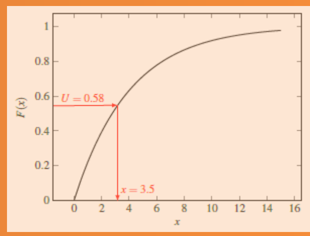
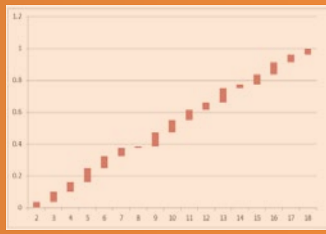
$$\begin{aligned}
F(x) &= \int_0^x f(z)\mathrm{d}z \\
&= \int_0^x \frac{1}{\beta}\exp\left(\frac{-z}{\beta}\right)\mathrm{d}z \\
&= -\exp\frac{-z}{\beta}\Big|_0^x \\
&= 1-\exp\frac{-x}{\beta} \\
&= U. \\
X &= F^{-1}(U) \\
&= -\beta\ln(1-U).
\end{aligned}$$





0.71464	0.41472	0.45395	0.20527	0.43824
0.72700	0.95528	0.93084	0.92111	0.25805
0.36477	0.94243	0.17866	0.36138	0.00827
0.46218	0.20972	0.22791	0.65858	0.87798
0.36458	0.49537	0.43067	0.65950	0.37429
0.26494	0.17691	0.51296	0.10539	0.71491
0.24901	0.72310	0.83471	0.64589	0.18068
0.27080	0.02045	0.39797	0.51331	0.95084
0.14335	0.24891	0.58877	0.91678	0.20341
0.39069	0.71483	0.95367	0.23682	0.21499
0.28527	0.30922	0.23930	0.67698	0.88679
0.05985	0.62431	0.65183	0.00875	0.52420
0.57042	0.16075	0.93422	0.89802	0.44419
0.55230	0.03058	0.35627	0.19582	0.14914
0.58464	0.31230	0.62784	0.77463	0.05814

3. Random numbers and random variates

THIS chapter covers the topics of random number generation and random variate generation. We need random numbers to induce (pseudo)randomness in our stochastic models, and we use these numbers when generating observations from distributions. These observations are used to replicate stochastic events, for example the inter-arrivals of entities in a process. The content of the chapter covers the main principles only; for in-depth discussions see for example Banks (1998) and Law (2015).

3.1 Random number generation

Stochastic processes originate due to randomness. *Randomness* is a thought-provoking concept, also from a philosophical point of view. It leads to uncertainty, which in turn can be classified as *objective uncertainty* and *subjective uncertainty*. Objective uncertainty is due to inherent randomness (in a process or system), while subjective uncertainty follows from lack of information (Denardo, 2002).

Since random numbers form the core of a stochastic simulation, we must investigate how a simulation software program (or any computer program) generates random numbers. This exercise seems to be unnecessary, because the general response is ‘Well, the computer simply does it’. A computer only follows instructions, and a ‘program’, ‘method’ or ‘recipe’ is therefore required to generate random numbers. Because of this ‘recipe’ followed, the random numbers generated are not truly random, and we use the term ‘pseudo-random’ when referring to these

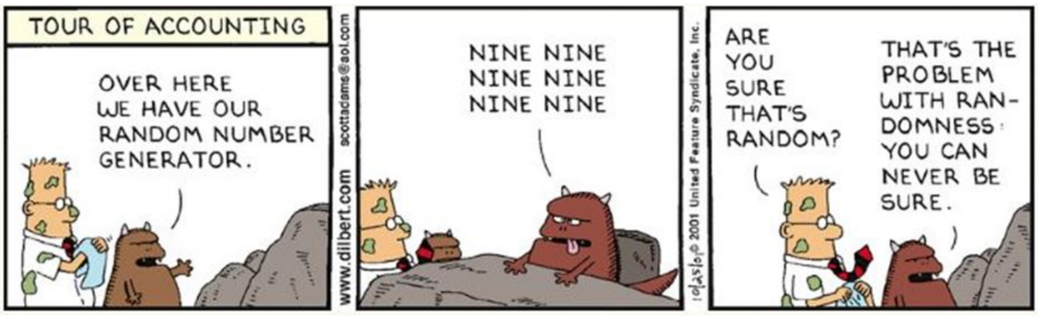


Figure 3.1: Randomness is not always well-understood

computer-generated numbers.

