Contents

1 Probability review 1
1.1 Introduction .................................................. 1
1.2 Random variables and distributions ............................. 3
1.3 Convergence of random variables ............................ 6

2 Monte Carlo methods 11
2.1 Introduction .................................................... 11
2.2 Pseudo-random numbers ....................................... 12
2.3 Tests for studying random data ............................... 14
2.4 Monte Carlo integration ...................................... 16
  2.4.1 Basic Monte Carlo integration .......................... 16
  2.4.2 Variance reduction ...................................... 19

Bibliography 23
Chapter 2

Monte Carlo methods

2.1 Introduction

Monte Carlo methods are numerical methods that use random numbers to compute some quantity of interest, typically by creating a random variable whose expected value is the desired quantity. See [8] for historical background.

Example 2.1 The value of \( \pi \) can be approximated using a Monte Carlo method. Consider the square \([0,1]^2\) and the inscribed quarter circle of radius 1 centered at the origin. Next, we draw \( N \) points \((x_i, y_i)_{i=1}^{N}\) where \( x_i \) and \( y_i, i = 1, \ldots, N \), are taken from a uniform distribution of the unit interval. Now, the ratio of the area of the quarter circle to that of the square, \( \pi/4 \), should be equal to the ratio of the number of points in the quarter circle to the total number of points \( N \). Figure 2.1 displays the distribution of 100,000 such points, left, and the error in the resulting approximation of \( \pi \).

The previous example illustrates two important properties of Monte Carlo methods.
it is important to have “good” random numbers (a non-uniform distribution would not have worked);

• the methods converge slowly as the number of points is increased.

Many more examples are discussed below.

2.2 Pseudo-random numbers

Monte Carlo methods require “random numbers”. Typically, this is done through the use of deterministic algorithms which produce sequences of numbers that appear to be random. Any such algorithm will then be viewed as a pseudo-random number generator if it passes a certain numbers of statistical tests. In other words and for practical purposes, the algorithm is then presumed innocent until proven guilty (see [5], p.41; a discussion of the meaning of “random sequence” can also be found there [5], pp. 149–183).

Let us consider methods for generating sequences of random real numbers uniformly distributed between zero and one. Since real numbers are only approximated in any floating point representation format, one usually generates integers $I_n$ between 0 and some maximal value $m$ (for instance the word size of the computer) and then consider the fraction

$$X_n = I_n/m.$$  \hspace{1cm} (2.1)

**Example 2.2** Linear congruential method: for a given modulus $m$, $m > 0$, one chooses a multiplier $a$, $0 \leq a < m$, an increment $c$, $0 \leq c < m$ and a starting value $I_0$, $0 \leq I_0 < m$ and constructs the sequence

$$I_{n+1} = (aI_n + c) \mod m, \hspace{0.5cm} n = 0, 1, \ldots$$  \hspace{1cm} (2.2)

Taking $c = 0$ to simplify, one way to choose $a$ and $m$ is to guarantee that (i) \{ $I_n$ \} has full period $m - 1$ for any seed $I_0$, (ii) the full period $\ldots, I_1, I_2, \ldots, I_{m-1}, \ldots$ is random (see exercise 2.3), (iii) the algorithm can be efficiently implemented. The values $m = 2^{31} - 1 = 2147483647$, $a = 7^5 = 16807$ and $c = 0$ can be shown to satisfy all three criteria (on 32 bit computers), see [10], and were for many years adopted as MATLAB uniform random number function `rand` (MATLAB version 4).

As vividly described in [9], the default random number generator for MATLAB version 5 through 7.3 was completely different from Example 2.2: Marsaglia’s generator is not congruential, is based on [6] and is specifically designed to produce floating-point values. The period of this generator is about $2^{1430}$. Starting with MATLAB version 7.4, the default random generator was changed again, this time to the so-called Mersenne twister [7]. This latter generator has a period of length $2^{19937} - 1$.

