On Random-Number Distributions for C++0x

Document number: WG21/N1588 = J16/04-0028
Date: February 13, 2004
Revises: None
Project: Programming Language C++
Reference: ISO/IEC IS 14882:2003(E)
Reply to: Marc Paterno <paterno@fnal.gov>
Mail Station 114
CEPA Dept., Computing Division
Fermi National Accelerator Laboratory
Batavia, IL 60510-0500

Contents

---

1 Purpose

Document N1452, *A Proposal to Add an Extensible Random Number Facility to the Standard Library (Revision 2)*, is of great interest to my community. Since the Oxford meeting, I have many times been asked about the features of the facility proposed. The proposal's general reception has been extremely positive.

In particular, the proposal includes a list of *random engines* and a list of *random-number distributions* to be made available by the C++ Standard Library. It is these lists that are of special interest to the scientific community.

The selection of random engines has met with widespread approval. The criteria for the inclusion of a random engine are given in section III.H of the proposal. These criteria are clear, sufficiently objective, and have been applied consistently to produce the proposed list of random engines.

Unfortunately, the rationale that is provided to guide the selection of distributions seems not nearly as strong. The sole stated criterion is given in section III.I of the proposal: "The following distributions were chosen due to their relatively widespread use."

This document explores this and other possible criteria for deciding upon, and defending, a list of suitable distributions. While it would have been best to have had a concrete proposal for the modifications to TR1, I think it more realistic to consider modifications, if any, to the forthcoming working document for C++0x.

2 Summary of the Distributions in N1452

Section III.I ("Which Distributions to Include") of N1452 presents a list of distributions proposed for inclusion in the Standard. The distributions are listed below:

integer uniform	normal	geometric
floating-point uniform	binomial	Poisson
exponential	gamma	Bernoulli

The same section of the proposal lists many additional distributions (and the sources that inspired their consideration) which were considered and rejected. But lacking from the section is any organizing principle according to which the acceptance or rejection can be defended.

3 Possible Criteria for Choosing the Canonical Distributions

What criteria might we apply in choosing the canonical distributions? The following come to mind:

1. **Appeal to authority.** This is the approach taken in document N1542, *A Proposal to add Mathematical Special Functions to the C++ Standard Library (version 3)*. There, an ISO standard was referenced to determine and rationalize the list of mathematical special functions to be included in the library.

   Unfortunately, in the domain of random-number distributions, there seems to be no agreed-upon authority to which we can appeal. I have found, for example, no ISO document specifying a set of random-number distributions, nor have I found a reference work of sufficient preeminence in the field to stand as an unimpeachable authority.

2. **Choose the minimal set.** The minimal set of distributions would contain only *integer uniform*, or *floating-point uniform*, or perhaps both. In theory at least, all other distributions can be generated from this set.

   This choice leaves far too much work to users — almost all users of the random-number facility would need to implement several distributions. The likely result is that almost all users would be displeased.

3. Survey the "prior art" in available resources, and **provide the ubiquitous**; that is, provide those distributions common to nearly *all* the resources surveyed.

   This approach seems to approximate the criterion used to determine the list of distributions in N1452. It has the advantage of producing a short list, since only the distributions of the very broadest interest will be included. But it seems to have the *disadvantage* that it will satisfy almost no user community, each of which finds additional distributions essential.

4. Survey the "prior art" in available resources, and **provide the (nearly) complete set**. This would presumably satisfy (nearly) all users.

   My original goal was to produce such a list, and to demonstrate that it was of reasonable size. Unfortunately, what I found was that it was very difficult to identify, especially for domains in which I lacked expertise, what distributions were truly important. However, in the domains for which I *do* have expertise, I found the distributions of interest mostly fell into a few different "families." This observation led to a final choice of criteria.

5. Consider the "prior art" in available resources, **identify the "families" formed by the commonly used distributions, and provide the natural set of distributions for each**

family. Distributions which do not naturally fall into any of the families seem to be the ones that are of interest only to specialized user communities. Such distributions do not seem viable candidates for inclusion in the canonical list.

Applying the criteria in [5] has produced a list which I believe will satisfy many users, yet is of reasonable length.

4 A Brief Survey of Popular Distributions

My goal in this survey was to list those distributions most commonly used by a representative handful of user communities, to see if we can generate a list that is:

- free of "gaping holes" that will attract the ire of large numbers of users, and yet
- short enough to be feasible.

I have *not* attempted a comprehensive survey of all interested user communities.

The observation that the list of commonly used distributions contains natural families has helped to identify the important missing distributions as well as to trim the candidate list of those distributions less widely used.