3.2 Basics of random number generators

There exist many types of random number generators (RNGs), for example

- the midsquare method,
- lagged Fibonacci generator (Knuth, 1997),
- linear congruential generators (LCGs) (based on the work of Derrick Lehmer (see Law (2015)), and the
- permuted congruential generator (O'Neill, 2014).

The LCGs produce a sequence of integers Z_i using

$$Z_i = (aZ_{i-1} + c) \bmod m \quad (3.1)$$

where m is the modulus, a is a multiplier, c is an increment and Z_0 is the seed or starting value. It is required that $m > 0$, $a < m$, $c < m$ and $Z_0 < m$. These Z_i are then converted to $U(0,1)$ numbers using $U_i = Z_i/m$. The LCG in (3.1) is *recursive*, Z_1 is thus determined by Z_0, Z_2 is determined by Z_1, and so on, which means that once we have selected the constants, the random number stream is completely defined. These random numbers are therefore called *pseudorandom* numbers. It is also necessary to specify Z_0, the *seed number* of the RNG.

■ **Example 3.1** Generate five random numbers using (3.1) with $a = 3$, $c = 0$, $m = 23$ and $Z_0 = 7$. Table the resulting calculations.

Answer:

i	Z_i	$U_i = Z_i/m$
0	7	0.304
1	21	0.913
2	17	0.739
3	5	0.217
4	15	0.652

Z_1 for example, was calculated by $Z_1 = (aZ_0 + c) \bmod m = (3 \times 7) \bmod 23 = 21$.
This illustrates the 'recipe' mentioned earlier in this section.

Random number generators have important characteristics, some are

1. They should be $U(0,1)$ distributed.
2. They can take only discrete values due to integer division, which yields the numbers (*when sorted*) $\frac{1}{m}, \frac{2}{m}, \ldots, \frac{m-1}{m}$. They are thus not perfectly continuous.
3. A random number stream is periodic, *i.e.* a set of numbers is generated before the stream starts to repeat itself. The length of this period is at most m, since $0 \le Z_i \le m - 1$.
4. It can degenerate. This means that the same random number is generated over and over. The mid-square method is a good example of a random number generation method that can cause quick degeneration.
5. A seed number is required to initiate the RNG.
6. The sequence of numbers generated should be reproducible for debugging purposes and also to evaluate different alternative systems on common grounds.

The choice for the various parameters has been researched extensively, as well as the development of new RNG methods, see for example the 'combined multiple recursive generator' (CMRG) in Kelton, R. Sadowski, and D. Sadowski (2002), pp. 475–476.

Here is a cool random number generator at `https://www.youtube.com/watch?v=1cUUfMeOijg`.

The site `https://www.random.org/` claims to generate true random numbers.

