1-p)^{n-2}\textstyle\sum_{x=1}^{n}\sum_{y=1}^{n-x} 1 \\
&= \tfrac{n(n-1)}{2}p^2(1-p)^{n-2} = \binom{n}{2}p^2(1-p)^{n-2} \quad \surd
\end{aligned}$$

We'll leave the other cases as an exercise... □

**Claim 2:**

Let $E_1, E_2, \ldots$ be iid exponential random variables with rate 1.

Let $X$ be the smallest integer such that

$$\sum_{i=1}^{X+1} \frac{E_i}{n-i+1} > -\ln(1-p).$$

Then $X \sim bin(n,p)$.

A brute force proof would be similar to that in the previous claim. For those with more MathStat experience, you might make a transformation first to deal with a product of Beta random variables rather than the sum of exponentials!

#### 2.1.4.3 Poisson

Let $X$ be a Poisson random variable with parameter $\lambda$. We write $X \sim Poisson(\lambda)$. Recall that the pdf for $X$ is

$$f(x) = P(X=x) = \frac{e^{-\lambda}\lambda^x}{x!}\, I_{\{0,1,2,\ldots,\}}(x).$$

If using the algorithms of Sections 2.1.1 and/or 2.1.2, you might want to take advantage of the recursive relationship

$$P(X=x+1) = \frac{\lambda}{x+1}\cdot P(X=x).$$

Here are two more “for fun" relationships that you could, but probably wouldn't, use for simulating a Poisson random variable.

**Claim 1:**

Let $E_1, E_2, \ldots$ be iid exponential random variables with rate 1.

Let $X$ be the smallest integer such that

$$\sum_{i=1}^{X+1} E_i > \lambda.$$

Then $X \sim Poisson(\lambda)$.

**(Proof to be added soon...)**

**Claim 2:**

Let $U_1, U_2, \ldots,$ be iid $unif(0,1)$ random variables.

Let $X$ be the smallest integer such that

$$\prod_{i=1}^{X+1} U_i > e^{-\lambda}.$$

Then $X \sim Poisson(\lambda)$.

**(Proof to be added soon...)**

## 2.2 Generating Some Continuous Random Variables

In this Chapter, we are focusing on one-dimensional random variables from pretty common distributions. Even in this simplified setting, there is no one method that will work for all distributions. Our first method assumes one can compute and invert the cdf of the distribution.

### 2.2.1 The Inverse CDF Method

Suppose that $X$ is a continuous random variable with cdf $F(x) = P(X \leq x)$. Suppose further that we can invert $F$.

Let $U \sim unif(0,1)$. Then the random variable defined as $F^{-1}(U)$ has the same distribution as $X$. We write

$$F^{-1}(U) \stackrel{d}{=} X$$

to say that they are “equal in distribution".

We will prove this, but first note that it gives us an easy way to generate copies of $X$:

1. Draw a $U \sim unif(0,1)$ with your RNG.
2. Output $F^{-1}(U)$ as your simulated value of $X$.

To prove the claim that $F^{-1}(U)$, note that all cdfs are non-decreasing and that all invertible cdfs are increasing. If we define a random variable $Y$ as $Y := F^{-1}(U)$, we have that the cdf of $Y$ is as follows.

$$\begin{aligned} F_Y(y) &= P(Y \leq y) = P(F^{-1}(U) \leq y) \\ &\overset{F \text{ inc}}{=} P(U \leq F(y)) = F(x). \end{aligned}$$

That last equality is due to the fact that the cdf for the $unif(0,1)$ distribution is

$$P(U \leq u) = \begin{cases} 0 & , \quad u < 0 \\ u & , \quad 0 \leq u < 1 \\ 1 & , \quad u \geq 1 \end{cases}$$

as well as the fact that a cdf takes values between 0 and 1.

So, we have shown that the cdf of $Y$ is the same as the cdf of $X$. This means that they have the same distribution. (If you are coming from a "pdf-centric" background, assume the cdfs are differentiable and take the derivative on both sides of $F_Y(y) = F(x)$ to get that the pdfs are the same.)

**Note:** The discrete random variable generating algorithms of Sections 2.1.1 and/or 2.1.2 that relied on chopping up the unit interval according to the desired probabilities is really this inverse cdf method using the generalized inverse

$$F^{-1}(y) = \inf\{x : F(x) \geq y\}.$$

**Example:**

Let's simulate $100,000$ values from the exponential distribution with rate $\lambda$. The pdf is

$$f(x) = \lambda e^{-\lambda x}$$

on $x > 0$.

The cdf is

$$F(x) = \int_0^x f(u)\,du = \int_0^x \lambda e^{-\lambda u}\,du = 1 - e^{-\lambda x}.$$

The inverse cdf is

$$F^{-1}(x) = -\frac{1}{\lambda}\ln(1-y).$$

Repeat the following $100,000$ times.

1. Draw $U \sim unif(0,1)$ using our RNG.
2. Output $-\frac{1}{\lambda}\ln(1-U)$

The results are shown in Figure 2.1. Note that we could have returned $-\frac{1}{\lambda}\ln U$ since $1-U$ and $U$ have the same distribution. □

### 2.2.2 The Accept-Reject Method

The inverse cdf method is a simple yet efficient way to simulate random variables from a univariate distribution. However, we often can not invert, or even compute in closed form, the cdf for a distribution. In this case, the **accept-reject method** for simulating random variables is usually the next thing pulled from the simulator's bag of tricks. While this method can be used for simulating from discrete distributions, we will state and prove everything in this Section in the continuous setting.

Suppose that the "target" pdf (the pdf that we wish to simulate from) is $f$. In order to use the accept-algorithm reject one must be able to find a function $g$ such that

1. $g(x) \geq f(x)$ for all $x$ in the support of $f$,
2. $g$ is integrable over the support of $f$, and
3. we can draw/simulate values from the pdf defined as

$$h(x) = \frac{1}{c}g(x)$$

where $c := \int g(x)\,dx$. The integral is taken over the support of the pdf $f$.

At this point in the course, Step 3 would have to be done, if possible, with the inverse cdf method.

The accept-reject sampling scheme is then described in Algorithm 2.

**Algorithm 2** The Accept-Reject Algorithm for Simulating One Value from $f$

**start loop**
  Simulate $Y \sim h$.
  Simulate $U \sim unif(0,1)$.
  **if** $U \leq \frac{f(Y)}{ch(Y)} = \frac{f(Y)}{g(Y)}$ **then**
    output Y
    exit loop
  **end if**
**end loop**

Note that for each "proposal" $Y$ from $h$, we will accept $Y$ as a draw from $f$ with probability $f(Y)/g(Y) \leq 1$. The algorithm may use several $Y$'s and $U$'s before a value is accepted. Let

$Y_1, Y_2, \ldots$ and $U_1, U_2, \ldots$ be the sequence of $Y$'s and $U$'s used. The $Y$'s are iid, the $U$'s are iid and the $Y$'s are independent of the $U$'s.

