GENERATING PSEUDORANDOM VARIATES FROM KNOWN DISTRIBUTIONS

Alvin G. Byars

Ralph H. Johnson VA Medical Center
Charleston, SC

Abstract

This paper introduces MUMPS techniques for generating pseudorandom variates from known probability distributions. It briefly discusses five methods for generating random variates and briefly describes twenty common distributions with typical examples. For your use, an attachment provides MUMPS extrinsic functions for generating pseudorandom variates from each of the twenty described distributions. Another attachment presents plots of generated random variates from typical examples of each of the twenty distributions.

Introduction

The ability to generate pseudorandom variates from known distributions can be useful for a variety of purposes. The author has encountered at least three situations where this usefulness has been demonstrated.

When conducting statistical analyses on sets of real data, for example, clinical patient data, the problem of missing data points must often be overcome. One useful way of dealing with the problem of missing data is to simply generate pseudorandom values for the missing data points.

Developing MUMPS routines for conducting statistical analyses requires sets of data for prototyping and validating the routines. Collecting real data for this purpose can be a tedious process. It is often simpler and more convenient to generate pseudorandom data sets.

Computer modeling and simulation can be a useful tool for studying a variety of processes, for example, simulating patient processing and waiting times in a clinic, or modeling unscheduled down time of a computer system. The ability to simulate data is essential for these activities.

The following sections of this paper discuss some common methods for generating pseudorandom data with known distribution characteristics and briefly describe twenty commonly used distributions.

Attachment A contains two MUMPS routines, *sample* and *math*. The routine named *sample* contains extrinsic functions for generating random samples from each of the twenty distributions. The routine named *math* contains extrinsic math functions that are required for some of the extrinsic sampling functions. It should be noted that the routine named *sample* makes use of MDC Type A extensions to the 1990 ANSI/MDC X11.1 MUMPS standard, specifically, exponentiation (x**y).

Attachment B contains descriptions of the calling parameters for each of the extrinsic sampling functions and plots of sample variates generated using typical parameters for each of the twenty distributions.

It is important to note that the term *pseudorandom* is used in this paper, because computer generated random values cannot be truly random. It is also important to note that each algorithm presented in this paper for generating pseudorandom variates ultimately depends on the $RANDOM intrinsic function. The quality of variates produced by these algorithms depends on the quality of the random number generator of the underlying MUMPS implementation.

Inverse Transformation Method

This method is based on the observation that for any random variable x with a cumulative density function $F(x)$, the variable $u = F(x)$ is uniformly distributed between 0 and 1. Therefore, x can be obtained by generating uniform random numbers between 0 and 1 and computing $x = F^{-1}(x)$. This method can be used to generate random variables with distributions for which F^{-1} can be determined either analytically or empirically.

For example, exponential variates can be generated using the inverse transformation method. The probability density function (pdf) for the exponential distribution is given by the following:

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

The cumulative density function (CDF) for the exponential distribution is given by the following:

$$F(x) = 1-e^{-\lambda x} = u$$

Inverting the CDF, we get the following:

$$x = -ln(1-u)/\lambda$$

Because u is uniformly distributed between 0 and 1, $1-u$ is also uniformly distributed between 0 and 1. Therefore, we can simplify the inverted equation to the following:

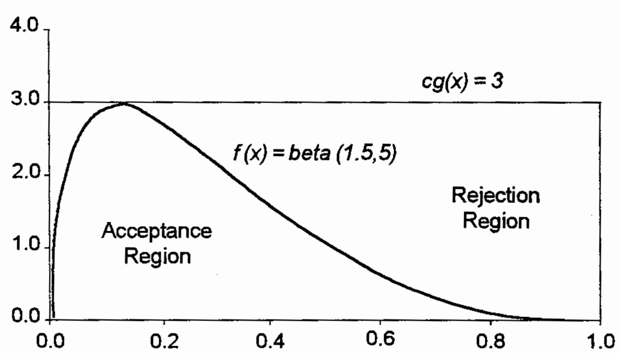
$$x = -ln(u)/\lambda$$

Using this last equation we can generate random exponential variates x_i from random uniform variates u_i.

Acceptance/Rejection Method

An acceptance/rejection method can be used to generate random variates from the distribution $f(x)$ if another distribution $cg(x)$ *majorizes* or envelops $f(x)$. That is, $cg(x) \geq f(x)$ for all values of x.

For example, consider the distribution *beta(1.5,5)*. We can use $c = 3$ and $g(x)=1$ to majorize the beta distribution as the following illustration shows:



We can then generate two uniform random variates; x on the interval from 0 to 1, and y on the interval from 0 to 3. If the point (x,y) falls above the beta density function $f(x)$, we reject it. If the point falls below $f(x)$, we accept it and return x as a random beta variate.