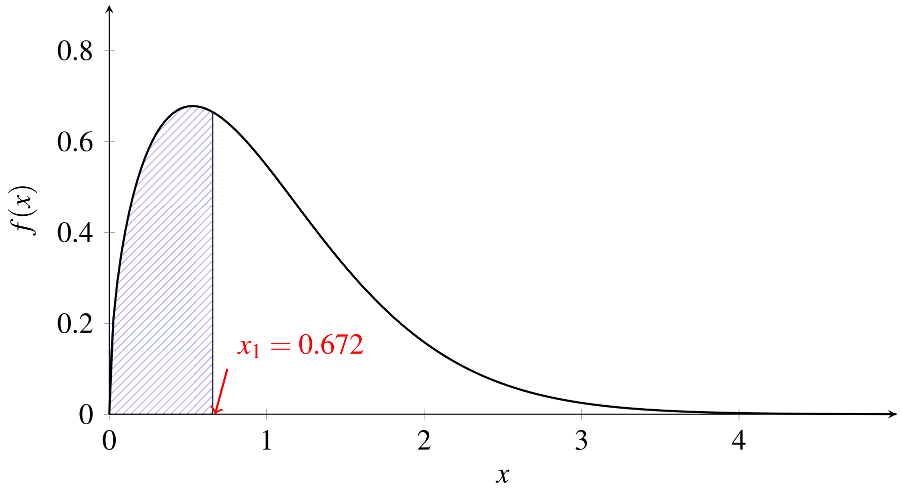


Figure 3.2: The area under the Weibull function $f(x,\alpha,\beta) = f(x,1.5,0.9)$ is associated with the value of $U = 0.37$. The associated x-value is $x_1 = 0.672$.

P Point to ponder: Our world seems to be driven by randomness. You cannot tell when the next client will arrive at your fruit stall, or what the client will buy. Is it really randomness or do we only perceive it?

3.3 Generating random variates

Referring to the discussion above, one can ask why we need random numbers? This question can be answered by asking: How does a simulation software program generate random observations from the various statistical distributions? There are several methods, but only the inverse transform will be discussed.

3.3.1 The inverse transform: continuous case

The *inverse transform* method is based on the use of the cumulative distribution function F of a variate, and the assumption is that 1) the inverse of F, namely F^{-1}, exists and 2) can be determined.

The inverse transform method for continuous distributions works as follows: We know how to generate a random number U, and $0 < U < 1$. But $0 \le F(x) \le 1$. So why not generate a U value and set it equal to $F(x)$? But what then? Let us first find $F(x)$ with

$$F(x) = \int_{-\infty}^{x} f(t)\mathrm{d}t \tag{3.2}$$
$$= U. \tag{3.3}$$

The value of $U = 0.37$ is shown as an area in Figure 3.2. The corresponding value is $X_1 = 0.672$.

In general, we generate a random number U from $U(0,1)$, and return $X = F^{-1}(U)$. This can be proved as follows:

Let F_X denote the distribution function of $X = F^{-1}(U)$. Then

$$\begin{aligned}
F_X(x) &= P\{X \leq x\} \\
&= P\{F^{-1}(U) \leq x\} \\
&= P\{F(F^{-1}(U)) \leq F(x)\} \text{ (because } F(x) \text{ is a monotone} \\
&= P\{U \leq F(x)\} \qquad\qquad \text{increasing function of } x) \\
&= F(x).
\end{aligned} \tag{3.4}$$

We thus determine the cumulative distribution function ($F(x)$) of a distribution from which we want to sample, then we 'randomly' select a value between 0 and 1 and insert it into the obtained inverse cumulative distribution function ($F(x) = U$) which returns an observed value $X = F^{-1}(U)$. By repeating this 'many times', the desired distribution is generated.

The mechanism is graphically illustrated in Figure 3.3 for an exponential distribution with $\beta = 4$. A random number on the vertical axis is selected, and at that height, a horizontal line is projected to the curve of $F(x)$, then a vertical line is followed downwards to the x-axis. The value at the intersection is the observation generated from the distribution. Note that this example uses $X = -\beta \ln(1 - \mathbf{U})$.

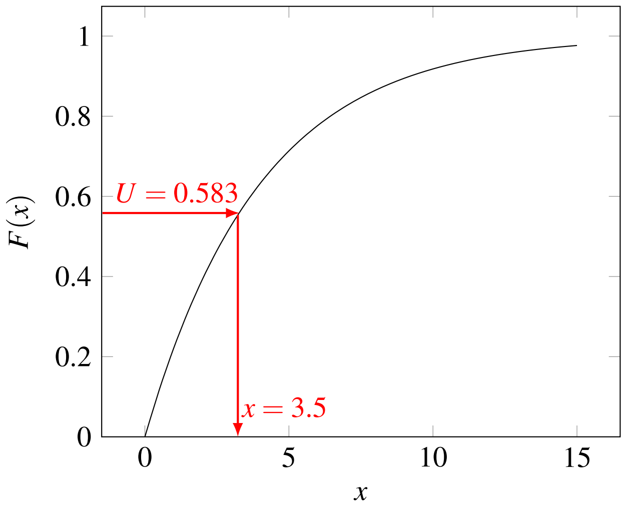


Figure 3.3: Random variate generation through an inverse transform (continuous)