All the families, save the last, are interesting to users working in a wide variety of domains because

- each family is identified with a specific *random process*, and
- these kinds of random processes can appear in any domain.

Thus random number generators that produce these distributions are widely needed, and are of general interest.

The final family is of interest for a different reason: it contains the most flexible of random number generators, allowing the interested user to generate random numbers from almost any distribution wanted. Often, especially in the field of simulations, this is a critical ability. The random number generators of this family are very widely used, and are of general interest.

The distributions which appear in the specialized literature of individual domains, in contrast, seem either to be associated with a process that does *not* appear in a wide variety of domains, or to lack a fundamental underlying process entirely. Distributions of this second kind seem often to have been created as flexible functions, capable of matching a wide variety of shapes, and thus capable of being fit to a wide variety of observed data. Random number generators that produce these distributions are less widely needed, and are of interest to more specialized communities.

For the interested reader, I have provided an appendix with brief descriptions and examples for each of the distributions in the families discussed below. These details are not necessary to the main argument presented in this paper.

4.1 Uniform Family

This family contains the very simplest and most basic distributions. They describe uniform random processes — random processes which can result in the occurrence of any element of a set, each of which is equally likely as the outcome. While sometimes useful on their own, the distributions in this family find their primary use as building blocks for the generation of other distributions.

The distributions making up the uniform family are:

- integer uniform
- floating-point uniform

Document N0352, *Proposal for Standardization of Random Number Generators in C++* also identified this family of distributions, recommending them for inclusion in the Standard in 1993.

4.2 The Bernoulli Family

The next family of distributions is related to *Bernoulli processes*. A Bernoulli process is a random process with exactly two possible outcomes, and with a fixed probability of obtaining each of those outcomes. (These probabilities need not be equal.) It is common to label one of the outcomes "success" and the other "failure," and to term each event a "trial."

Bernoulli processes appear in many problem domains:

1. The flipping of a coin, whether "fair" or not, is a Bernoulli process.
2. If we have an unreliable mechanical component that has a chance of failing upon use, and that chance of failing *does not change* with time or use, then use of that component is a Bernoulli process.
3. The decay of a π^0 particle, into either (a) two photons or (b) "something else," is a Bernoulli process. Note that the second alternative is merely "something other than the first alternative." It does *not* have to be more precisely specified. It could be, and in this case is, that "something else" is a label applied to a group of other possible results, among which we do not distinguish.

The members constituting the Bernoulli family are:

- Bernoulli
- binomial
- geometric
- negative binomial

4.3 The Poisson Family

The next family of distributions is related to *Poisson processes*. A Poisson process is a random process such that:

- events happen one at a time, not in batches (*i.e.*, the probability of two events occurring *exactly* simultaneously is zero),
- the probability of an event occurring in a given interval is directly proportional to the length of the interval (the constant of proportionality determines the *rate* of the process), and
- the rate of the process does not change.

Poisson processes appear in many application domains:

1. At a uniformly busy call center (*i.e.*, one at which the probability of receiving a call in any one second is constant over the day), the arrival of calls is a Poisson process.
2. For a radioactive sample containing a large number of radioactive nuclei, the decay of nuclei is a Poisson process. (A "large" number of nuclei are needed so that the rate of decay does not change over the period of time during which observations are made.)

The members of the Poisson family are:

- Poisson
- exponential
- gamma
- Weibull
- extreme value

4.4 The Normal Family

The distributions in the normal family are all related to the *normal distribution*, which gains its importance from the *central limit theorem*. The central limit theorem, roughly stated, says that the distribution of the *sum* of a set of independent random variables tends toward the normal distribution as the number of variables being summed grows large.[1]

The normal distribution, also called the *Gaussian* distribution, or "the bell curve," is widely used to model measurement errors, using the argument that the errors in question result from the accumulation of a large number of small, independent effects.

The other members of this family are all related to the normal distribution in that they are the distributions obtained by performing elementary mathematical functions on one or more normal random variables. Thus these distributions appear often in the modeling of quantities that are calculated from measurements containing errors. The ubiquity of the normal distribution makes these other distributions widely used as well.

The members of the normal family are:

- normal
- lognormal
- chi-squared
- Cauchy
- Fisher's F
- Student's t

4.5 Sampling Family

The final family is differs from the others in one respect: its members are not based upon an underlying, fundamental random process. Rather, the two entries in this family are methods of

[1] Strictly speaking, the distribution function for each of the variables being summed must have finite mean and variance. The variables need not have the same distribution functions.

specifying a distribution of (almost) arbitrary shape, in a fashion that allows for efficient generation of random numbers according to that distribution.