To prove that the accept-reject algorithm is simulating from $f$, let $X$ be a value output from the algorithm. Since the pdf for a continuous random variable does not have a nice intepretation as probability, we will first compute the cdf for $X$ and then take it's derivative to get to the pdf. Hopefully we will see that the pdf for $X$ is $f$.

The cdf for $X$ is $F_X$ where

$$\begin{aligned} F_X(x) &= P(X \le x) = \textstyle\sum_{n=1}^{\infty} P(X \le x, \text{1st accept is on } n\text{th trial}) \\ &= \textstyle\sum_{n=1}^{\infty} P\left(n-1 \text{ failures}, Y_n \le x, U_n \le \frac{f(Y_n)}{ch(Y_n)}\right) \\ &\overset{indep}{=} \textstyle\sum_{n=1}^{\infty} P(n-1 \text{ failures}) \cdot P\left(Y_n \le x, U_n \le \frac{f(Y_n)}{ch(Y_n)}\right) \end{aligned}$$

For the probability involving $Y_n$ and $U_n$, we will condition on the value of $Y_n$:

$$\begin{aligned} P\left(Y_n \le x, U_n \le \tfrac{f(Y_n)}{ch(Y_n)}\right) &= \textstyle\int_{-\infty}^{\infty} P\left(Y_n \le x, U_n \le \frac{f(Y_n)}{ch(Y_n)} \middle| Y_n = y\right) h(y)\, dy \\ &= \textstyle\int_{-\infty}^{\infty} P\left(y \le x, U_n \le \frac{f(y)}{ch(y)} \middle| Y_n = y\right) h(y)\, dy \\ &= \textstyle\int_{-\infty}^{x} P\left(U_n \le \frac{f(y)}{ch(y)} \middle| Y_n = y\right) h(y)\, dy \\ &\overset{indep}{=} \textstyle\int_{-\infty}^{x} P\left(U_n \le \frac{f(y)}{ch(y)}\right) h(y)\, dy \end{aligned}$$

Since $U_n \sim unif(0,1)$ and $\frac{f(y)}{ch(y)} = \frac{f(y)}{g(y)} \le 1$, we have that

$$\begin{aligned} P\left(Y_n \le x, U_n \le \tfrac{f(Y_n)}{ch(Y_n)}\right) &= \textstyle\int_{-\infty}^{x} P\left(U_n \le \frac{f(y)}{ch(y)}\right) h(y)\, dy \\ &= \textstyle\int_{-\infty}^{x} \frac{f(y)}{ch(y)} h(y)\, dy \\ &= \textstyle\frac{1}{c}\int_{-\infty}^{x} f(y)\, dy = \frac{1}{c}F(x) \end{aligned}$$

where $F$ is the cdf associated with the target density $f$.

Putting it all together, we have that the cdf for $X$, the value output by the accept-reject algorithm is

$$F_X(x) = \sum_{n=1}^{\infty} P(n-1 \text{ failures}) \cdot \frac{1}{c} F(x) \tag{2.2}$$

Since cdfs have the property of approching 1 as the argument goes to infinity, taking the limit

of both sides as $x \to \infty$ gives

$$1 = \sum_{n=1}^{\infty} P(n-1 \text{ failures}) \cdot \frac{1}{c}$$

which implies that

$$\sum_{n=1}^{\infty} P(n-1 \text{ failures}) = c.$$

Plugging this back into (2.2) gives us that

$$F_x(x) = F(x).$$

Taking derivatives of both sides gives us that

$$f_X(x) = f(x).$$

Since $X$ was defined as the output of the accept-reject algorithm, we see that the algorithm produces an $X$ with pdf equal to the target pdf $f$, as desired.

**Example:** Suppose that we want to simulate values from the $\Gamma(\alpha, \beta)$ distribution with pdf

$$f(x) - \frac{1}{\Gamma(\alpha)} \beta^{\alpha} x^{\alpha-1} e^{-\beta x} \, I_{(0,\infty)}(x)$$

for fixed $\alpha > 0$ and $\beta > 0$. The pdf is depicted in Figure 2.2 for the cases where $0 < \alpha < 1$ and $\alpha \geq 1$. Identifying an appropriate and tractable $g$ to use accept-reject in the first case is hard! We will address simulation of the $\Gamma(\alpha, \beta)$ distribution for $0 < \alpha < 1$ later in Section **??**. (Not written yet and not sure where it's going to be yet!) For this example, we will assume that $\alpha > 1$. (If $\alpha = 1$, the gamma distribution is an exponential distribution and one should revert back to the inverse cdf method!)

First, we note that if $X \sim \Gamma(\alpha, \beta)$, and $c > 0$ is a constant, then $cY \sim \Gamma(\alpha, \beta/c)$. Thus, for simulating a $\Gamma(\alpha, \beta)$, we will use the accept-reject method to simulate from the $\Gamma(\alpha, 1)$ distribution and divide our resulting values by $\beta$. We will attempt to bound the target pdf

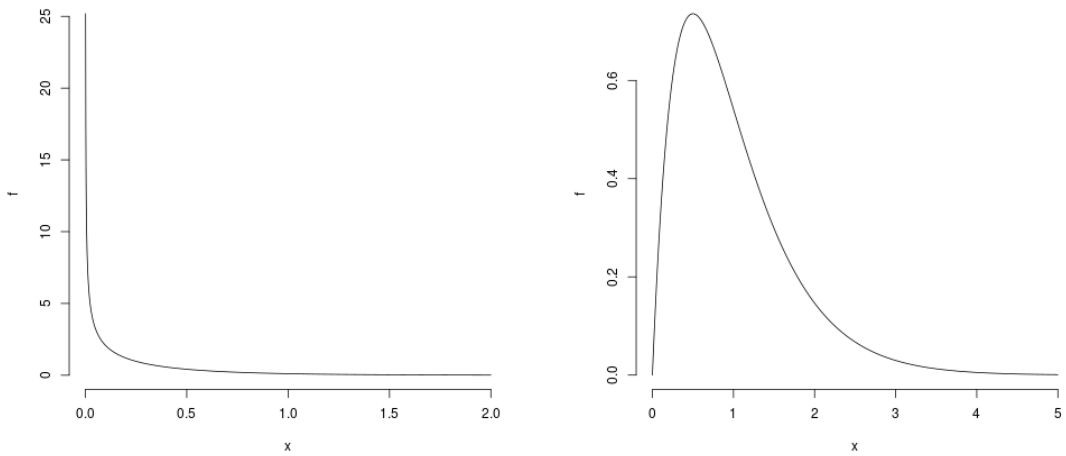
$$f(x) = \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x} \, I_{(0,\infty)}(x)$$

above by a function of the form $g(x) = e^{-\lambda x}$ for some $\lambda > 0$. If we can do this, $g$ can be easily normalized into an exponential pdf (i.e. $h(x) = \lambda e^{-\lambda x} \, I_{(0,\infty)}$) which is easy to draw from using the inverse cdf method. In fact, let's start with $h(x) = \lambda e^{-\lambda x} \, I_{(0,\infty)}(x)$ and work backwards to find a $c$ such that $ch(x) \geq f(x)$ which is a $c$ such that

$$c \geq \frac{f(x)}{h(x)}.$$

The tighter the bound, the better for faster acceptance. So, we will attempt to maximize

Figure 2.2: The $\Gamma(\alpha, \beta)$ PDF




(a) $0 < \alpha < 1$ (b) $\alpha \geq 1$

$f/h$. (Note: If we were really concerned with speeding up the algorithm, we may not want to use the exponential distribution as a proposal distribution. Ideally, one would try to find a bounding function $g$ that more closely mimics the target density $f$.)