All the previous examples are related to uniform distributions. How does one sample from non-uniform distributions? Assume we want to sample from a given
density \( f \). The question becomes: if \( Y \) has uniform distribution, can we find a function \( \Phi \) such that \( X = \Phi(Y) \) admits \( f \) as its density function?

For any function \( g \), we observe
\[
E(g(X)) = \int g(x)f(x) \, dx = \int g(F^{-1}(y)) \, dy,
\]
by the change of variable \( y = F(x) \) where \( F \) is the distribution function corresponding to \( f \), i.e., \( F'(x) = f(x) \). In other words, we can just take \( \Phi(y) = F^{-1}(y) \) to answer the above question. Note that it may be difficult to compute the inverse, see Exercise 2.6. Exercise 2.7 discusses a method that does not require taking the inverse of \( F \) but is limited to normal distributions.

Yet another method for sampling from an arbitrary density \( f \) is due to von Neumann and is usually referred to as the acceptance-rejection procedure. In its simplest form, the method is as follows

1. select \( x_i \) from a uniform distribution on the domain of \( f \),
2. select \( y_i \) from a uniform distribution on the range of \( f \),
3. accept \( x_i \) if \( y_i \leq f(x_i) \).

The proportion of points selected that fall in a small interval along the \( x \)-axis is then proportional to the height of the curve on that interval. This ensures that the selected numbers are distributed according to \( f \). Figure 2.2 illustrates the process for a standard normal distribution \( f \).

![Figure 2.2.](image)

**Figure 2.2.** Histogram for acceptance-rejection process to sample from a standard normal distribution using a uniformly distributed number generator; 12,760 accepted points out of a total of 100,000.
2.3 Tests for studying random data

When is a sequence sufficiently random? What does that even mean? The answer to this is surprisingly complex, see Exercise 2.3. The theory of statistics provides some quantitative tests for randomness. We will limit ourselves to the $\chi^2$ test which is the most well known statistical test. It is limited to discrete random variables.

Example 2.3 Assume we throw two fair dice. There are 36 possible outcomes the probability of which is displayed in Table 2.1. If we throw the dice $n$ times, we expect to obtain the total $T$ approximately $np_T$ times. Let’s take $n = 216$ and run the experiment using the MATLAB default $\text{rand}$ function to mimic the fair dice and generate the throws. The results are displayed in Table 2.2. Are the dice loaded or, here, is $\text{rand}$ random enough? ■

<table>
<thead>
<tr>
<th>$T$</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_T$</td>
<td>$\frac{1}{36}$</td>
<td>$\frac{1}{18}$</td>
<td>$\frac{1}{12}$</td>
<td>$\frac{5}{36}$</td>
<td>$\frac{1}{6}$</td>
<td>$\frac{5}{36}$</td>
<td>$\frac{1}{9}$</td>
<td>$\frac{1}{12}$</td>
<td>$\frac{1}{18}$</td>
<td>$\frac{5}{36}$</td>
<td>$\frac{1}{9}$</td>
</tr>
</tbody>
</table>

Table 2.1. Probability $p_T$ of obtaining a given total $T$ from throwing two fair dice.

<table>
<thead>
<tr>
<th>$T$</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>$np_T$</td>
<td>6</td>
<td>12</td>
<td>18</td>
<td>24</td>
<td>30</td>
<td>36</td>
<td>30</td>
<td>24</td>
<td>18</td>
<td>12</td>
<td>6</td>
</tr>
<tr>
<td>$Y_T$</td>
<td>7</td>
<td>17</td>
<td>13</td>
<td>16</td>
<td>27</td>
<td>43</td>
<td>34</td>
<td>22</td>
<td>16</td>
<td>11</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 2.2. Expected and observed totals after throwing the dice 216 times.