■ **Example 3.2** Suppose a random variate X is exponentially distributed with probability density function

$$f_X(x) = \frac{1}{\beta} \exp^{-x/\beta}, \quad x > 0, \quad \beta > 0. \tag{3.5}$$

The function can be inverted to obtain an expression for X. We use the result from (3.4): first determine the integral of $f(x)$, then set it equal to U, and solve for x. We denote x as X because it is a variate.

$$\begin{aligned}
F(x) &= \int_0^x f(z)\mathrm{d}z \\
&= \int_0^x \frac{1}{\beta}\exp\left(\frac{-z}{\beta}\right)\mathrm{d}z \\
&= -\exp\frac{-z}{\beta}\Big|_0^x \\
&= 1-\exp\frac{-x}{\beta} \\
&= U. \\
X &= F^{-1}(U) \\
&= -\beta\ln(1-U).
\end{aligned} \tag{3.6}$$

P Points to ponder:

- In (3.5) it is shown that $x>0$, but the result in (3.6) has a negative in the equation. Can you explain?
- In (3.6), analysts often use $-\beta\ln(U)$. Is that valid?

■ **Example 3.3** Determine the inverse transform of the Weibull distribution $f(x)=\alpha\beta^{-\alpha}x^{\alpha-1}\mathrm{e}^{-(x/\beta)^\alpha}$.

$$\begin{aligned}
F(x) &= \int_0^x f(t)\mathrm{d}t \\
&= \int_0^x \alpha\beta^{-\alpha}t^{\alpha-1}\mathrm{e}^{-(t/\beta)^\alpha}\mathrm{d}t \\
\text{Let } u &= \left(\frac{t}{\beta}\right)^\alpha \\
\frac{\mathrm{d}u}{\mathrm{d}t} &= \alpha\beta^{-\alpha}t^{\alpha-1} \\
\text{If } t &= 0 \text{ then } u=0, \\
\text{if } t &= x \text{ then } u=\left(\frac{x}{\beta}\right)^\alpha. \\
F(x) &= \int_0^{(x/\beta)^\alpha} \mathrm{e}^{-u}\mathrm{d}u \\
&= 1-\mathrm{e}^{-(x/\beta)^\alpha} \\
&= U. \\
X &= \beta(-\ln(1-U))^{1/\alpha}.
\end{aligned}$$

■ **Example 3.4** Determine the inverse transform of the continuous uniform distribution on $[a,b]$.

$$\begin{aligned}
f(x) &= \frac{1}{b-a} \\
F(x) &= \int_a^x \frac{1}{b-a}\mathrm{d}t \\
&= \frac{x-a}{b-a} \\
&= U. \\
X &= a+(b-a)U.
\end{aligned}$$

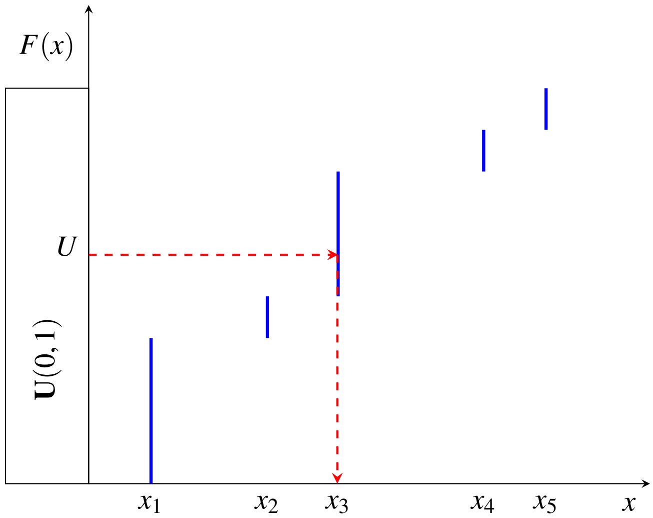


Figure 3.4: Random variate generation through an inverse transform (discrete)

P Point to ponder: Why should the result of the continuous uniform distribution inverse transform be intuitive?