Composition Method

If the CDF $F(x)$ can be expressed as a weighted sum of n other CDF's, $F_1(x)$, $F_2(x)$, ... , $F_n(x)$, then we can use the composition method for generating random variates. Consider the following example:

$$F(x) = p_1F_1(x) + p_2F_2(x) + ... + p_nF_n(x)$$

In this example, $F_i(x)$ is one of the component distributions and p_i is the proportional weight for the component distribution such that the sum of the p_i's is one. Using this example, we can generate random samples from the distribution $F(x)$ by sampling from each of the component distributions $F_i(x)$ in turn with the probability p_i.

Convolution Method

If a random variable x from a distribution can be expressed as a sum of n random variables y_1, y_2, ... , y_n, then a random variate x can be generated by simply generating n random variates y_i and then summing them. That is,

$$x = y_1 + y_2 + ... + y_n$$

For example, an *Erlang(k)* random variate is the sum of k exponential random variates. A binomial variate with parameters n and p is the sum of n Bernoulli variates with probability p. A chi-square variate with v degrees of freedom is a sum of squares of v unit normal variates.

The most well known example of this method is embodied in the Central Limit Theorem, which states that *the sum of a large number of variates from* ***any*** *distribution is normally distributed.*

Characterization Method

Some distributions have special characteristics that allow variates to be generated using algorithms specially tailored for them. This method is known as characterization.

For example, the *a*th smallest number in a sequence of $a + b + 1$ *uniform(0,1)* variates has a *beta(a,b)* distribution. A chi-square variate with even degrees of freedom v is the same as a gamma variate with parameters 2 and $v/2$. If x_1 and x_2 are two gamma variates with parameters (a,b) and (a,c), then the ratio $x_1 / (x_1 + x_2)$ has a beta distribution with parameters b and c.

Commonly Used Distributions

Twenty probability distributions will now be briefly described. These distributions are commonly used in the areas of statistical analysis and computer simulation. MUMPS extrinsic functions for generating sample variates from each of these distributions are presented for your use in Attachment A. In attachment B, you will find descriptions of the calling parameters for each extrinsic sampling function, notes on the characteristics of each distribution and plots of sample variates using typical parameters for each distribution.

Bernoulli distribution The *Bernoulli distribution* is the simplest discrete distribution. A Bernoulli variate can take only two values, 0 and 1, or failure and success. The Bernoulli distribution is used to model the probability of an outcome having a desired characteristic, for example, the likelihood that a computer system is up or down. See Figure 1.

Beta distribution The *beta distribution* is used to represent random variates that are bounded, for instance, between 0 and 1. The beta distribution can be bell shaped, symmetric or asymetric, U-shaped, or linear. The beta distribution can be used to model random proportions, for example, the fraction of ethernet packets requiring retransmission. The beta distribution is often used by Bayesian statisticians in decision theory to model a prior distribution on which a subsequent binomial decision is to be based. See Figure 2.

Binomial distribution The number of successes in a sequence of n Bernoulli trials has a *binomial distribution*, for example, the number of ethernet packets that reach their destination without data loss, or the number of face cards received in a hand of poker. See Figure 3.

Chi-square distribution The *chi-square distribution* is used whenever a sum of squares of normal variables is involved. For example, the chi-square distribution could be used to model sample variances. See Figure 4.

Erlang distribution The *Erlang distribution* is often used in queueing models. It is commonly used to model patient examination durations in a clinic with n physicians. See Figure 5.

Exponential distribution The *exponential distribution* is used extensively in queueing models. For example, it might be used to model patient arrival times in a medical clinic, or the times between successive failures of a device. The exponential is unique among continuous distributions in that it is *memoryless*, the time since the last event does not help in predicting the time until the next event. See Figure 6.

F distribution The ratio of two chi-square variates has an *F distribution*. The F distribution is used to measure the ratio of sample variances. This distribution is used extensively in statistics for regression analysis and analysis of variance (ANOVA). See Figure 7.

Gamma distribution The *gamma distribution* is a generalization of the Erlang distribution where the defining parameters are not restricted to integers. The gamma distribution is used to model service times or activity durations in a manner similar to the Erlang distribution. See Figure 8.

Geometric distribution The distribution of the total number of Bernoulli trials required to obtain the first success is referred to as the *geometric distribution*. This distribution is the discrete equivalent of the continuous exponential distribution. An example of the use of the geometric distribution is modeling the number of successfully transmitted ethernet packets between packets requiring retransmission. The geometric distribution is *memoryless* in the same way as the exponential distribution. See Figure 9.