Let’s generalize the question from the previous example. We assume that we have $n$ independent observations, each of them falling into $k$ categories. In Example 2.3, $k = 11$. Let $Y_T$ be the number of observations that correspond to category $T$, $T = 1, \ldots, k$, and let $p_T$ the probability that each observation falls into category $T$. We have of course

$$\sum_{T=1}^{k} p_T = 1, \quad \text{and} \quad \sum_{T=1}^{k} Y_T = n. \quad (2.3)$$

The probability to have $Y_1 = y_1, \ldots, Y_k = y_k$ is given by the multinomial distribution (1.3)

$$\frac{n!}{y_1! \ldots y_k!} p_1^{y_1} \cdots p_k^{y_k}. \quad (2.4)$$
2.3. Tests for studying random data

Assume that $Y_T$ follows a Poisson distribution (we’ll justify this a posteriori below), i.e., its mass density (probability that $Y_T$ has the value $y_T$) is

$$f(y_T) = \frac{(np_T)^{y_T}}{y_T!} e^{-np_T}, \quad (2.5)$$

where $np_T$ is the expected value of $Y_T$. Assuming the $Y_T$’s to be independent, the probability that $(Y_1, \ldots, Y_k) = (y_1, \ldots, y_k)$ is then

$$\prod_{T=1}^{k} \frac{(np_T)^{y_T}}{y_T!} e^{-np_T}.$$

On the other hand, the probability that $\sum_{T=1}^{k} Y_T = n$ is

$$\sum_{y_1 + \cdots + y_k = n} \prod_{T=1}^{k} \frac{(np_T)^{y_T}}{y_T!} e^{-np_T} = \frac{n^n}{n!} e^{-n}.$$

Now, the probability that $(Y_1, \ldots, Y_k) = (y_1, \ldots, y_k)$ under the condition $\sum_{T=1}^{k} Y_T = n$ is thus

$$\prod_{T=1}^{k} \frac{(np_T)^{y_T}}{y_T!} e^{-np_T} \frac{n^n}{n!} e^{-n},$$

which is equal to (2.4)! The conclusion of this digression is that the $Y_T$’s can be regarded as Poisson variables which are independent except for the condition $\sum_{T=1}^{k} Y_T = n$.

Our goal is to measure how likely it is for each of the $Y_T$ to wander away for its expected value $np_T$. To this end, we consider

$$V = \sum_{T=1}^{k} \frac{(Y_T - np_T)^2}{np_T} = \frac{1}{n} \sum_{T=1}^{k} \left( \frac{Y_T^2}{p_T} \right) - n, \quad (2.6)$$

where the last relation results from (2.3). Consider now the variable $Z_T = \frac{Y_T - np_T}{\sqrt{np_T}}$, $T = 1, \ldots, k$. The above quantity $V$ becomes $V = \sum_{T=1}^{k} Z_T^2$ while the condition $\sum_{T=1}^{k} Y_T = n$ from (2.3) is now

$$\sum_{T=1}^{k} \sqrt{p_T} Z_T = 0. \quad (2.7)$$

We denote by $\mathcal{H}$ the hyperplane of dimension of dimension $k - 1$ defined by (2.7). By (2.5), since $Y_T$ has mean and variance equal to $np_T$, then for large $n$, each of the $Z_T$’s is approximately $N(0, 1)$, see Exercise 1.11. Therefore, points in a differential volume $dz_2 \ldots dz_k$ of $\mathcal{H}$ occur with probability approximately proportional to $\exp((- \sum_{T=1}^{k} z_T^2)/2)$. So, for $n$ large enough, the probability that $V \leq v$ is then given by