3.3.2 The inverse transform: discrete case

Assume we have a probability mass function

$$f(x) = \begin{cases} p_0 & \text{if} \quad x = x_0 \\ p_1 & \text{if} \quad x = x_1 \\ \vdots & \vdots \\ p_j & \text{if} \quad x = x_j, \end{cases}$$

and the discrete inverse transform must be determined. The cumulative distribution function for the probability mass function $f(x) = P(X = x)$ is $F(x) = \sum_{t \le x} f(t)$. Take the Unif(1,6) distribution for example: $F(3) = f(1) + f(2) + f(3) = 1/6 + 1/6 + 1/6 = 0.5$. Now generate $U \sim U(0,1)$, and determine the smallest positive integer i so that $U \le F(x_i)$ and return $X = x_i$. Formally, we use

$$X = \begin{cases} x_0 & \text{if } U < p_0 \\ x_1 & \text{if } p_0 \le U < p_0 + p_1 \\ \vdots & \\ x_j & \text{if } \sum_{i=0}^{j-1} p_i \le U < \sum_{i=0}^{j} p_i \end{cases}$$

as an algorithm. The inverse transform approach for discrete random variates is graphically illustrated in Figure 3.4.

The variates are thus generated in discrete steps, while the probability of creating a variate is proportional to the probability mass function.

■ **Example 3.5** Using $U_1 = 0.33$, $U_2 = 0.65$ and $U_3 = 0.45$, generate three observations from

$$f(x) = \begin{cases} 0.35, & x = 1 \\ 0.20, & x = 2 \\ 0.45, & x = 3. \end{cases}$$

Figure 3.4 has been adapted to the given $f(x)$, as shown in Figure 3.5.

1. $U_1 = 0.33 < 0.35$, return $X = 1$.
2. $0.35 + 0.20 = 0.55 < U_2 = 0.65 < 0.35 + 0.20 + 0.45 = 1.00$, return $X = 3$.
3. $0.35 < U_3 = 0.45 < 0.35 + 0.20 = 0.55$, return $X = 2$.

The observations 1, 2, and 3 are returned in any order, as dictated by the random numbers U_i. Once 'many' observations are generated, the proportions of 1, 2 and 3 should approach 0.35, 0.20 and 0.45 respectively.

P Points to ponder: How would you make the process applied in the example efficient? Here 'efficient' means the computer makes the least number of comparisons when executing the process.

■ **Example 3.6** The discrete, uniform distribution $U(a,b)$, $a < b$ and both integers, is defined by

$$f(x) = \begin{cases} \dfrac{1}{b-a+1} & \text{if } x \in \{a, a+1, \ldots, b\} \\ 0 & \text{otherwise.} \end{cases}$$

To sample from this distribution, we use $X = a + \text{INT}((b-a+1)U)$, where 'INT' returns the integral part of a number. It is also denoted by $\lfloor x \rfloor$. Examples: $\text{INT}(0.9) = 0$, $\text{INT}(6.2) = 6$.

P Points to ponder: How would you generate observations from a Bernoulli distribution with p.m.f. $f(x) = p^k(1-p)^{1-k}$ for $k \in \{0,1\}$, with p the probability of success? And the Poisson distribution with p.m.f. $f(x) = e^{-\lambda}\lambda^x/x!, \quad x = 0, 1, 2, \ldots$?

Bringing it together

Here is an example to show how the generation of random numbers and random variates are done.