The members of the sampling family are:

- histogram sampling
- cumulative distribution function sampling

5 Distributions Unsuitable for Inclusion

The distributions in this section are fundamentally different from those in the preceding section in that they produce not single values, but *collections* of values. The natural return type of a random distribution from one of these families would have to be capable of representing a collection of values. The proposed random number facility is designed to deal with random distributions which return only *single instances* of numeric types.[2]

While the capability to generate samples from such distributions would surely be of some value, the random number facility as defined in the proposal is not be capable of supporting these distributions. I believe the following families of distributions are *unsuitable* for inclusion in the standard, because the necessary modification of the proposed random number facility would be significant.

Nonetheless, I describe the families, in case of such modification in the future.

5.1 The Combinatorics Family

This next family of distributions involves *sampling processes*. An example of such a process is the (random) selection of a set of balls from a bag containing balls of several different colors. These distributions often arise such fields as combinatorics and game theory, and in simulations.

The distributions of the combinatoric family are:

- multinomial
- hypergeometric

5.2 Other Multivariate Distributions

Many of the distributions listed in §4 can be naturally extended to multivariate versions. Several such distributions are widely used; most notable is the *multivariate normal* distribution. However, these distributions suffer from the same problem as do the "combinatoric family" of distributions: the proposed random number facility is not designed to handle the return of a collection of values.

5.3 Unclassified Distributions

In the course of researching the families of distributions, I came across a number of distributions that did not seem to fall into any of the given families. While each is of use in at least one

[2] The table listing "number generator requirements" specifies that `X::result_type` must be of type `T`, such that `std::numeric_limits<T>::is_specialized` is true.

application domain, they seem less widely-used. They therefore do not seem suitable candidates for inclusion in the list of canonical distributions for the Standard. I list them below, without further description or analysis.

beta	triangular	hypoexponential
logistic	log-logistic	hyperexponential
Pareto	Johnson S_U	Pearson type V
Landau	Johnson S_B	Pearson type VI
Gibbs	Boltzmann	Kolmogorov-Smirnov
		Cramer-Smirnov-von Mises

6 Summary and Conclusion: Where to Proceed?

I believe that the identification of these families of related distributions can lead to a more coherent set of "canonical distributions". I believe that such a set can be defended reasonably, that is is of reasonable size, and that it will satisfy a large part of the user community. Table 1 presents the full list of distributions identified in this document which seem suitable for inclusion in the canonical list of distributions. Approximately half of the distributions identified are already in the proposal; these are identified with the symbol ✓.

Uniform Family	Bernoulli Family	Poisson Family	Normal Family	Sampling Family
integer ✓	Bernoulli ✓	Poisson ✓	normal ✓	histogram
floating-point ✓	binomial ✓	exponential ✓	lognormal	CDF
	geometric ✓	gamma ✓	chi-squared	
	negative binomial	Weibull	Cauchy	
		extreme value	Fisher's F	
			Student's t	

Table 1: Random-number distributions suggested for inclusion in the random number facility. Distributions already in N1452 are marked with the symbol ✓.

It seems that significant work would be necessary to enhance the proposed random number facility to deal with multivariate distributions. Regrettably, that does not seem possible at this time.

7 Acknowledgments

I would like to thank Walter Brown, for considerable assistance in the development of this paper. Thanks also to Jim Kowalkowski, for the suggestion of several helpful clarifications.

I also thank the Fermi National Accelerator Laboratory's Computing Division, sponsors of my participation in the C++ standards effort, for their support.

A Appendix

A.1 Examples of the Uniform Family

integer uniform The *integer uniform* distribution describes the outcome of a process with a finite number of possible results, each of which is equally likely. Examples include:

- the number of spots appearing resulting from a single roll of a fair die, and
- the drawing of a specific card from a deck of cards.

floating-point uniform The *floating-point uniform* distribution is the extension of the integer uniform distribution to the limiting case of an infinite number of possible outcomes. A floating-point uniform process results in one value from a continuous range of finite extent. The probability of the value falling within any sub-range must depend on only the size of that sub-range.

This distribution is often called the "flat" distribution, because a graph of the distribution function is a flat line.

A generator of floating-point random numbers clearly only approximates a continuous distribution, since a computer uses a finite number of bits to represent each number. Nevertheless, in most situations where a continuous uniform distribution is wanted, a floating-point uniform distribution is more convenient to use than an integer uniform distribution.

A.2 Examples of the Bernoulli Family

See §4.2 for the definition of a Bernoulli process.