$$P(V \leq v) \approx \frac{\int_{\sum_{T=1}^{k} z_T^2 \leq v} \exp((- \sum_{T=1}^{k} z_T^2)/2) \, dz_2 \ldots dz_k}{\int_{z \in \mathcal{H}} \exp((- \sum_{T=1}^{k} z_T^2)/2) \, dz_2 \ldots dz_k}.$$
Let’s rewrite the above integrals using generalized spherical coordinates, i.e., we set
\[
\begin{align*}
  z_2 &= \chi \cos \theta_1, \\
  z_3 &= \chi \sin \theta_1 \cos \theta_2, \\
  &\ldots \ldots, \\
  z_{k-1} &= \chi \sin \theta_1 \sin \theta_2 \ldots \sin \theta_{k-3} \cos \theta_{k-2}, \\
  z_k &= \chi \sin \theta_1 \sin \theta_2 \ldots \sin \theta_{k-3} \sin \theta_{k-2},
\end{align*}
\]
where \( \chi > 0 \) is the radius and where the angles satisfy \( 0 < \theta_T < \pi, T = 1, \ldots, k - 3 \) and \( 0 < \theta_{k-2} < 2\pi \). The previous expression takes the form
\[
P(V \leq v) \approx \frac{\int_{\chi^2 \leq v} \exp(-\chi^2/2) \phi(\theta_1, \ldots, \theta_{k-2}) \chi^{k-2} d\chi d\theta_1 \ldots d\theta_{k-2}}{\int \exp(-\chi^2/2) \phi(\theta_1, \ldots, \theta_{k-2}) \chi^{k-2} d\chi d\theta_1 \ldots d\theta_{k-2}},
\]
for some function \( \phi \). Note that the contribution from integration over the angular variables cancels out between numerator and denominator, leading to
\[
P(V \leq v) \approx \frac{\int_{\chi^2 \leq v} \exp(-\chi^2/2) \chi^{k-2} d\chi}{\int \exp(-\chi^2/2) \chi^{k-2} d\chi}.
\]
This is the so-called chi-square distribution with \( k - 1 \) degree of freedom. And yes, it got its name because in the original paper [11], Pearson used the symbol \( \chi \) for the radius. Using the change of variable \( t = \chi^2/2 \) together with standard notation, i.e., the gamma and incomplete gamma functions defined respectively by
\[
\Gamma(x) = \int_0^\infty e^{-t} t^{x-1} dt \quad \text{and} \quad \gamma(\alpha, x) = \int_0^x e^{-t} t^{\alpha-1} dt,
\]
the distribution can be rewritten
\[
P(V \leq v) \approx \frac{\gamma(\frac{k-1}{2}, \frac{v}{2})}{\Gamma(\frac{k-1}{2})},
\]
which corresponds also to \( (1.5) \) with \( k - 1 \) instead \( k \).

How does one use this?

Let’s go back to Example 2.3. First, we need to compute \( V \) corresponding to Table 2.2. Using \( (2.6) \), we find \( V \approx 11.64 \). Is this reasonable? The values of the right hand side of \( (2.9) \) are often consulted from tables. In the present case, \( k = 11 \) and the right hand side of \( (2.9) \) is displayed in Figure 2.3. As per Figure 2.3, the value 11.64 falls close to the 50% mark and so the observations from Table 2.2 appear sufficiently random with respect to this test. Other experiments may lead to different results however, see Exercise 2.4.

### 2.4 Monte Carlo integration

#### 2.4.1 Basic Monte Carlo integration

Consider the integral
\[
I = \int_0^1 g(x) \, dx,
\]
using the Monte Carlo method.