Lognormal distribution The natural logarithm of a normally distributed variate has a *lognormal distribution*. The Central Limit Theorem, which states that additive processes are normally distributed, can be used to show that multiplicative processes have a lognormal distribution. This distribution is used to model a great variety of processes in areas from biology to economics, for example, personal incomes, bank deposits, and particle sizes. See Figure 10.

Negative Binomial distribution The number of failures x in a sequence of Bernoulli trials before the mth success is obtained has a *negative binomial distribution*. An example of the use of the negative binomial distribution is modeling the number of error-free patient medications before some critical number of medication errors occurs. See Figure 11.

Normal distribution The *normal distribution* or Gaussian distribution is the most prominent distribution in probability and statistics. The normal distribution is appropriate whenever the randomness is caused by several independent sources acting additively. See Figure 12.

Pareto distribution The *Pareto distribution* is a power curve that can be easily fit to observed data. It has sometimes been used to describe income distributions. See Figure 13.

Pascal distribution The number of Bernoulli trials required to obtain m successes has a *Pascal distribution*. The Pascal distribution is an extension of the geometric distribution. For example, the Pascal distribution might be used to model the total number of attempts required to successfully send m packets across an ethernet. See Figure 14.

Poisson distribution The *Poisson distribution* is a limiting form of the binomial distribution. That is, it is analogous to the binomial distribution where the number of trials approaches infinity. The Poisson distribution is extensively used in queueing models to describe the number of arrivals over a given time interval, for example, the number of disk drive failures in a computer system over a year, or the number of walk-in patients arriving in a clinic over a year. See Figure 15.

Student's t distribution The *student's t distribution* is used whenever the ratio of a normal variate and the square root of a chi-square variable is involved. This distribution is commonly used in statistics in *t*-tests and for setting confidence intervals when estimating statistical parameters. See Figure 16.

Triangular distribution The *triangular distribution* is used whenever a variable's estimated value is described by a minimum, a maximum, and a mode. The triangular distribution is sometimes used in PERT/CPM analysis when activity durations are estimated by an optimistic time, a pessimistic time, and a most likely time. See Figure 17.

Uniform distribution (continuous and discrete) A *uniform distribution* specifies that every value between a minimum and maximum is equally likely. It generally implies a complete lack of knowledge concerning the random variable other than the minimum and maximum allowable values. If the variable can take on any value between the minimum and maximum, the distribution is *continuous*. See Figure 18. If the variable can only take on integer values, the distribution is said to be *discrete*. See Figure 19. A common use of the uniform distribution is describing the seek times for a disk drive.

Weibull distribution The *Weibull distribution* is commonly used in reliability analysis. For example, the distribution is commonly used to model lifetimes of computer components. See Figure 20.

Summary

This paper has presented MUMPS extrinsic functions for your use for generating variates from twenty common probability distributions. The author has found these functions to be useful in a variety of situations. These include generating values to deal with the problem of missing data in statistical analyses, generating sets of data for prototyping and validating statistical analysis routines, and simulating data for computer modeling. It is hoped that you will find these to be useful as well.

References

Abramowitz, M. and Stegun, A.I., Eds., *Handbook of Mathematical Functions*, Dover Publications, Inc., New York, 1972.

Jain, R., *The Art of Computer Systems Performance Analysis*, New York, John Wiley & Sons, Inc., 1991.

Pritsker, A.A.B, *Introduction to Simulation and SLAM II*, John Wiley & Sons, Inc., New York, 1986.