A random number generator and a distribution are given below. Let $z_0 = 54$ and $k = 2$, and create three random variables from the distribution, starting with z_1.

$$\begin{aligned} z_{i+1} &= (39z_i + 71) \quad \text{mod } 513 \\ f(x) &= kx^{-k-1}, \quad x \geq 1. \end{aligned}$$

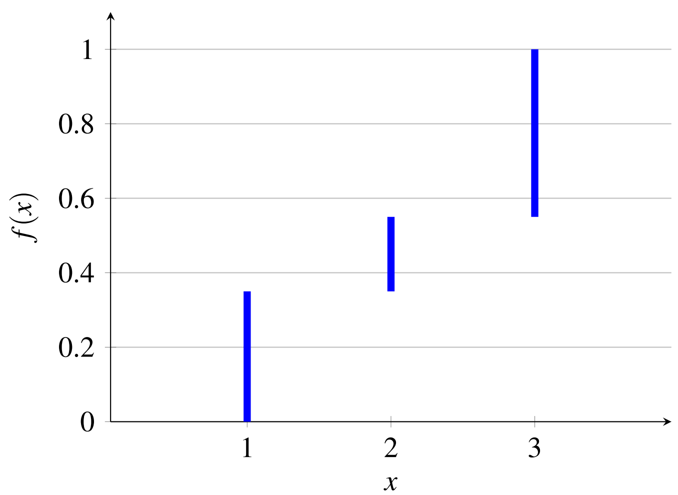


Figure 3.5: Random variate generation through an inverse transform (discrete)

Answer:
Use the inverse transform:

$$\begin{aligned} F(x) &= \int_1^x f(y)dy \\ &= \int_1^x 2y^{-3}dy \\ &= 1-x^{-2}, \end{aligned}$$

and the inverse then is $X=(1-U)^{-1/2}$. Starting with $z_0=54$ we construct the following table, with the required three observations shown in the last column.

$z_1 =$	125	$U_1 = Z_1/513 =$	0.243	$X_1 = 1.150$
$z_2 =$	329	$U_2 = Z_2/513 =$	0.641	$X_2 = 1.670$
$z_3 =$	77	$U_3 = Z_3/513 =$	0.150	$X_3 = 1.085$

3.3.3 Monte-Carlo simulation

Now that we know how random numbers are generated, we can study a specific application in the static, stochastic simulation domain. This application is called *Monte-Carlo (M-C) simulation*. It originated during World War II and was developed by nuclear scientists to solve difficult mathematical problems, including double and triple integrals. M-C simulation can be illustrated as follows: Suppose we want to determine

$$I = \int_a^b f(x)\mathrm{d}x, \quad -\infty < a < b < \infty, \tag{3.7}$$

and I does not have an exact solution. We could possibly determine I with numerical methods, but for illustration, M-C simulation is used. Let $X \sim U(a,b)$, then $g(x)=1/(b-a)$. Define

$$Y=(b-a)f(X),$$

then

$$\begin{aligned} E[Y] &= E[(b-a)f(X)] \\ &= \int_a^b (b-a)g(x)f(x)\mathrm{d}x \\ &= \int_a^b f(x)\mathrm{d}x \\ &= I. \end{aligned}$$

If we can estimate the $E[Y]$, we have an approximation for I. To do so, we first transform the limits $[a,b]$ of the integral to the region (0,1) and obtain a function $f(y)$, then draw pseudorandom numbers U_i in the place of y and determine $f(U_i)$. The variable x has been transformed into the variable y, which is defined on (0,1). We do this n times (simulating), and estimate an approximation for I, as follows:

$$\bar{Y} = \frac{(b-a)\sum_{i=1}^{n} f(U_i)}{n}. \tag{3.8}$$

Suppose in (3.7), $a=0$ and the upper limit is infinity, then we transform the range to $[0,1]$ by letting $y=1/(x+1)$, then $\mathrm{d}y=-\mathrm{d}x/(1+x)^2$ and $I=\int_0^1 h(y)\mathrm{d}y$. Different transformations are needed for the integral boundaries. The cases are shown in Table 3.1, with $f(x)=e^{x^2}$ as example.