With Calculus, is is routine to show that $f/h$ is maximized at

$$x = \frac{\alpha - 1}{1 - \lambda}.$$

Note that the support of the distribution is $(0, \infty)$ and the rate $\lambda$ for the exponential distribution must be positive. Thus, we are free to choose any $\lambda$ in $(0, 1)$. (Yes, there may be an optimal $\lambda$ to choose in $(0, 1)$ but I will not look for it here. Again, if we cared so deeply about the speed of this algorithm, we probably wouldn't be using the exponential distribution in the first place!)

So, we now have that

$$c = \max_{x>0} \frac{f(x)}{h(x)} = \frac{1}{\lambda \Gamma(\alpha)} \left( \frac{\alpha - 1}{1 - \lambda} \right)^{\alpha - 1} e^{-(1-\lambda)\left(\frac{\alpha-1}{\lambda-1}\right)}.$$

### 2.2.3 The Normal Distribution

While we could sample from the normal distribution using the accept-reject algorithm, this Section gives two popular "normal specific" ways to do it instead. The methods herein will generate standard normal random variables which can be transformed to more general normals via the relationship

$$Z \sim N(0, 1) \qquad \Rightarrow \qquad X := \sigma Z + \mu \sim N(\mu, \sigma^2).$$

#### 2.2.3.1 The Box-Muller Transformation

This method, from the late 50's is named for G.E. Box and M.E. Muller.

Suppose that $U_1$ and $U_2$ are iid uniform $(0,1)$ random variables.

Define

$$X_1 = \sqrt{-2\ln U_1}\cos(2\pi U_2) \quad \text{and} \quad X_2 = \sqrt{-2\ln U_1}\sin(2\pi U_2).$$

Then $X_1$ and $X_2$ are independent standard normal random variables!

PROOF Consider $X_1^2 + X_2^2$:

$$\begin{aligned} X_1^2 + X_2^2 &= (-2\ln U_1\cos^2(2\pi U_2) + (-2\ln U_2)\sin^2(2\pi U_2) \\ &= (-2\ln U_1)(\cos^2(2\pi U_2) + \sin^2(2\pi U_2)) \\ &= (-2\ln U_1)\cdot 1 = -2\ln U_1 \end{aligned}$$

So, we have

$$U_1 = e^{-\frac{1}{2}(X_1^2+X_2^2)}.$$

To solve for $U_2$, I will use the fact that $\tan x = \sin x/\cos x$, and consider the ratio $X_2/X_1$:

$$\begin{aligned} \tfrac{X_2}{X_1} &= \tfrac{\sqrt{-2\ln U_1}\sin(2\pi U_2)}{\sqrt{-2\ln U_1}\cos(2\pi U_2)} \\ &= \tfrac{\sin(2\pi U_2)}{\cos(2\Pi U_2)} = \tan(2\pi U_2) \end{aligned}$$

So, we have that

$$U_1 = g_1^{-1}(X_1, X_2) = e^{-\frac{1}{2}(X_1^2+X_2^2)} \qquad \text{and} \qquad U_2 = g_2^{-1}(X_1, X_2) = \frac{1}{2\pi}\tan^{-1}\left(\frac{X_2}{X_1}\right)$$

and the Jacobian of the transformation is

$$J = \begin{vmatrix} \frac{\partial u_1}{\partial x_1} & \frac{\partial u_1}{\partial x_2} \\ \frac{\partial u_2}{\partial x_1} & \frac{\partial u_2}{\partial x_2} \end{vmatrix} = \begin{vmatrix} -x_1\, e^{-\frac{1}{2}(x_1^2+x_2^2)} & -x_2\, e^{-\frac{1}{2}(x_1^2+x_2^2)} \\ \frac{1}{2\pi}\left(-\frac{x_2}{x_1^2}\right)\frac{1}{1+(x_2/x_1)^2} & \frac{1}{2\pi}\left(\frac{1}{x_1}\right)\frac{1}{1+(x_2/x_1)^2} \end{vmatrix}$$

$$= \begin{vmatrix} -x_1\, e^{-\frac{1}{2}(x_1^2+x_2^2)} & -x_2\, e^{-\frac{1}{2}(x_1^2+x_2^2)} \\ \frac{1}{2\pi}\frac{-x_2}{x_1^2+x_2^2} & \frac{1}{2\pi}\frac{x_1}{x_1^2+x_2^2} \end{vmatrix} = -\frac{1}{2\pi}e^{-\frac{1}{2}(x_1^2+x_2^2)}.$$

Therefore, the joint density of $X_1$ and $X_2$ is given by

$$\begin{aligned} f_{X_1,X_2}(x_1,x_2) &= f_{U_1,U_2}(g_1^{-1}(x_1,x_2), g_2^{-1}(x_1,x_2)) \cdot |J| \\ &= I_{(0,1)}\left(e^{-\frac{1}{2}(x_1^2+x_2^2)}\right) \cdot I_{(0,1)}\left(\tfrac{1}{2\pi}\tan^{-1}\left(\tfrac{x_2}{x_1}\right)\right) \cdot \tfrac{1}{2\pi} e^{-\frac{1}{2}(x_1^2+x_2^2)} \end{aligned}$$

Here begins a long note about indicators... brace yourself.

Note that the first indicator is always 1 since $e^{-\frac{1}{2}(x_1^2+x_2^2)}$ lives between 0 and 1. (Don't be alarmed about the idea that it could equal 1. That will happen when both $x_1$ and $x_2$ are zero. Since they are continuous random variables, this will happen with probability zero. For similar reasons, you could have started with your uniforms on $[0,1]$ as opposed to $(0,1)$ since those single endpoints don't matter for a continuous random variable.) Since the first indicator is always 1, we can drop it.

Now, we are left with only one indicator that takes the value 1 whenever

$$0 < \frac{1}{2\pi}\tan^{-1}(x_2/x_1) < 1.$$

By examination of arctan, this appears to only be true for $x_2/x_1 > 0$, which leads us to quadrants 1 and 3 of the $(x_1, x_2)$ plane. This does not seem to have us heading in the right direction for independent standard normal random variables!

