Let’s see what is involved in using the above *transformation method* to generate some arbitrary desired distribution of $y$’s, say one with $p(y) = f(y)$ for some positive function $f$ whose integral is 1. (See Figure 7.2.1.) According to (7.2.4), we need to solve the differential equation

$$\frac{dx}{dy} = f(y)$$

(7.2.6)

But the solution of this is just $x = F(y)$, where $F(y)$ is the indefinite integral of $f(y)$. The desired transformation which takes a uniform deviate into one distributed as $f(y)$ is therefore

$$y(x) = F^{-1}(x)$$

(7.2.7)

where $F^{-1}$ is the inverse function to $F$. Whether (7.2.7) is feasible to implement depends on whether the inverse function of the integral of $f(y)$ is itself feasible to compute, either analytically or numerically. Sometimes it is, and sometimes it isn’t.

Incidentally, (7.2.7) has an immediate geometric interpretation: Since $F(y)$ is the area under the probability curve to the left of $y$, (7.2.7) is just the prescription: choose a uniform random $x$, then find the value $y$ that has that fraction $x$ of probability area to its left, and return the value $y$.

**Normal (Gaussian) Deviates**

Transformation methods generalize to more than one dimension. If $x_1, x_2, \ldots$ are random deviates with a joint probability distribution $p(x_1, x_2, \ldots)$ $dx_1 dx_2 \ldots$, and if $y_1, y_2, \ldots$ are each functions of all the $x$’s (same number of $y$’s as $x$’s), then the joint probability distribution of the $y$’s is

$$p(y_1, y_2, \ldots)dy_1 dy_2 \ldots = p(x_1, x_2, \ldots) \left| \frac{\partial(x_1, x_2, \ldots)}{\partial(y_1, y_2, \ldots)} \right| dy_1 dy_2 \ldots$$

(7.2.8)

where $\left| \frac{\partial}{\partial} \right|$ is the Jacobian determinant of the $x$’s with respect to the $y$’s (or reciprocal of the Jacobian determinant of the $y$’s with respect to the $x$’s).
An important example of the use of (7.2.8) is the Box-Muller method for generating random deviates with a normal (Gaussian) distribution,

\[ p(y)dy = \frac{1}{\sqrt{2\pi}} e^{-y^2/2} dy \]  

(7.2.9)

Consider the transformation between two uniform deviates on (0,1), \( x_1, x_2 \), and two quantities \( y_1, y_2 \),

\[ y_1 = \sqrt{-2 \ln x_1 \cos 2\pi x_2} \]
\[ y_2 = \sqrt{-2 \ln x_1 \sin 2\pi x_2} \]  

(7.2.10)

Equivalently we can write

\[ x_1 = \exp \left[ -\frac{1}{2} (y_1^2 + y_2^2) \right] \]
\[ x_2 = \frac{1}{2\pi} \arctan \frac{y_2}{y_1} \]  

(7.2.11)

Now the Jacobian determinant can readily be calculated (try it!):

\[ \frac{\partial(x_1, x_2)}{\partial(y_1, y_2)} = 
\begin{vmatrix}
\frac{\partial x_1}{\partial y_1} & \frac{\partial x_1}{\partial y_2} \\
\frac{\partial x_2}{\partial y_1} & \frac{\partial x_2}{\partial y_2}
\end{vmatrix} = 
\left[ \frac{1}{\sqrt{2\pi}} e^{-y_1^2/2} \right] \left[ \frac{1}{\sqrt{2\pi}} e^{-y_2^2/2} \right] \]  

(7.2.12)

Since this is the product of a function of \( y_2 \) alone and a function of \( y_1 \) alone, we see that each \( y \) is independently distributed according to the normal distribution (7.2.9).

One further trick is useful in applying (7.2.10). Suppose that, instead of picking uniform deviates \( x_1 \) and \( x_2 \) in the unit square, we instead pick \( v_1 \) and \( v_2 \) as the ordinate and abscissa of a random point inside the unit circle around the origin. Then the sum of their squares, \( R^2 \equiv v_1^2 + v_2^2 \) is a uniform deviate, which can be used for \( x_1 \), while the angle that \((v_1, v_2)\) defines with respect to the \( v_1 \) axis can serve as the random angle \( 2\pi x_2 \). What’s the advantage? It’s that the cosine and sine in (7.2.10) can now be written as \( v_1 / \sqrt{R^2} \) and \( v_2 / \sqrt{R^2} \), obviating the trigonometric function calls!

We thus have

```c
#include <math.h>

float gasdev(long *idum)
Returns a normally distributed deviate with zero mean and unit variance, using ran1(idum) as the source of uniform deviates. 
{
  float ran1(long *idum);
  static int iset=0;
  static float gset;
  float fac,rsq,v1,v2;
  if (*idum < 0) iset=0;  
  if (iset == 0) {
    we don't have an extra deviate handy, so  
    do {  
      v1=2.0*ran1(idum)-1.0;  
      v2=2.0*ran1(idum)-1.0;  
      rsq=v1*v1+v2*v2;  
    } while (rsq >= 1.0);  
  
  fac=1.0/sqrt(-2.0*log(v1));  
  v2=atan2(v2,v1);  
  return fac*v1;  
}
```
} while (rsq >= 1.0 || rsq == 0.0);  
and if they are not, try again.
fac=sqrt(-2.0*log(rsq)/rsq);
Now make the Box-Muller transformation to get two normal deviates. Return one and
save the other for next time.
gset=v1*fac;
iset=1;  
Set flag.
return v2*fac;
} else {
We have an extra deviate handy,
iset=0;  
so unset the flag,
return gset;
}
}


CITED REFERENCES AND FURTHER READING:
[1]
(Reading, MA: Addison-Wesley), pp. 116ff.

7.3 Rejection Method: Gamma, Poisson,
Binomial Deviates

The rejection method is a powerful, general technique for generating random
deviates whose distribution function $p(x) dx$ (probability of a value occurring between
$x$ and $x + dx$) is known and computable. The rejection method does not require
that the cumulative distribution function [indefinite integral of $p(x)$] be readily
computable, much less the inverse of that function — which was required for the
transformation method in the previous section.

The rejection method is based on a simple geometrical argument:
Draw a graph of the probability distribution $p(x)$ that you wish to generate, so
that the area under the curve in any range of $x$ corresponds to the desired probability
of generating an $x$ in that range. If we had some way of choosing a random point in
two dimensions, with uniform probability in the area under your curve, then the $x$
value of that random point would have the desired distribution.

Now, on the same graph, draw any other curve $f(x)$ which has finite (not
infinite) area and lies everywhere above your original probability distribution. (This
is always possible, because your original curve encloses only unit area, by definition
of probability.) We will call this $f(x)$ the comparison function. Imagine now
that you have some way of choosing a random point in two dimensions that is
uniform in the area under the comparison function. Whenever that point lies outside
the area under the original probability distribution, we will reject it and choose
another random point. Whenever it lies inside the area under the original probability
distribution, we will accept it. It should be obvious that the accepted points are
uniform in the accepted area, so that their $x$ values have the desired distribution. It