Let's rewrite the above integrals using generalized spherical coordinates, i.e., we set
\[
\begin{align*}
  z_2 &= \chi \cos \theta_1, \\
  z_3 &= \chi \sin \theta_1 \cos \theta_2, \\
  &\ldots \ldots, \\
  z_{k-1} &= \chi \sin \theta_1 \sin \theta_2 \ldots \sin \theta_{k-3} \cos \theta_{k-2}, \\
  z_k &= \chi \sin \theta_1 \sin \theta_2 \ldots \sin \theta_{k-3} \sin \theta_{k-2},
\end{align*}
\]
where \( \chi > 0 \) is the radius and where the angles satisfy \( 0 < \theta_T < \pi, T = 1, \ldots, k - 3 \) and \( 0 < \theta_{k-2} < 2\pi \). The previous expression takes the form
\[
P(V \leq v) \approx \frac{\int_{\chi^2 \leq v} \exp(-\chi^2/2) \phi(\theta_1, \ldots, \theta_{k-2}) \chi^{k-2} d\chi d\theta_1 \ldots d\theta_{k-2}}{\int \exp(-\chi^2/2) \phi(\theta_1, \ldots, \theta_{k-2}) \chi^{k-2} d\chi d\theta_1 \ldots d\theta_{k-2}},
\]
for some function \( \phi \). Note that the contribution from integration over the angular variables cancels out between numerator and denominator, leading to
\[
P(V \leq v) \approx \frac{\int_{\chi^2 \leq v} \exp(-\chi^2/2) \chi^{k-2} d\chi}{\int \exp(-\chi^2/2) \chi^{k-2} d\chi}.
\]
This is the so-called chi-square distribution with \( k - 1 \) degree of freedom. And yes, it got its name because in the original paper [11], Pearson used the symbol \( \chi \) for the radius. Using the change of variable \( t = \chi^2/2 \) together with standard notation, i.e., the gamma and incomplete gamma functions defined respectively by
\[
\Gamma(x) = \int_0^\infty e^{-t} t^{x-1} dt \quad \text{and} \quad \gamma(\alpha, x) = \int_0^x e^{-t} t^{\alpha-1} dt,
\]
the distribution can be rewritten
\[
P(V \leq v) \approx \frac{\gamma(\frac{k-1}{2}, \frac{v}{2})}{\Gamma(\frac{k-1}{2})},
\]
which corresponds also to \( (1.5) \) with \( k - 1 \) instead \( k \).

How does one use this?

Let’s go back to Example 2.3. First, we need to compute \( V \) corresponding to Table 2.2. Using \( (2.6) \), we find \( V \approx 11.64 \). Is this reasonable? The values of the right hand side of \( (2.9) \) are often consulted from tables. In the present case, \( k = 11 \) and the right hand side of \( (2.9) \) is displayed in Figure 2.3. As per Figure 2.3, the value 11.64 falls close to the 50% mark and so the observations from Table 2.2 appear sufficiently random with respect to this test. Other experiments may lead to different results however, see Exercise 2.4.

### 2.4 Monte Carlo integration

#### 2.4.1 Basic Monte Carlo integration

Consider the integral
\[
I = \int_0^1 g(x) \, dx,
\]
for some given smooth enough function $g$. Now consider a random variable $X$ uniformly distributed on $[0, 1]$. The above integral can be regarded as the expected value of the random variable $g(X)$, i.e.

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

Consider now $N$ independent random variables $X_1, \ldots, X_N$ uniformly distributed in $[0, 1]$ and define the estimator

$$I_N = \frac{1}{N} \sum_{i=1}^N g(X_i).$$

A realization of $I_N$ is a sample mean of $g(X)$, i.e.,

$$\frac{1}{N} \sum_{i=1}^N g(x_i)$$

where the $x_i$’s are $N$ points uniformly distributed in $[0, 1]$ is an estimate of $I$. In fact, on average, $I_N$ gives the correct result regardless of $N$. Indeed

$$E(I_N) = \int_0^1 \frac{1}{N} \sum_{i=1}^N g(x) \, dx = \int_0^1 g(x) \, dx = I.$$

In what sense and how fast do realizations of $I_N$ converge to $I$? In order to answer, we need the variance of $I_N$. We have by elementary properties of the variance

$$\text{var}(I_N) = \text{var} \left( \frac{1}{N} \sum_{i=1}^N g(X_i) \right) = \frac{1}{N^2} \text{var} \left( \sum_{i=1}^N g(X_i) \right) = \frac{1}{N^2} \sum_{i=1}^N \text{var}(g(X_i))$$
...and thus
\[ \text{var}(I_N) = \frac{1}{N} \mathcal{V}, \]

where \( \mathcal{V} = \text{var}(g(X)) = E(g(X)^2) - E(g(X))^2 = \int_0^1 g^2(x) \, dx - I^2. \) As a direct consequence of Chebyshev’s inequality (see Lemma 1.12), we then obtain
\[ P\left(|I_N - I| > \frac{\delta}{\sqrt{N}}\right) \leq \frac{\mathcal{V}}{\delta^2}, \]

for any \( \delta > 0. \) Therefore, \( I_N \) converges to \( I \) in probability with rate \( O(N^{-1/2}) \) as \( N \to \infty. \)