The problem here is with the definition of the arctan. $y = \tan(x)$ is not invertible until we restrict its domain. It is usually restricted to $(-\pi/2, \pi/2)$ but doesn't have to be. We may invert it on different regions. In this problem, the restriction ends up restricting possible values for $x_1$ and $x_2$! Indeed, at a previous point in this solution, we had that

$$\frac{x_2}{x_1} = \tan(2\pi u_2).$$

Here, the right-hand side is taking on values from $-\infty$ to $\infty$ so the left-hand side should be able to take on these values as well. It appears at this point that $x_1$ and $x_2$ can both take on values from $-\infty$ to $\infty$. It was not until we applied the arctan that we got some restriction.

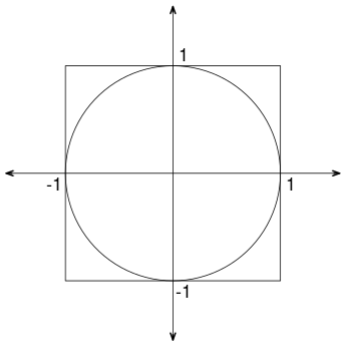
Indeed, if one looks at the original definitions of $x_1$ and $x_2$,

$$\begin{aligned} x_1 &= \sqrt{-2\ln u_1}\cos(2\pi u_2) \\ x_2 &= \sqrt{-2\ln u_1}\sin(2\pi u_2) \end{aligned}$$

it is "easy" to see that any $(x_1, x_2)$ pair is possible. Since this region (the entire plane!) is rectangular, we have that the joint pdf for $X_1$ and $X_2$ is

$$\begin{aligned} f_{X_1,X_2}(x_1,x_2) &= \tfrac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x_1^2+x_2)}\, I_{(-\infty,\infty)}(x_1) \cdot I_{(-\infty,\infty)}(x_2) \\ &= \tfrac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x_1^2}\, I_{(-\infty,\infty)}(x_1) \cdot \tfrac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x_2^2} I_{(-\infty,\infty)}(x_2). \end{aligned}$$

Figure 2.3: A Circle in a Square




Since the joint density factors, we see that $X_1$ and $X_2$ are independent. Furthermore, from the form of the desity, we see that they are $N(0,1)$ random variables! □

#### 2.2.3.2 Marsaglia's Polar Method

In the late 60's, G. Marsaglia proposed the following method for generating $N(0,1)$ random variables.

Let $U_1$ and $U_2$ be iid uniform $(0,1)$ random variables.

Define $V_1$ and $V_2$ as

$$V_1 := 2U_1 - 1 \quad \text{and} \quad V_2 := 2U_2 - 1.$$

Note that $V_1, V_2 \overset{iid}{\sim} unif(-1,1)$ and that the point $(V_1, V_2)$ is uniformly distributed within the unit square.

If $S := V_1^2 + V_2^2 \leq 1$, let

$$C = \sqrt{\frac{-2}{S}\ln(S)}.$$

$\ni Then \mathrm{X}_1 = CV_1$ and $X_2 = CV_2$ are independent $N(0,1)$ random variables.

(Note: If $S > 1$, one must start all over with new $U_1$ and $U_2$.)

Proof As mentioned, the point $(V_1, V_2)$ is uniformly distributed on the square shown in Figure 2.3.

If this point $(V_1, V_2)$ happens to fall in the unit circle (ie: $S := V_1^2 + V_2^2 \leq 1$), then it will also be uniformly distributed in the circle. Assuming we have accepted the point $(V_1, V_2)$, it has

fallen in the unit circle and it therefore uniformly distributed in the circle. The joint pdf is therefore

$$f_{V_1,V_2}(v_1,v_2) = \frac{1}{\pi} I_{(0,1)}(v_1^2+v_2^2).$$

We can transform $V_1$ and $V_2$ to polar coordinates and write:

$$V_1 = \sqrt{S}\cos\Theta \qquad V_1 = \sqrt{S}\sin\Theta$$

for some random $\Theta$ on $(0, 2\pi)$.

We now find the joint density of $S$ and $\Theta$. We have $S$ and $\Theta$ represented implicitly as some functions $g_1(V_1, V_2)$ and $g_2(V_1, V_2)$. The system is already inverted as

$$\begin{aligned} V_1 &= g_1^{-1}(S,\Theta) = \sqrt{S}\cos\Theta \\ V_2 &= g_1^{-1}(S,\Theta) = \sqrt{S}\sin\Theta \end{aligned}$$

Now,

$$f_{S,\Theta}(s,\theta) = f_{V_1,V_2}(g_1^{-1}(s,\theta), g_2^{-1}(s,\theta)) \cdot |J|$$

where

$$J = \begin{vmatrix} \frac{\partial v_1}{\partial s} & \frac{\partial v_1}{\partial \theta} \\ \frac{\partial v_2}{\partial s} & \frac{\partial v_2}{\partial \theta} \end{vmatrix} = \begin{vmatrix} \frac{1}{2\sqrt{s}} sin(\theta) & \sqrt{s}\, cos(\theta) \\ \frac{1}{2\sqrt{s}}\, cos(\theta) & -\sqrt{s}\, sin(\theta) \end{vmatrix} = -\frac{sin^2(\theta)}{2} - \frac{cos^2(\theta)}{2} = -\frac{1}{2}.$$

So,

$$\begin{aligned} f_{S,\Theta}(s,\theta) &= f_{V_1,V_2}(\sqrt{s}\, sin(\theta), \sqrt{s}\, cos(\theta)) \cdot |-1| = -\tfrac{1}{2} \\ &= \tfrac{1}{\pi} I_{(0,1)}(s) I_{(0,2\pi)}(\theta). \end{aligned}$$

So we see that $S$ and $\Theta$ are independent and are uniform on $(0, 1)$ and $(0, 2\pi)$, respectively.

We can now rewrite the Box-Muller transformations, which we know gives independent standard normals, as

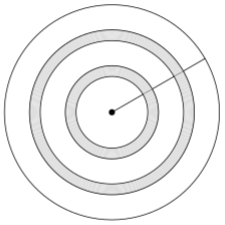
$$X_1 := \sqrt{-2\ln(U_1)}\cos(\theta) = \sqrt{-2\ln(S)}\,\frac{V_2}{\sqrt{S}} = \sqrt{\frac{-2\ln(S)}{S}} V_2 = CV_1$$

and

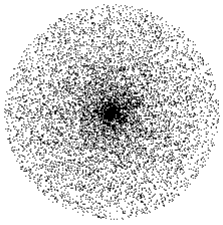
$$X_1 := \sqrt{-2\ln(U_1)}\sin(\theta) = \sqrt{-2\ln(S)}\,\frac{V_1}{\sqrt{S}} = \sqrt{\frac{-2\ln(S)}{S}} V_1 = CV_2.$$

□

Figure 2.4: Uniform $\Theta$ and Uniform $R$ is Bad!