Bernoulli The *Bernoulli* distribution directly represents a Bernoulli process. It generates a result of *true* (success) with a specified probability, and *false* (failure) the rest of the time. The distribution is *memoryless* in that each trial has the same chance of success, regardless of the history of previous results.

binomial The *binomial* distribution describes the number of successes which occur in a given number of trials of a Bernoulli process. Examples include:

- the number of times that "heads" occurs in a fixed number of coin flips,
- the number of failures of an unreliable mechanical component (as described above) in a fixed number of uses, and
- the number of two-photon decays that occur in a sample of π^0 decays.

geometric The *geometric* distribution describes the number of failures of a Bernoulli trial that occur before the first success. Examples include:

- the number of times a coin is flipped before the first "tails" is found, and
- the number of times an unreliable mechanical component is used before it fails, and

- the number of two-photon π^0 decays observed before the first decay *other* than a two-photon decay.

negative binomial The *negative binomial* distribution describes the number of trials needed before obtaining the nth success in a series of independent Bernoulli trials. Examples include:

- the number of "tails" observed in a series of coin flips before the observation of the third "heads", and
- the number of two-photon π^0 decays before the 5th "other" decay is observed, and
- the number of successful uses of an unreliable component before the second failure occurs.

A.3 Examples of the Poisson Family

See §4.3 for the definition of a Poisson process.

Poisson The *Poisson* distribution describes the number of occurrences of a Poisson process observed in a given interval.

Examples include:

- the number of calls arriving at a uniformly busy call center in any 10 minute interval, and
- the number of nuclear decays observed in a large radioactive sample in one second.

exponential The *exponential* distribution describes the distribution of interval lengths (*i.e.*, "waiting times") between individual occurrences of a Poisson process. Examples include:

- the time between the arrival of calls at a uniformly busy call center, and
- the time between observations of nuclear decay in a large body of radioactive material.

gamma The *gamma* distribution describes the distribution of lengths of the interval required to observe a given number of occurrences of a Poisson process. Examples include:

- the time taken to receive 10 calls at a uniformly busy call center, and
- the time taken observe 1000 nuclear decays in a large body of radioactive material.

Weibull The *Weibull* distribution is related to the exponential distribution. If one generates a series of random numbers from the exponential distribution, and raises each number to some power α, the resulting values are distributed according to the Weibull distribution. More succinctly, if X is generated according to the exponential distribution, then $Y = X^\alpha$ is distributed according to the Weibull distribution.

Examples include:

- If the cost incurred due to operator idle time between calls at a uniformly busy call center is proportional to the square of the idle time, then the distribution of costs follows a Weibull distribution.

- If size of the signal from a detector of radioactive decays is proportional to the square root of the time between the observation of such decays, then the distribution of signals from a detector exposed to a large body of radioactive decays will follow the Weibull distribution.

Because of its flexible shape and ability to model a wide range of failure rates, it is often used as a purely *empirical* model, even in cases when there is little or no theoretical justification. The Weibull distribution is often recommended for the modeling of such things as the time taken to complete a task, or the time until the failure of a piece of equipment.

extreme value The *extreme value* distribution (also called the *Gumbel* distribution) is related to the Weibull distribution. If one generates a series of random numbers according to the Weibull distribution, and takes the logarithm of each number, the resulting values are distributed according to the extreme value distribution. More briefly, if a quantity X is generated pursuant to the Weibull distribution, then the quantity $Y = \ln(X)$ is distributed according to the extreme value distribution.

The extreme value distribution is often used to model the time to failure for a system which has many competing failure processes.

A.4 Examples of the Normal Family

normal The *normal* distribution is arguably the single most widely used distribution.

In most fields involving the statistical analysis of data, "random" measurement errors are frequently modeled as following the normal distribution, often justified by the claim that the error being modeled is the sum of a large number of independent effects, with an appeal to the central limit theorem.

In simulations, the normal distribution is often used to add "noise" to a signal, to model measurement errors in a real system.

In statistical physics, the normal distribution appears as the distribution with the largest entropy for a given mean and variance.

In some of the descriptions below, I use the term *standard* normal distribution. This means a normal distribution with mean of zero and variance of unity.

lognormal If one generates a series of random numbers generated according to the normal distribution, and takes the exponential of each of the generated numbers, the result is distributed according to the lognormal distribution. More briefly, if a quantity X is generated according to the normal distribution, then the the quantity $Y = \exp(X)$ is distributed according to the lognormal distribution.

Where the normal distribution is often used to model *additive* measurement errors, *multiplicative* factors in a measurement error are often modeled using the lognormal distribution.

Future values of stock prices (in the model most often used[3]) are distributed according to the lognormal distribution.