Attachment A

Routine *sample*: Extrinsic Functions for Generating Random Samples

```
sample   ;agb;08:31 PM  15 Dec 1992
         ;generate random sample variates
         ;
bernoull(p)        ;bernoulli distribution
         q:'$d(p) ""
         q:p<0!(p>1) ""
         q $$random()'>p
         ;
beta(alpha,beta,min,max)    ;beta distribution
         q:'$d(alpha)!'$d(beta)  ""
         q:alpha'>0 ""
         q:beta'>0 ""
         i '$d(min) n min s min=0
         i '$d(max) n max s max=1
         q:max'>min ""
         i alpha<1&(beta<1) n x,y d  q +$j(x/(x+y)-min/(max-min),0,6)
         .f  s x=$$random()**(1/alpha),y=$$random()**(1/beta)  q:x+y'>1
         n gamma1,gamma2
         s gamma1=$$gamma(1,alpha),gamma2=$$gamma(1,beta)
         q +$j(gamma1/(gamma1+gamma2)-min/(max-min),0,6)
         ;
binomial(p,n)       ;binomial distribution
         q:'$d(p)!'$d(n)  ""
         q:p'>0!(p'<1)  ""
         q:n\1'=n!(n'>0)  ""
         n i,sum
         s sum=0 f i=1:1:n s sum=sum+($$random()<p)
         q sum
         ;
chisquar(df)        ;chi-square distribution
         q:'$d(df)  ""
         q:df\1'=df!(df'>0)  ""
         i df=1 q +$j($$normal(0,1)**2,0,6)
         i df#2 q +$j($$normal(0,1)**2+$$chisquar(df-1),0,6)
         n i,prod
         s prod=1 f i=1:1:df/2 s prod=prod*$$random()
         q +$j(-$$ln^math(prod)/2,0,6)
         ;
erlang(alpha,m)     ;erlang distribution
         q:'$d(alpha)!'$d(m)  ""
         q:alpha'>0 ""
         q:m\1'=m!(m'>0)  ""
         n i,prod
         s prod=1 f i=1:1:m s prod=prod*$$random()
         q +$j(-alpha*$$ln^math(prod),0,6)
```

Attachment A

Routine ***sample***: Extrinsic Functions for Generating Random Samples (continued)

```
         ;
exponent(alpha)    ;exponential  distribution
         q:'$d(alpha)  ""
         q:alpha'>0  ""
         q  +$j(-alpha*$$ln^math($$random()),0,6)
         ;
f(dfn,dfm)         ;f  distribution
         q:'$d(dfn)!'$d(dfm)  ""
         q:dfn\1'=dfn!(dfn'>0)  ""
         q:dfm\1'=dfm!(dfm'>0)  ""
         q  +$j(($$chisquar(dfn)/dfn)/($$chisquar(dfm)/dfm),0,6)
         ;
gamma(alpha,beta)  ;gamma  distribution
         q:'$d(alpha)!'$d(beta)  ""
         q:alpha'>0  ""
         q:beta'>0  ""
         i  beta\1=beta  q  +$j($$erlang(alpha,beta),0,6)
         i  beta<1  q  +$j(alpha*$$beta(beta,1-beta)*$$exponent(1),0,6)
         q  +$j($$gamma(alpha,beta\1)+$$gamma(alpha,beta-(beta\1)),0,6)
         ;
geometri(p)        ;geometric  distribution
         q:'$d(p)  ""
         q:p'>0!(p'<1)  ""
         n g
         s  g=$$ln^math($$random())/$$ln^math(1-p)
         s:g\1'=g  g=g+1\1
         q g
         ;
lognorml(mean,sd)  ;lognormal  distribution
         q:'$d(mean)!'$d(sd)  ""
         q:mean<0  ""
         q:sd'>0  ""
         q  +$j($$exp^math(sd*$$normal(0,1)+mean),0,6)
         ;
negbinom(p,m)      ;negative  binomial  distribution
         q:'$d(p)!'$d(m)  ""
         q:p'>0!(p'<1)  ""
         q:m\1'=m!(m'>0)  ""
         n  fail,pass
         s  (fail,pass)=0 f  d  q:pass=m
         .i  $$random()>p  s  fail=fail+1  q
         .s  pass=pass+1
         q fail
         ;
```

Attachment A

Routine ***sample***: Extrinsic Functions for Generating Random Samples (continued)

```
normal(mean,sd,n)  ;normal  distribution
          q:'$d(mean)!'$d(sd)  ""
          q:sd'>0  ""
          n  i,sum
          i  '$d(n)  n  n  s  n=12
          s  sum=0  f  i=1:1:n  s  sum=sum+$$random()
          q   +$j(sum-(n/2)/((n/12)**.5)*sd+mean,0,6)
          ;
pareto(alpha)       ;pareto  distribution
          q:'$d(alpha)  ""
          q:alpha'>0  ""
          q   +$j(1/($$random()**(1/alpha)),0,6)
          ;
pascal(p,m)         ;pascal  distribution
          q:'$d(p)!'$d(m)  ""
          q:p'>0!(p'<1)  ""
          n  i,sum
          s  sum=0  f  i=1:1:m  s  sum=sum+$$geometri(p)
          q  sum
          ;
poisson(lambda)     ;poisson  distribution
          q:'$d(lambda)  ""
          q:lambda'>0  ""
          n  crit,n,prod
          s   crit=$$exp^math(-lambda),prod=1
          f  n=0:1  s  prod=prod*$$random()  q:prod<crit
          q  n
          ;
random()  ;random  (uniform  0  to  1)
          q   $r(1000001)/1000000
          ;
t(df)     ;student's  t  distribution
          q:'$d(df)  ""
          q:df\1'=df!(df'>0)  ""
          q   +$j($$normal(0,1)/(($$chisquar(df)/df)**.5),0,6)
          ;
triangle(min,mode,max)      ;triangular  distribution
          q:'$d(min)!'$d(mode)!'$d(max)  ""
          q:max'>min  ""
          q:mode<min!(mode>max)  ""
          n  crit,rndm
          s   crit=(mode-min)/(max-min),rndm=$$random()
          i   rndm'>crit  q  +$j(min+((mode-min*(max-min)*rndm)**.5),0,6)
          q   +$j(max-((max-mode*(max-min)*(1-rndm))**.5),0,6)
          ;
```