(a) Radius selected along angle line in two intervals of length 1/3.



(b) Results!

## 2.3 Of Disks and Spheres

This is a homeless Section that needed a place in the notes somewhere before an example in Chapter 3.

**In $\mathbb{R}^2$:**

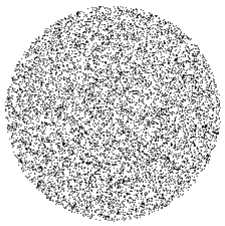
Suppose we want to draw points uniformly on the unit circle in $\mathbb{R}^2$. One method would be to enclose the circle in the unit square (as in Figure 2.3), draw points uniformly on the unit square, and accept any point that falls inside the circle. In this Section, we look for a way to generate such point directly without an acceptance rejection step. (Indeed, the accept-reject approach will result in a significant slowdown when randomly sampling from high-dimensional unit balls!)

A natural idea that one might have is to generate an angle $\Theta \sim unif(0, 2\pi)$ and then independently generate a radius $R \sim unif(0, 1)$.

In Figure 2.4 (a), two bands with the same width are shaded in grey. For the given angle, if we draw a radius uniformly, we are equally likely to choose a radius in either band. Letting the angle vary, the proposed two step procedure would result in sampled points that are equally likely to be in either of the shaded bands. However, since the outer band has more area than the inner band we should be choosing points in the outer band with greater frequency. Since we are not, our resulting "uniform" draws on the unit disk will end up with too many points towards the smaller bands in the center, as depicted in Figure 2.4 (b).

Given the approach of drawing first a uniform angle on $(0, 2\pi)$ and then drawing a radius, how can we draw a radius to result in uniform draws in the disk? Note that, if we want a uniform draw on the disk, probability that we fall in any subregion of the disk must be the area of the subregion divided by the area of the disk. In particular the cdf for the radius must

Figure 2.5: Uniform Draws in the Unit Disk



be

$$P(R \leq r) = \frac{\text{area of disk of radius } r}{\text{area of unit disk}} = \frac{\pi r^2}{\pi(1)^2} = r^2$$

and so the pdf for the radius must be

$$f_R(r) = \frac{d}{dr} P(R \leq r) = 2r\, I_{(0,1)}(r).$$

Using the inverse cdf method, we would draw $U \sim unif(0,1)$ and let $R = \sqrt{U}$. The result is shown in Figure 2.5.

**In $\mathbb{R}^3$:**

To draw points uniformly from the unit ball $B := \{(x_1, x_2, x_3) : x_1^2 + x_2^2 + x_3^2 \leq 1\}$, we will take a similar approach to the 2-d case by first drawing a point uniformly on the surface of the sphere and then shrinking it towards the origin by drawing a radius $R$ from an appropriate density. The cdf for this radius must be

$$P(R \leq r) = \frac{\text{volume of ball of radius } r}{\text{volume of unit ball}} = \frac{\frac{4}{3}\pi r^3}{\frac{4}{3}\pi(1)^3} = r^3$$

and so the pdf for the radius must be

$$f_R(r) = \frac{d}{dr} P(R \leq r) = 3r^2\, I_{(0,1)}(r).$$

which we can again draw from using the inverse cdf method.

As for getting a point uniform on the surface of the unit sphere, you might be thinking of uniformly and independently drawing inclination and azimuthal angles. However, for reasons

similar to the “band argument" depicted in Figure 2.4 (a), this will not give you a point uniformly distributed on the surface of the sphere. So, what to do....

**Claim:** Let $X_1, X_2, X_3$ be iid $N(0,1)$ random variables. Then the point

$$\left( \frac{X_1}{\sqrt{X_1^2 + X_2^2 + X_3^2}}, \frac{X_2}{\sqrt{X_1^2 + X_2^2 + X_3^2}}, \frac{X_3}{\sqrt{X_1^2 + X_2^2 + X_3^2}} \right)$$

is uniformly distributed on the surface of the unit sphere.

As for the proof of the Claim, my plan is to come back to it after I catch up on completing Chapter 3. The Claim may be extended to any dimension. In fact, we already know it is true in $\mathbb{R}^2$. Recall that we drew a point on the unit circle by drawing a $\Theta \sim unif(0, 2\pi)$. This corresponds to a point

$$(X, Y) = (\cos\Theta, \sin\Theta).$$

To draw that $unif(0, 2\pi)$, we would draw $U \sim unif(0, 1)$ and set $\Theta = 2\pi U$. Thus, our random point can be written as

$$(X, Y) = (\cos(2\pi U), \sin(2\pi U)). \tag{2.3}$$

By the Box-Muller transformation, we know that we can draw $X_1, X_2 \overset{iid}{\sim} N(0,1)$ by drawing $U_1, U_2 \overset{iid}{\sim} unif(0,1)$ and setting

$$\begin{aligned} X_1 &= \sqrt{-2\ln U_1}\cos(2\pi U_2) \\ X_2 &= \sqrt{-2\ln U_1} \sim (2\pi U_2). \end{aligned}$$

According to the Claim made in this Section, we would then get a point on the unit circle using

$$\left( \frac{X_1}{\sqrt{X_1^2 + X_2^2}}, \frac{X_2}{\sqrt{X_1^2 + X_2^2}} \right) = \left( \frac{\sqrt{-2\ln U_1}\cos(2\pi U_2)}{\sqrt{\ln U_1}}, \frac{\sqrt{-2\ln U_1}\sin(2\pi U_2)}{\sqrt{\ln U_1}} \right)$$

$$= (\cos(2\pi U_2), \sin(2\pi U_2)).$$

This has the same distribution as 2.3 which we know is uniformly distributed on the unit circle!

**In $\mathbb{R}^3$:**

To be written– a direct and “obvious" generalization of the $\mathbb{R}^3$ case.

## Homework Problems

1. Use the inverse cdf method to simulate $100,000$ values from the continuous Pareto distribution with pdf
$$f(x) = \frac{\gamma}{(1+x)^{\gamma+1}} I_{(0,\infty)}(x).$$
Give a histogram of your results with the true density superimposed.

2. Let $f$ be a generic continuous target density on $(-\infty, \infty)$. Suppose that $f$ is bounded above by a function $g$ that is suitable (tractable) for the accept-reject algorithm. Derive an expression for the expected number of times the algorithm will have to loop to produce a single draw from $f$.

3. Perform the accept-reject algorithm to simulate $100,000$ values from the $\Gamma(3,2)$ distribution. (You are more than welcome to use the proposal/example in Section 2.2.2.) Give a histogram of your results with the true density superimposed. Also, keep track of the mean time to acceptance and compare this with your answer to Problem 1.

Chapter 3

# Monte Carlo Integration and Variance Reduction Techniques

The goal of this Chapter is to evaluate the integral

$$I = \int g(\vec{x})\, d\vec{x}$$

over some region using simulation of random variables.

Why use random methods?