The Central Limit Theorem 1.16 can be used to describe the size and statistical properties of Monte Carlo integration errors, leading to

**Theorem 2.4 (Monte Carlo integration errors).** For \( N \) large,
\[ I_N - I \approx \sqrt{\frac{\mathcal{V}}{N}}, \]

where \( \mathcal{V} \) is the variance of the integral of \( g, \) i.e., \( \mathcal{V} = \int_0^1 g^2(x) \, dx - \left( \int_0^1 g(x) \, dx \right)^2 \)
and \( N \) is a \( N(0,1) \) random variable (standard normal).

Several remarks are in order. First, the above probabilistic result is not a “usual error estimate” in that it does not provide an absolute upper bound. It is also “tight”: this is an equality. Second, it can be seen (exercise) that the spatial dimension does not in fact play any fundamental role, neither in the above derivation nor in the convergence result. (Note however that dimension may enter through the variance term in the above estimate.) The method is therefore very promising for high dimensional problems where standard deterministic quadrature methods perform poorly.

Exercise 2.5 “verifies” some of the above properties in practice.

There are two ways to reduce the error of a Monte Carlo integration method. One way is to increase the number of samples \( N. \) The other way is to reduce the variance \( \mathcal{V}. \) Before we look at specific variance reduction methods, it is useful to look at how one could sample from non uniform variables through transformation of uniform variables.

Let \( X \) be a random variable with density \( f. \) The expected value of \( g(X) \) is
\[ E(g(X)) = \int g(x) f(x) \, dx = I. \]

If \( \{x_n\} \) is a sequence of random numbers distributed according to the density \( f, \) then the sample approximation to \( I \) is
\[ I_N = \frac{1}{N} \sum_{i=1}^N g(x_i). \]
Proceeding as above, we then obtain

\[ I_N - I \approx \sqrt{\frac{\mathcal{V}}{N}} N, \]

where now \( \mathcal{V} = \int (g(x) - \bar{g})^2 f(x) \, dx \) where \( \bar{g} = I \).

### 2.4.2 Variance reduction

Variance reduction methods aim at speeding up the convergence of Monte Carlo methods.

**Stratification**

Stratification combines the idea of composite quadrature rules with the Monte Carlo method. More precisely, the domain of integration \( \Omega = (0, 1) \) is split into \( M \) pieces \( \Omega_k = \left[ \frac{k-1}{M}, \frac{k}{M} \right], k = 1, \ldots, M \), and the integral is rewritten as

\[
I = \int_0^1 g(x) \, dx = \sum_{k=1}^M \int_{\Omega_k} g(x) \, dx = \frac{1}{M} \sum_{k=1}^M \frac{1}{|\Omega_k|} \int_{\Omega_k} g(x) \, dx,
\]

since \(|\Omega_k| = 1/M\). Now we can approximate the average of \( g \) over \( \Omega_k \) by a sample average as above, i.e.,

\[
\frac{1}{|\Omega_k|} \int_{\Omega_k} g(x) \, dx \approx \frac{1}{N/M} \sum_{i=1}^{N/M} g(x^k_i),
\]

where for each \( k \) we use \( N/M \) points \( x^k_i, i = 1, \ldots, N/M \), uniformly distributed in \( \Omega_k \). This leads to the new Monte Carlo estimator

\[
I^*_N = \frac{1}{N} \sum_{k=1}^M \sum_{i=1}^{N/M} g(x^k_i).
\]