Attachment A

Routine ***sample***: Extrinsic Functions for Generating Random Samples (continued)

```
uniformc(min,max)  ;uniform  distribution  (continuous)
         q:'$d(min)!'$d(max)  ""
         q:max'>min  ""
         q  +$j(max-min*$$random()+min,0,6)
         ;
uniformd(min,max)  ;uniform  distribution  (discrete)
         q:'$d(min)!'$d(max)  ""
         q:min\1'=min  ""
         q:max\1'=max  ""
         q:max'>min  ""
         q  max-min+1*$$random()+min\1
         ;
weibull(alpha,beta)         ;wiebull  distribution
         q:'$d(alpha)!'$d(beta)  ""
         q:alpha'>0  ""
         q:beta'>0  ""
         q  +$j(alpha*(-$$ln^math($$random())**(1/beta)),0,6)
```

Continued on next page

Attachment A

Routine *math*: Extrinsic Math Functions Required for Sampling Functions

```
math      ;wlm,cfb,drh,agb;12:04 PM   15 Dec 1992
          ;math functions
          ;
exp(x,pr) ;exponential
          q:'$d(x) ""
          n l,m,n,o,p,y,lim
          s:'$d(pr) pr=6 s m=1
          s l=x,y=x+1
          s lim=$s(pr+3'>11:pr+3,1:11),@("lim=1E-"_lim)
          f o=2:1 s l=l*x/o,y=y+l q:$tr(l,"-")<lim
          q +$j(y,0,$s(pr-$l(y\1)'<0:pr-$l(y\1),1:0))
          ;
gamma(x)  ;gamma (factorial) function
          n %
          s %=$s(x>2:x-1*$$gamma(x-1),x>1:$$pgamma(x-1),x>0:1/$$rgamma(x),1:"")
          q +$j(%,0,6-$l($p(%,".")))
          ;
ln(x,pr)  ;naperian logarithm
          q:'$d(x) ""
          q:x'>0 ""
          q:x=1 0
          n l,m,n,o,p,y,lim
          s:'$d(pr) pr=11 s m=1
          i x>0 f n=0:1 q:x/m<10  s m=m*10
          i x<1 f n=0:-1 q:x/m>.1  s m=m*.1
          s x=x/m,(x,y,l)=x-1/(x+1)
          s lim=$s(pr+3'>11:pr+3,1:11),@("lim=1E-"_lim)
          f o=3:2 s l=l*x*x,m=l/o,y=m+y s:m<0 m=-m q:m<lim
          q +$j(y*2+(n*2.30258509298749),0,$s(pr-$l(y\1)'<0:pr-$l(y\1),1:0))
          ;
pgamma(x) ;polynomial approximation of gamma(1+x)
          q:'$d(x)!(x<0)!(x>1) ""
          n b,g
          s b(1)=-.577191652,b(2)=.988205891,b(3)=-.897056937,b(4)=.918206857
          s b(4)=-.756704078,b(6)=.482199394,b(7)=-.193527818,b(8)=.035868343
          s b(0)=1,g=b(8) f b=8:-1:1 s g=g*x+b(b-1)
          q +$j(g,0,6)
          ;
rgamma(x) ;1/gamma(x)
          q:'$d(x) ""
          n k,r
          s k(1)=1,k(2)=.5772156649015329,k(3)=-.6558780715202538
          s k(4)=-.0420026350340952,k(5)=.1665386113822915
          s k(6)=-.0421977345555443,k(7)=-.009621971527877
          s k(8)=.007218943246663,k(9)=-.0011651675918591
```