Computation by "deterministic quadrature" can quickly become expensive and inaccurate as grid points add up quickly in high dimensions. Bad choices of grid may misrepresent $g(\vec{x})$. That said, this Chapter will focus on the basics of "Monte Carlo" integration for one-dimensional integrals since, at this point in the course, we only know how to simulate values for some one-dimensional random variables. Many of the instances of $x$ and $dx$ to follow can easily be replaced with $\vec{x}$ and $d\vec{x}$.

## 3.1 Integrating

### 3.1.1 Hit and Miss Monte Carlo

Suppose we have a non-negative function $g(x)$ that is bounded over an interval $[a, b]$, and that we wish to integrate $g$ over $[a, b]$. We can put a bounding box $[a, b] \times [0, c]$ around it as shown in Figure 3.1 where $c = \max_{a \le x \le b} g(x)$. Since the integral is the shaded area under the curve, we could throw "uniform darts" at the bounding box and estimate $I$ as the proportion of time we land under the curve multiplied by the area of the rectangle.

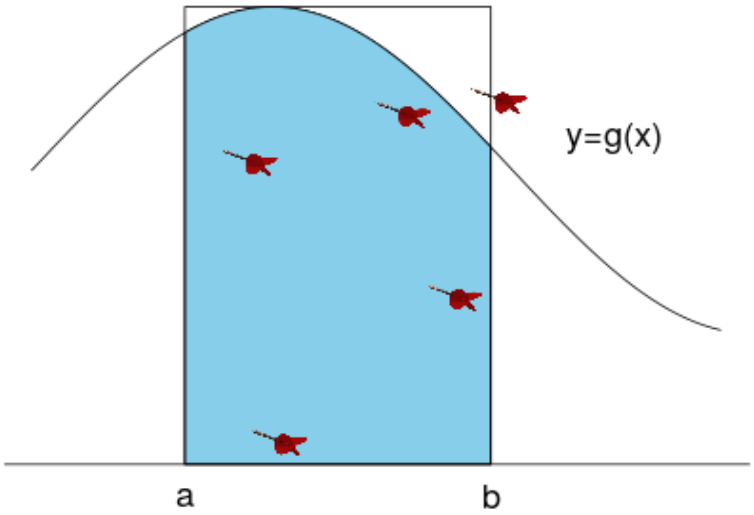
The probability that a dart lands below the curve is

$$p := \frac{I}{c(b-a)}, \tag{3.1}$$

so if we can estimate $p$ with an observed proportion $\widehat{p}$, we can estimate $I$ with

$$\widehat{I} = \widehat{p}c(b-b).$$

Figure 3.1: Throwing Darts at a Bounding Box




Throw $n$ darts and let

$$\widehat{p} = \frac{\#\text{ times under the curve}}{n}$$

then $\widehat{p}$ is a sample mean of a Bernoulli random variable with parameters $n$ and $p$. By the Central Limit Theorem (CLT), this sample mean is approximately normally distributed for large $n$ with mean $p$ and variance $p(1-p)/n$. Thus, we have that

$$Z := \frac{\widehat{p} - p}{\sqrt{\frac{p(1-p)}{n}}} \overset{approx}{\sim} N(0,1). \tag{3.2}$$

An approximate $100(1-\alpha)\%$ confidence interval for $p$ can be found by putting $Z$ between the two z-critical values $-z_{\alpha/2}$ and $z_{\alpha/2}$ and solving for $p$ "in the middle". So, we want to solve for $p$ in

$$-z_{\alpha/2}\sqrt{\frac{p(1-p)}{n}} < \widehat{p} - p < z_{\alpha/2}\sqrt{\frac{p(1-p)}{n}}.$$

It is possible to solve for $p$ exactly by squaring all sides and plowing through the algebra to get

$$\frac{\widehat{p} + \frac{z^2_{\alpha/2}}{2n} \pm z_{\alpha/2}\sqrt{\frac{\widehat{p}(1-\widehat{p})}{n} + \frac{z^2_{\alpha/2}}{4n^2}}}{1 + \frac{z^2_{\alpha/2}}{n}}.$$

However, most people don't bother with this kind of precision since we are already approximating via the CLT. Instead, they replace the $p$'s under the square root in (3.2) with $\widehat{p}$'s and

just return

$$\widehat{p} \pm z_{\alpha/2}\sqrt{\frac{\widehat{p}(1-\widehat{p})}{n}}$$

as an approximate $100(1-\alpha)\%$ confidence interval for $p$. By (3.1) we can multiply both endpoints of this interval by $c(b-a)$ to get an approximate confidence interval for our target integral $I$.

**Example:**

Consider the integral

$$I = \int_0^3 e^x \, dx = e^3 - 1 \approx 19.08554.$$

An upper bound for $y = e^x$ on the interval $[0, 3]$ is $c = e^3$.

We can simulate our dart hit points by

- drawing an $x$-coordinate uniformly over $(0, 3)$
- drawing a $y$-coordinate uniformly over $(0, e^3)$.

I ran 5 different simulations, each with $100,000$ dart throws. The results were as follows.

| $\widehat{p}$ | $\widehat{I}$ | 95% CI |
|---|---|---|
| 0.31365 | 19.06217885 | (18.88849413, 19.23586357) |
| 0.31552 | 19.01216586 | (18.83860388, 19.18572785) |
| 0.31729 | 19.11882006 | (18.94499712, 19.29264301) |
| 0.31642 | 19.06639681 | (18.89270180, 19.24009185) |
| 0.31672 | 19.08447380 | (18.91073460, 19.25821302) |

### 3.1.2 Sample Mean Monte Carlo

In this section, we again wish to evaluate an integral over a finite interval $[a, b]$. Note that we can rewrite our target integral

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

as

$$I = (b-a)\int_a^b g(x)\frac{1}{b-a}\, dx = (b-a)\mathsf{E}[g(X)]$$

where $X \sim unif(a, b)$. Thus, we can approximate $I$ with

$$\widehat{I} = (b-a) \cdot \frac{1}{n}\sum_{i=1}^{n} g(X_i)$$

where $X_1, X_2, \ldots, X_n$ is a random sample from the uniform distribution on $(a, b)$. It is easy to verify that $\mathsf{E}[\widehat{I}] = I$.