[3]The Black-Scholes model, assuming a Wiener process (Brownian motion) for the stock price.

chi-squared The *chi-squared* distribution (usually written as χ^2) is of central importance to the statistical analysis of data, and is used to describe the distribution of quality-of-fit parameters.

If one generates tuples of N numbers from N independent standard normal distributions, then the *sum* of the N elements of each tuple will be distributed according to the chi-squared distribution with parameter N. More succinctly, if X_i ($i = 1, \ldots N$) are independent standard normal random variates, then the sum $\chi^2 = \sum_{i=1}^{N} X_i^2$ is distributed according to the chi-squared distribution with parameter N.

It is precisely this sort of sum that is minimized by most orthodox data fitting methods.

Cauchy The *Cauchy* distribution (also called the *Lorentz* distribution, or the *Breit-Wigner* distribution) describes the ratio of two independent standard normal random variables. This distribution appears frequently in the statistical analysis of data, especially in so-called "robust" fitting methods.

In physics, the Breit-Wigner function appears as the distribution describing the energy distribution of quantum states with a finite lifetime.

Fisher's *F* *Fisher's* distribution (usually called F) is used in the analysis of variance (ANOVA), and is thus one of the most important distributions in the statistical analysis of data.

If one generates a sequence of pairs of random numbers, each generated from an independent chi-squared distribution, the ratio of the values in each pair is distributed according to the F distribution. More precisely, if X_1 is a chi-squared random variable with parameter n_1, and X_2 is a chi-squared random variable with parameter n_2, then

$$Y = \frac{X_1/n_1}{X_2/n_2}$$

is distributed according to the F distribution (with parameters n_1 and n_2).

Student's *t* *Student's* t distribution[4] appears in the analysis of the mean of a sample of data generated according to the normal distribution.

If one generates a sequence of pairs of numbers, the first element from a standard normal distribution and the second element from a chi-squared distribution, the value of the ratios of the pairs is generated according to the t distribution. More precisely, if X is a standard normal random variable, and Y is a chi-squared random variable, then $Z = X/Y$ is distributed according to the t distribution.

A.5 Examples of the Combinatorics Family

multinomial The *multinomial* distribution describes a process of sampling *with replacement*. It is a generalization of the binomial distribution to processes with more than two possible outcomes (that is, to non-Bernoulli events).

As an example, consider the selection of some number r of balls from a bag which contains a known number of balls of several colors. After the selection of each ball, we replace the selected ball, until we have drawn r times. The probability of choosing r_1 balls of the first color, r_2 balls of color the second color, *etc.*, up to r_n of the nth color is given by the multinomial distribution.

[4] "Student" was the pseudonym of William Sealy Gosset (1876-1937). He was a statistician who worked for the Guinness brewing company, under the stipulation that he not publish under his own name.

hypergeometric The *hypergeometric* distribution is similar to the multinomial distribution, except that it describes the similar probabilities in a process of sampling *without replacement*. Examples include the following.

- If we take (at random) 10 parts from a box of 100 parts, 5 of which are known to be defective, the probability of drawing 2 defective parts is given by the hypergeometric distribution.
- If we consider the decay of a collection of n π^0 mesons, the probability of observing n_1 two-photon decays, n_2 electron + positron + photon decays, and n_3 of all other kinds of decays is given by the multinomial distribution.

A.6 The Sampling Family

Histogram sampling In the *histogram* sampling method the user specifies a histogram which is the distribution function according to which random numbers are desired. The histogram sampling random number generator thus configured returns random numbers distributed according to the histogram.

Cumulative distribution function sampling In the *cumulative distribution function* (CDF) sampling method the user supplies the cumulative distribution function[5] — the integral of the distribution function (DF) — which describes the distribution required. The random number generator thus configured returns random numbers distributed according to the related distribution function.

As stated above, direct sample from an arbitrary distribution function (DF) is intentionally omitted It is omitted because is is generally less efficient than the use of cumulative distribution function (CDF) sampling method. I know of only two general methods of implementation for a DF-based random number generator.

One method is to integrate the DF numerically, obtaining the related CDF, and then use the CDF method. If a user is able to integrate his DF analytically, he is better served using the CDF method directly on the result. If not, he can always integrate it numerically himself, and then use the resulting CDF in the CDF method.

The other method is the *rejection method*. This is a Monte-Carlo based method that is relatively slow unless the function is fairly uniform.

It seems sufficient to support the histogram and CDF sampling methods, and that support of the PD sampling method is unnecessary. Of course, if evidence to the contrary arises, it would be sensible to include a DF sampling random number generated as well.

[5] Some sources call this the *probability function*.