Attachment A

Routine ***math***: Extrinsic Math Functions Required for Sampling Functions (continued)

```
s  k(10)=-.0002152416741149,k(11)=.0001280502823882
s  k(12)=-.0000201348547807,k(13)=-.0000012504934821
s  k(14)=.000001133027232,k(15)=-.0000002056338417
s  k(16)=.000000006116095,k(17)=.0000000050020075
s  k(18)=-.0000000011812746,k(19)=.0000000001043427
s  k(20)=.0000000000077823,k(21)=-.0000000000036968
s  k(22)=.00000000000051,k(23)=-.0000000000000206
s  k(24)=-.0000000000000054,k(25)=.0000000000000014
s  k(26)=.0000000000000001
s k(0)=0,r=k(26)  f k=26:-1:1  s  r=r*x+k(k-1)
q  +$j(r,0,10)
```

Continued on next page

Attachment B

Calling Parameters for MUMPS Extrinsic Sampling Functions and Plots of Typical Distributions

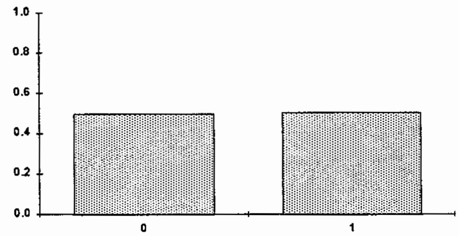
Figure 1 *Bernoulli Distribution*

```
s x=$$bernoulli^sample(p)
```

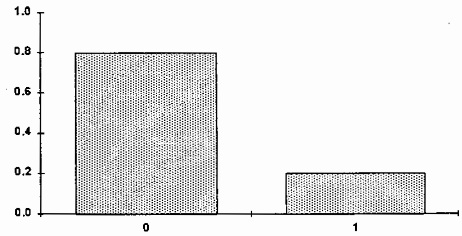
p = *probability of success, (p'<0)&(p'>1)*

Mean: p
Variance: p*(1-p)
Range: (x=0)!(x=1)

Plots of 100,000 Bernoulli variates; y-axis = sample density; x-axis = value of variate; width of interval = 1.



Bernoulli (0.5)



Bernoulli (0.2)

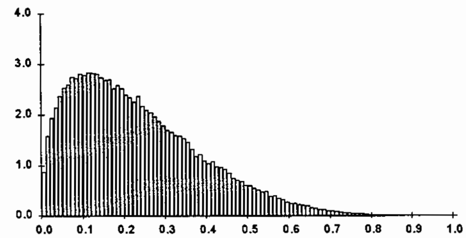
Figure 2 *Beta Distribution*

```
s x=$$beta^sample(alpha,beta)
```

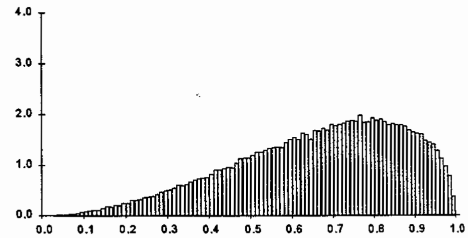
alpha = *shape parameter, alpha>0*
beta = *shape parameter, beta>0*

Mean: alpha/(alpha+beta)
Variance: (alpha*beta)(((alpha+beta)**2)
*(alpha+beta+1))
Range: (x'<0)&(x'>1)

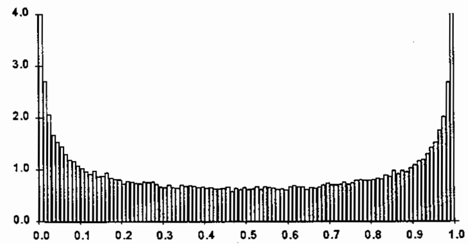
Plots of 100,000 beta variates; y-axis = sample density, truncated above 4.0; x-axis = value of variate; width of interval = 0.01.



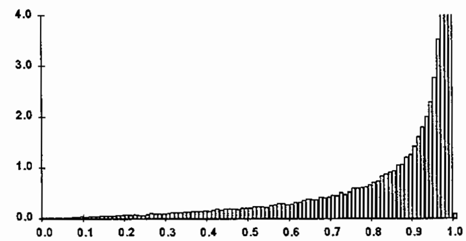
beta (1.5,5)



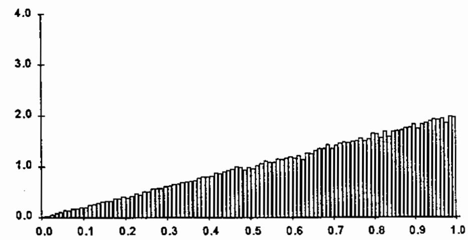
beta (3,1.5)



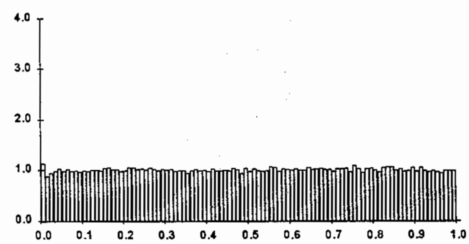
beta (0.5,0.5)