Proceeding as above, the error for this stratified quadrature is

\[ I^*_N - I \approx \sqrt{\frac{\mathcal{V}^*}{N}} N, \]

where \( \mathcal{V}^* \) is the variance of the composite integral of \( g \), i.e.,

\[ \mathcal{V}^* = \sum_{k=1}^M \int_{\Omega_k} (g(x) - \bar{g}_k)^2 \, dx, \]

where \( \bar{g}_k = \frac{1}{|\Omega_k|} \int_{\Omega_k} g(x) \, dx \). Let’s compare the two variances \( \mathcal{V} \) and \( \mathcal{V}^* \). First, we notice that the minimizer of the quadratic form

\[ J(c) = \int_{\Omega_k} (g(x) - c)^2 \, dx, \]
is \( c = \bar{g}_k \) (exercise). Therefore

\[
\mathcal{V}^s = \sum_{k=1}^{M} \int_{\Omega_k} (g(x) - \bar{g}_k)^2 \, dx \leq \sum_{k=1}^{M} \int_{\Omega_k} \left( g(x) - \int_0^1 g(y) \, dy \right)^2 \, dx = \mathcal{V}.
\]

Consequently, stratified Monte Carlo quadrature is always better than the unstratified quadrature.

**Importance sampling**

As the name suggests, the idea is to put more points where the function is “important”, i.e., large, by drawing them from a well chosen distribution \( f \) as opposed to a uniform distribution as above. We then have

\[
I = \int_0^1 g(x) \, dx = \int_0^1 \frac{g(x)}{f(x)} f(x) \, dx = E \left( \frac{g(X)}{f(X)} \right),
\]

where \( X \) is now a random variable that admits \( f \) as density. The corresponding Monte Carlo estimator is then

\[
I_N^{is} = \frac{1}{N} \sum_{i=1}^{N} \frac{g(X_i)}{f(X_i)},
\]

where the \( X_i, i = 1, \ldots, N \), are i.i.d. random variables admitting \( f \) as density. Based on the previous considerations, a natural way to choose \( f \) is to minimize the variance of \( I_N^{is} \), i.e.

\[
\text{var}(I_N^{is}) = \frac{1}{N} \int_0^1 \left( \frac{g(x)}{f(x)} - E \left( \frac{g(X)}{f(X)} \right) \right)^2 f(x) \, dx.
\]

Taking the density \( f \) proportional to \( g \), i.e., \( f(x) = \alpha g(x) \) clearly minimizes \( \text{var}(I_N^{is}) \) since then \( \text{var}(I_N^{is}) = 0 \). Now, note that to compute the density \( f \) we need the normalization constant \( \alpha \) which in turn requires computing \( \int_0^1 g(x) \, dx \) which was our goal in the first place.

While choosing the importance sampling function \( f \) exactly proportional to \( g \) (or more precisely \(|g|\)) is not feasible in practice, it is clear to having \( f \) “close” to being proportional to \( g \) would be beneficial. In short, a good importance sampling function \( f \) satisfies

1. \( f(x) > 0 \) whenever \( g(x) \neq 0 \),

2. \( f \) is close to being proportional to \(|g|\),

3. it is easy to compute the density \( f \) and simulate values from it.

Note that fulfilling the above requirements may be challenging.
2.4. Monte Carlo integration

Example 2.5 We use Monte Carlo integration to compute \( \frac{1}{\sqrt{\pi}} \int_{-10}^{10} e^{-x^2} \, dx = \text{erf}(10) \). Figure 2.4 shows the histograms of 5000 Monte Carlo estimates with \( N = 1000 \). On the left, the basic Monte Carlo integration method is used while on the right, the importance sampling function is the \( t \)-distribution with \( k = 3 \) degree of freedom, i.e.,

\[
f(x) = \frac{\Gamma\left(\frac{k+1}{2}\right)}{\sqrt{k\pi}\Gamma\left(\frac{k}{2}\right)} \left(1 + \frac{x^2}{k}\right)^{-\frac{k+1}{2}}.
\]

Figure 2.4. Histogram of 5000 Monte Carlo estimates with \( N = 1000 \) of \( \frac{1}{\sqrt{\pi}} \int_{-10}^{10} e^{-x^2} \, dx = \text{erf}(10) \) which is extremely close to 1; left: standard integration, right: importance sampling with \( t \)-distribution with 3 degrees of freedom.