Note that

$$\widehat{I} = \frac{b-a}{n}\sum_{i=1}^{n} Y_i = (b-a)\overline{Y}$$

where $Y_i = g(X_i)$ for $i = 1, 2, \ldots, n$. So, we can once again appeal to the CLT for an approximate confidence interval for $I$. A rough $100(1-\alpha)\%$ confidence interval for $I$ is given by

$$\widehat{I} \pm z_{\alpha/2}\sigma_{\widehat{I}}$$

where

$$\begin{aligned}
\sigma^2_{\widehat{I}} &= Var[\widehat{I}] = Var\left[(b-a)\tfrac{1}{n}\textstyle\sum_{i=1}^{n} g(X_i)\right] \\
&= \tfrac{(b-a)^2}{n^2} Var\left[\textstyle\sum_{i=1}^{n} g(X_i)\right] \\
&\overset{iid}{=} \tfrac{(b-a)^2}{n^2} n Var[g(X_1)] = \tfrac{(b-a)2}{n} Var[g(X_1)]
\end{aligned}$$

To compute $Var[g(X_1)]$ we will need to be able to compute

$$\mathsf{E}[g^2(X_1)] = \int_a^b g^2(x)\frac{1}{b-a}\,dx.$$

However, if we couldn't compute the original integral $I$, it's highly unlikely that we can compute this! So, we will approximate $\widehat{\sigma^2_I}$, we by the sample variance

$$S_{\widehat{I^2}=\frac{(b-a)^2}{n}S^2_Y}$$

where

$$S^2_Y = \frac{\sum Y_i^2 - (\sum Y_i)^2}{n-1}.$$

In this Section, we rewrote the integral so that it looks like an expected value of a function of a uniform distribution. In Section 3.1.4, we will generalize this approach to integration so that we can use all kinds of different distributions. At this point though, we would prefer to use SMMC over hit and miss since

- SMMC does not require a non-negative integrand,
- SMMC does not require that we must be able to put the integrand in a box, and
- as we will see in the next Section, the variance of our SMMC estimator of $I$ is less than or equal to that of the hit and miss estimator.

### 3.1.3 Comparison of Hit and Miss and SMMC

### 3.1.4 Importance Sampling

## 3.2 Variance Reduction

### 3.2.1 Antithetic Monte Carlo

### 3.2.2 Control Variates

### 3.2.3 Rao-Blackwellization

Chapter 4

# Markov Chain Monte Carlo: Part I

Chapter 5

# Markov Chain Monte Carlo: Part II

Chapter 6

# Perfect Simulation

Chapter 7

# Applications and Neat-o Examples

Appendix A

# R Code for Figures

Here lies R code for most of the Figures in these notes. In some cases it might not work for you "right out of the box". You may need to install libraries/packages.

Figure 1.1
```
> a<-2^(16)+3
> m<-2^(31)
> # Set x0
> randu<-1
> for(i in 2:100000){
> randu[i]<-(a*randu[i-1])%%m}
> randu<-randu/m
> br<-seq(0,1,0.1)
> hist(randu,prob=T,breaks="br")
```

Figure 1.2
```
> # Break up into triples
> x<-randu[seq(1,100000,3)]
> y<-randu[seq(2,100000,3)]
> z<-randu[seq(3,100000,3)]
> # x has one too many!
> x<-x[1:33333]
> plot3d(x,y,z,axes=F,cex=0.5,col="grey",xlab="",ylab="",zlab="")
> box3d()
```

Figure 1.3
```
> # 100,000 uniforms are stored in a vector called "mysample"
> x<-mysample[seq(1,100000,2)]
> y<-mysample[seq(2,100000,2)]
> k<-20
> plot(c(0,1),c(0,1),type="n",axes=F,xlab="",ylab="")
> for(i in 0:k){
+    lines(c(i/k,i/k),c(0,1))
+    lines(c(0,1),c(i/k,i/k))
+
+ }
> points(x,y,pch=".")
```

Figure 1.4
```
> mydata<-c(1.46,4.26,0.83,1.67,0.18,3.30)
> plot(ecdf(mydata))  # easy built-in function but I want more customization
```

```
> x<-seq(0,6,0.001)
> F<-1-exp(-0.6*x)
> plot(x,F,type="l",xlim=c(-0.2,6),ylim=c(0,1))
> mydata<-mydata[order(mydata)]
> numpoints<-length(mydata)
> points(mydata,seq(1/numpoints,1,1/numpoints),pch=16)
> lines(c(-1,mydata[1]),c(0,0))
> points(mydata[1],0)
> empcdf<-1/numpoints
> for(i in 2:length(mydata)){
> lines(c(mydata[i-1],mydata[i]),c(empcdf,empcdf))
> points(mydata[i],empcdf)
> empcdf<-empcdf+(1/numpoints)
> }
> lines(c(mydata[numpoints],10),c(1,1))
> for(i in 1:length(mydata)){
> lines(c(mydata[i]+0.15,mydata[i]+0.15),c((i-1)/6,1-exp(-0.6*mydata[i])),
+ col="red")
> lines(c(mydata[i]+0.10,mydata[i]+0.20),c((i-1)/6,(i-1)/6),col="red")
> lines(c(mydata[i]+0.10,mydata[i]+0.20),c(1-exp(-0.6*mydata[i]),
+ 1-exp(-0.6*mydata[i])),col="red")
> }
> for(i in 1:length(mydata)){
> lines(c(mydata[i],mydata[i]),c(i/6,1-exp(-0.6*mydata[i])),col="blue",lwd=2)
> }
```

Figure 2.1

```
> # mysample is a vector already obtained "from scratch" or using "rexp"
> min(mysample)
[1] 8.802391e-05
> max(mysample)
[1] 56.83071
> br<-seq(0,57,1)
> hist(mysample,prob=T,breaks=br)
> x<-seq(0,50,0.001)
> f<-0.2*exp(-0.2*x)
> lines(x,f)
```

Figure 2.2

```
> alpha<-0.5
> beta<-2
> x<-seq(0,2,0.001)
> f<-(1/gamma(alpha))*(beta^alpha)*(x^(alpha-1))*exp(-beta*x)
> plot(x,f,type="l",axes=F)
> axis(1)
> axis(2)
>
```

```
> alpha<-2
> beta<-2
> x<-seq(0,5,0.001)
> f<-(1/gamma(alpha))*(beta^alpha)*(x^(alpha-1))*exp(-beta*x)
> plot(x,f,type="l",axes=F)
> axis(1)
> axis(2)
```

Figure 2.3

```
> plot(c(-1.5,1.5),c(-1.5,1.5),type="n",axes=F,xlab="",ylab="",asp=1)
> lines(c(-1.5,1.5),c(0,0))
> lines(c(0,0),c(-1.5,1.5))
> arrows(1.5,0,1.55,0,length = 0.1,angle=20,lwd=2)
> arrows(-1.5,0,-1.55,0,length = 0.1,angle=20,lwd=2)
> arrows(0,1.5,0,1.55,length = 0.1,angle=20,lwd=2)
> arrows(0,-1.5,0,-1.55,length = 0.1,angle=20,lwd=2)
> rect(-1,-1,1,1)
> draw.circle(0,0,1)
> text(1.1,-0.1,"1",cex=1.5)
> text(-1.1,-0.1,"-1",cex=1.5)
> text(0.1,1.1,"1",cex=1.5)
> text(0.1,-1.1,"-1",cex=1.5)
```

Figure 2.4 (a)

```
> plot(c(-1.5,1.5),c(-1.5,1.5),asp=1,type="n",axes=F,xlab="",ylab="")
> theta<-2*pi*runif(10000)
> r<-runif(10000)
> points(r*cos(theta),r*sin(theta),pch=".")
```