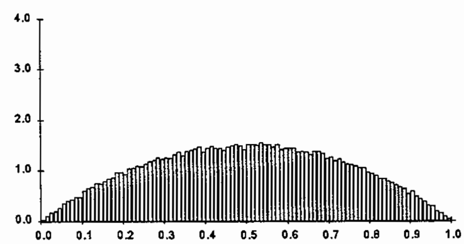
beta (2,0.2)



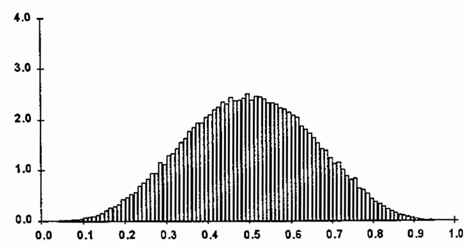
beta (2,1)



beta (1,1)



beta (2,2)



beta (5,5)

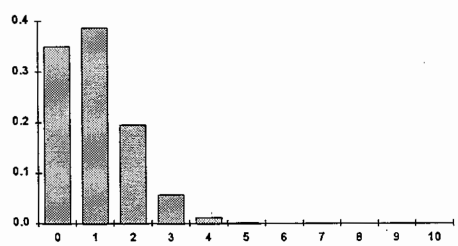
Figure 3 *Binomial Distribution*

```
s x=$$binomial^sample(p,n)
```

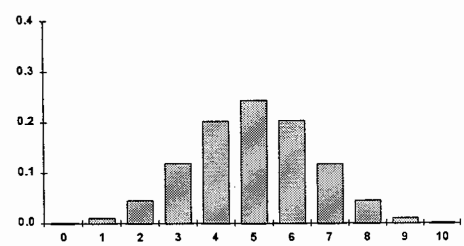
p = *probability of success in a trial, (p>0)&(p<1)*
n = *number of trials, (n>0)&(n\1=n)*

Mean: n*p
Variance: n*p*(1-p)
Range: (x'<0)&(x'>n)&(x\1=x)

Plots of 100,000 binomial variates; y-axis = sample density; x-axis = value of variate; width of interval = 1.



binomial (0.1,10)



binomial (0.5,10)

Attachment B

Calling Parameters for MUMPS Extrinsic Sampling Functions and Plots of Typical Distributions (continued)

Figure 4 *Chi-Square Distribution*

`s x=$$chisquar^sample(df)`

`df` = *degrees of freedom, (df>0)&(df\1=1)*

Mean: `df`
Variance: `2*df`
Range: `x'<0`

Plots of 100,000 chi-square variates; y-axis = sample density, truncated above 1.0; x-axis = value of variate, truncated above 10.0; width of interval = 0.1.

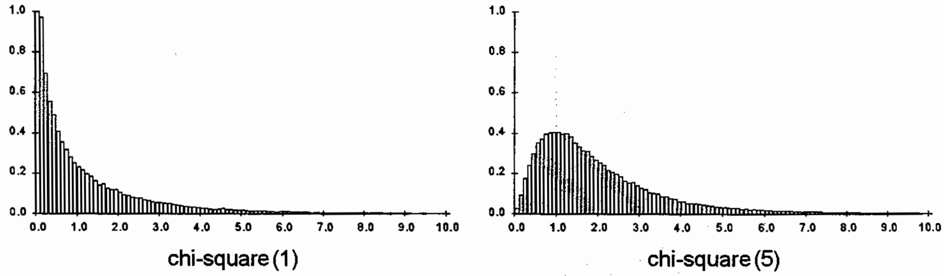


Figure 5 *Erlang Distribution*

`s x=$$erlang^sample(alpha,m)`

`alpha` = *scale parameter, alpha>0*
`m` = *shape parameter, (m>0)&(m\1=m)*

Mean: `alpha*m`
Variance: `(alpha**2)*m`
Range: `x'<0`

Plots of 100,000 Erlang variates; y-axis = sample density, truncated above 1.0; x-axis = value of variate, truncated above 10.0; width of interval = 0.1.