Exercises for Chapter 2

2.1 For \( m = 13 \), find the multipliers \( a, 0 \leq a < m \) that lead to full period sequences (for any seed \( I_0 \)).

2.2 The infamous RANDU (or RND) generator that was part of the Scientific Subroutine Package on IBM mainframe computers in the 1960s [3]; this generator corresponds to the choices \( a = 65539 \), \( c = 0 \) and \( m = 2^{31} \) in (2.2). Show that modulo \( 2^{31} \),

\[
I_{n+2} - 6I_{n+1} + 9I_n = 0.
\]

Implement RANDU and verify graphically its severe lack of equi-distribution by creating a three dimensional plot of the first 10,000 points generated for some odd \( I_0 \) of your choice (more precisely, plot \( \{(I_{n-1}, I_n, I_{n+1})/m, n = 1, \ldots, 10,000\} \)).

2.3 Is 5 a random number? Read Section 3.5 [5].

2.4 1. Repeat the experiment that lead to Table 2.2 up to five times. Use the following classification for the range of \( V [5] \): if \( V \) is less than the 1% entry
or more than the 99\% entry, the numbers are rejected as not sufficiently random, if \( V \) is between the 1\% and 5\% entries or between the 95\% and 99\% entries, the numbers are suspect and if they are between the 5\% and 10\% entries or between the 90\% and 95\% entries, they are almost suspect. Does the default MATLAB function \texttt{rand} still pass the test?

2. Repeat with the other random number generators from Section 2.2 which can still be used by changing the arguments of \texttt{rand}. Do you observe any significant differences? Is this a good test?

3. Repeat the last with RANDU. Comment.

2.5 1. If not done already, download and install GSL, the GNU Scientific Library http://www.gnu.org/software/gsl/

2. Use GSL’s plain Monte Carlo integration routine to compute

\[
\left( \frac{2}{\sqrt{\pi}} \right)^d \int_0^2 \cdots \int_0^2 \exp \left( -x_1^2 \cdots - x_d^2 \right) \, dx_1 \cdots dx_d,
\]

where \( d \) is the spatial dimension. Hints: examples are given in the GSL online documentation. Further, for comparison purposes, use the “exact” value obtained by expressing the above function as \((\text{erf}(2))^d\).

3. Given a budget of 500,000 function evaluations, what is the largest dimension \( d \) for which the routine gives a reasonable answer? Justify your answer.

2.6 1. Show that the distribution function of a standard normal random variable \((N(0, 1))\) is given by \( F(x) = \frac{1}{2} + \frac{1}{2} \text{erf}(x/\sqrt{2}) \) where the error function \( \text{erf} \) is defined by \( \text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} \, dt \).

2. Show that \( X = \sqrt{2} \text{erf}^{-1}(2Y - 1) \) is \( N(0, 1) \) if \( Y \) is uniformly distributed. Discuss how to compute \( \text{erf}^{-1} \).

2.7 Let \( Y_1 \) and \( Y_2 \) be two independent random variables uniformly distributed in \([0, 1]\). Show that

\[
X_1 = \sqrt{-2 \log Y_1 \cos(2\pi Y_2)} \quad \text{and} \quad X_2 = \sqrt{-2 \log Y_1 \sin(2\pi Y_2)},
\]

are standard normal variables. Hint: invert the transform to get \( Y_1 \) and \( Y_2 \) and compute the Jacobian. This scheme is usually referred to as the Box-Muller method.

2.8 1. Investigate the relationship between the \( t \)-distribution with \( k \) degrees of freedom introduced in Example 2.5 and the normal distribution.

2. Implement of the two methods discussed in Example 2.5 and repeat the test done there. Hint: you can for instance use the acceptance-rejection methods to sample from the \( t \)-distribution.
Bibliography


[11] KARL PEARSON, On the criterion that a given system of deviations from the probable in the case of correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling, Philosophical Mag., 50 (1900), pp. 157–175.