Table 3.1: Monte-Carlo simulation transformations

		Bound-aries	Transformation	Transformed integral
Case 1:	$I=\int_a^b f(x)\mathrm{d}x=\int_a^b \mathrm{e}^{x^2}\mathrm{d}x$	$[a{,}b]$	$y=\dfrac{x-a}{b-a}$ $x=a+(b-a)y$ $x=a$: $y=0$ $x=b$: $y=1$ $\dfrac{\mathrm{d}y}{\mathrm{d}x}=\dfrac{1}{b-a}$ $\mathrm{d}x=(b-a)\mathrm{d}y$	$I=\int_0^1 (b-a)f(y)\mathrm{d}y$ $I=\int_0^1 (b-a)\mathrm{e}^{(a+(b-a)y)^2}\mathrm{d}y$
Case 2:	$I=\int_0^\infty f(x)\mathrm{d}x=\int_0^\infty \mathrm{e}^{-x^2}\mathrm{d}x$	$[0{,}\infty]$	$y=\dfrac{1}{x+1}$ $x=\dfrac{1}{y}-1$ $x=0\ \ : y=1$ $x\to\infty$: $y=0$ $\dfrac{\mathrm{d}y}{\mathrm{d}x}=\dfrac{-1}{(x+1)^2}$ $=-y^2$ $\mathrm{d}x=-\dfrac{\mathrm{d}y}{y^2}$	$I=-\int_1^0 \dfrac{f(y)}{y^2}\mathrm{d}y$ $=\int_0^1 \dfrac{f(y)}{y^2}\mathrm{d}y$ $I=\int_0^1 \dfrac{\mathrm{e}^{-((1/y)-1)^2}}{y^2}\mathrm{d}y$
Case 3:	$I=\int_{-\infty}^{\infty} f(x)\mathrm{d}x$ $=\int_{-\infty}^{\infty}\mathrm{e}^{x^2}\mathrm{d}x$ $=\int_{-\infty}^{0}\mathrm{e}^{x^2}\mathrm{d}x+\int_0^\infty \mathrm{e}^{x^2}\mathrm{d}x$ $=I_1+I_2$ (We address I_1 here).	$[-\infty{,}0]$	$y=\dfrac{1}{1-x}$ $x=1-\dfrac{1}{y}$ $x\to-\infty$: $y=0$ $x=0\ \ \ : y=1$ $\dfrac{\mathrm{d}y}{\mathrm{d}x}=\dfrac{1}{(1-x)^2}$ $=y^2$ $\mathrm{d}x=\dfrac{\mathrm{d}y}{y^2}$	$I_1=\int_0^1 \dfrac{f(y)}{y^2}\mathrm{d}y$ $I_1=\int_0^1 \dfrac{\mathrm{e}^{(1-1/y)^2}}{y^2}\mathrm{d}y$ I_2 is according to **Case 2**.

To do the actual simulation, one would generate 'many' values for $y\sim U(0,1)$ and calculate *many values* of I in column 4, then average them. This will give an *estimation* of the integral.

Example of implementation: Take the transformed integral of Case 2 and estimate its value by generating 10 000 observations, then take the average. The exact answer is $\sqrt{\pi}/2 = 0.886$.

Obs. no.	$y = U$	$\frac{e^{((1/y)-1)^2}}{y^2}$
1	0.981	1.039692
2	0.753	1.583729
3	0.915	1.184991
4	0.674	1.742165
...	...	...
10 000	0.412	0.772859
	Ave. =	0.88945

Survival kit:

Conquer Mount Randomness easily by carrying the following in your assessment pack:

1. Understand the need for and the principle of generating random numbers.
2. Know the properties of random number generators.
3. Be able to generate random numbers using a given formulation.
4. Do the inverse transform for both the discrete and continuous cases.
5. Do Monte-Carlo simulation, given any of the three cases.