Figure 2.4 (b)

```
> plot(c(-1.5,1.5),c(-1.5,1.5),asp=1,type="n",axes=F,xlab="",ylab="")
> draw.circle(0,0,1)
> lines(c(0,cos(pi/6)),c(0,sin(pi/6)))
> points(0,0,pch=19)
> for(i in 1:10){
> draw.circle(0,0,1/3+0.1*i/10,border="lightgrey")
> }
> draw.circle(0,0,1/3)
> draw.circle(0,0,1/3+0.1)
> for(i in 1:10){
>  draw.circle(0,0,2/3+0.1*i/10,border="lightgrey")
> }
> draw.circle(0,0,2/3)
> draw.circle(0,0,2/3+0.1)
```

Figure 2.5

```
> plot(c(-1.5,1.5),c(-1.5,1.5),asp=1,type="n",axes=F,xlab="",ylab="")
> theta<-2*pi*runif(10000)
> r<-sqrt(runif(10000))
> points(r*cos(theta),r*sin(theta),pch=".")
```

Figure 3.1

```
> x<-seq(0,4.5,0.001)
> y<-0.5*sin(x)
> plot(x,y,type="l",ylim=c(-1,0.8),axes=F,xlab="",ylab="")
> polygon(c(1,x[1001:3001],3),c(-0.9,y[1001:3001],-0.9),col='skyblue')
> lines(c(-1,5),c(-0.9,-0.9))
> rect(1,-0.9,3,0.5)
> text(1,-1,"a",cex=1.5)
> text(3,-1,"b",cex=1.5)
> text(3.8,0.1,"y=g(x)",cex=1.5)
> dart <- readPNG("dart.png")
> x1<-2*runif(5)+1
> y1<-1.4*runif(5)-0.9
> for(i in 1:5){
+  rasterImage(dart, x1[i],y1[i], x1[i]+0.4,y1[i]+0.1)
+ }
```

Appendix B

# Miscellaneous MathStat

Miscellaneous things live here...

## B.1 The $\chi^2$ Goodness of Fit Test

Recall the familiar binomial distribution with parameters $n$ and $p$. If $X$ has this distribution then it can be thought of as the total number of "successes" in $n$ trials of an experiment where each trial can result in either "success" or "failure", the trials are independent, and $p$ is the probability of success for any one trial. A generalization of the binomial distribution to more than two categories is the *multinomial distribution*.

Suppose that one repeats a series of $n$ independent trials of an experiment where each trial can result in one of $k$ possible outcomes. Further suppose that the probability that outcome $i$ occurs on any one trial is given by $p_i$ where $\sum_{i=1}^{k} p_i = 1$. For $i = 1, 2, \ldots, k$, let $X_i$ be the number of trials (out of $n$) that result in outcome $i$. Then the random vector $(X_1, X_2, \ldots, X_k)$ is said to have a **multinomial distribution**. It is eay to show/argue that the joint probabilities for the $X_i$ are given by

$$P(X_1 = x_1, X_2 = x_2, \ldots, X_k = x_k) = \frac{n!}{x_1! x_2! \cdots x_k!} \, p_1^{x_1} p_2^{x_2} \cdots p_k^{x_k}$$

as long as $\sum_{i=1}^{k} x_i = n$. Otherwise, the probability is zero.

$X_i$ is the observed number of outcomes in category $i$. The expected number in category $i$ is $\mathsf{E}[X_i] = np_i$.

**Claim:** The quantity

$$\sum_{i=1}^{k} \frac{(obs_i - exp_i)^2}{exp_i} := \sum_{i=1}^{k} \frac{(X_i - np_i)^2}{np_i}$$

has, for "large enough" expectations in each category, approximately a $\chi^2$ distribution with $k - 1$ degrees of freedom. A rule of thumb for "large enough" is usually to expect at least 5 outcomes in each category. i.e.: $np_i \geq 5$ for $i = 1, 2, \ldots, k$.

We now give a sketch of the proof of this claim in the binomial case ($k = 2$) that should be readable by anyone who has had a course in MathStat. If you haven't, feel free to skip the sketch.

In the binomial case, there are two categories: success and failure. We are used to looking at the random variable $X$ defined as the total number of successes in $n$ trials. In the multinomial setting, this would correspond to a random variable $X_1$ while the number of failures would be an $X_2$ where $X_2 = n - X_1$. We will denote the sucees probability $p$ by $p_1$ and the failure probability by $p_2 = 1 - p_1$.

Note that:

- The binomial random variable $X_1$, when though of as a a sum of Bernoulli random variables, adheres to the Central Limit Theorem. That is, when properly standardized by it's mean $\mathsf{E}[X_1] = np_1$ and standard deviation $\sqrt{Var[X_1]} = \sqrt{np(1-p)}$, we have

$$\frac{X_1 - np_1}{\sqrt{np_1(1-p_1)}} \xrightarrow{d} N(0,1).$$

- Convergence in distribution is preserved through continuous functions. Using this and the fact that a standard normal random variable squared has a $\chi^2(1)$ distribution, we have

$$\frac{(X_1 - np_1)}{np_1(1-p_1)} \xrightarrow{d} \chi^2(1).$$

  Thus, we say that the left-hand side has approximately a $\chi^2(1)$ distribution for large enough $n$.

- To get to the Claim, note that

$$\begin{aligned}
\tfrac{(X_1-np_1)^2}{np_1(1-p_1)} &= \tfrac{(X_1-np_1)^2}{np_1} + \tfrac{(X_1-np_1)^2}{n(1-p_1)} \\
&= \tfrac{(X_1-np_1)^2}{np_1} + \tfrac{[(n-X_1)-n(1-p_1)]^2}{n(1-p_1)} \\
&= \tfrac{(X_1-np_1)^2}{np_1} + \tfrac{(X_2-np_2)^2}{np_2}
\end{aligned}$$

  which is the quantity given in the Claim. Thus

$$\frac{(X_1 - np_1)^2}{np_1} + \frac{(X_2 - np_2)^2}{np_2} = \frac{(X_1 - np_1)^2}{np_1(1-p_1)} \overset{approx}{\sim} \chi^2(1)$$

  for large enough $n$.

# Bibliography

[1] Z.W. Birnbaum. "Numerical Tabulation of Kolmogorov's Statistic for Finite Sample Size". In: *Journal of the American Statistical Association* 47 (1952), pp. 425–441 (cit. on p. 28).

[2] A. Kolmogorov. "Sulla determinazione empirica di una legge di distribuzionc". In: *1st. Ital. Attuari. G.* 4 (1933), pp. 1–11 (cit. on p. 28).

[3] R. Simard and P. L'Ecuyer. "Computing the Two-Sided Kolmogorov-Smirnov Distribution". In: *Journal of Statistical Software, Articles* 39.11 (2011), pp. 1–18 (cit. on p. 28).