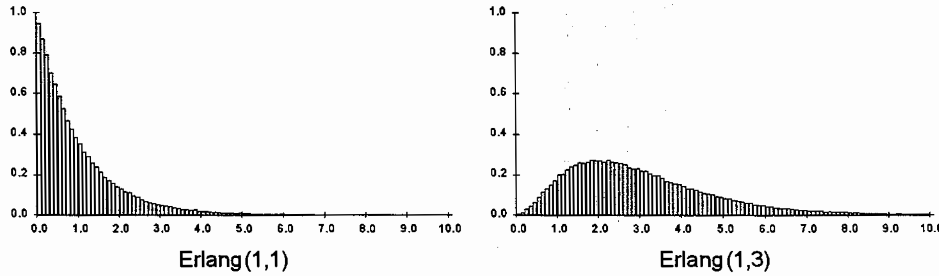


Figure 6 *Exponential Distribution*

`s x=$$exponent^sample(alpha)`

`alpha` = *scale parameter = mean, alpha>0*

Mean: `alpha`
Variance: `alpha**2`
Range: `x'<0`

Plots of 100,000 exponential variates; y-axis = sample density, truncated above 2.0; x-axis = value of variate, truncated above 6.0; width of interval = 0.1.

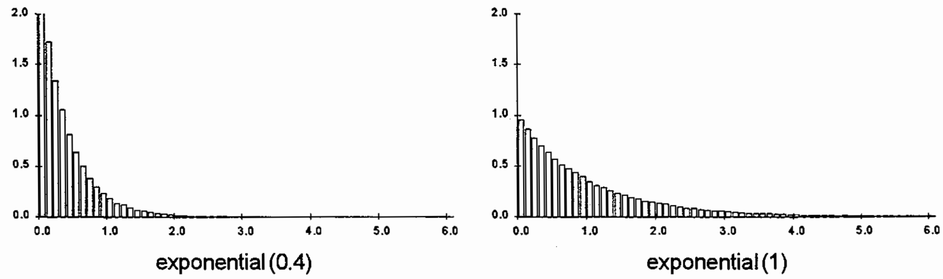


Figure 7 *F Distribution*

`s x=$$f^sample(dfn,dfm)`

`n` = *numerator degrees of freedom, (n>0)&(n\1=n)*
`m` = *denominator degrees of freedom, (m>0)&(m\1=m)*

Mean: `dfm/(dfm-2)` *for dfm>2*
Variance: `(2*(dfm**2)*(dfn+dfm-2))/(dfn*((dfm2)**2)*(dfm-4))` *for dfm>4*
Range: `x'<0`

Plots of 100,000 F variates; y-axis = sample density; x-axis = value of variate, truncated above 50; width of interval = 1.

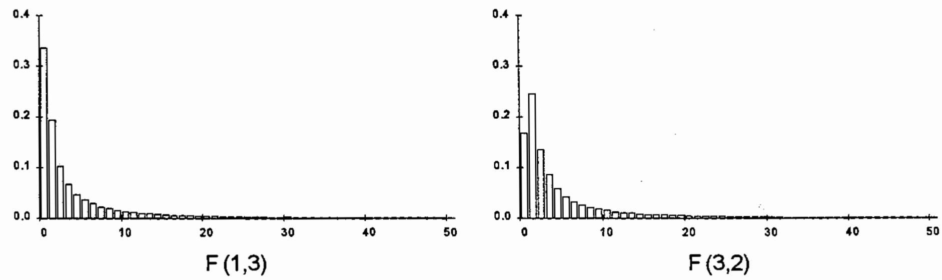


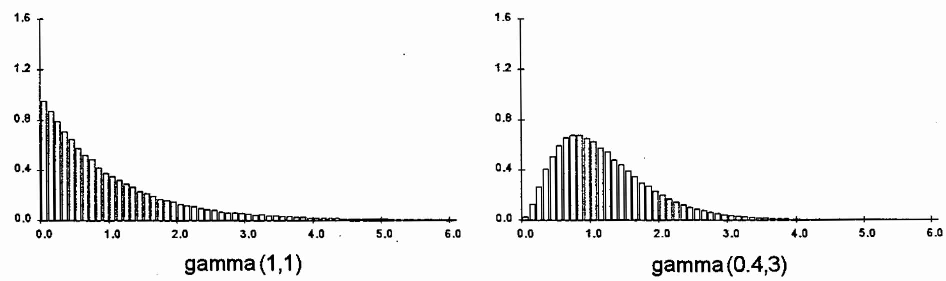
Figure 8 *Gamma Distribution*

`s x=$$gamma^sample(alpha,beta)`

`alpha` = *scale parameter, alpha>0*
`beta` = *shape parameter, beta>0*

Mean: `alpha*beta`
Variance: `(alpha**2)*beta`
Range: `x'<0`

Plots of 100,000 gamma variates; y-axis = sample density; x-axis = value of variate, truncated above 6.0; width of interval = 0.01.



Attachment B

Calling Parameters for MUMPS Extrinsic Sampling Functions and Plots of Typical Distributions (continued)

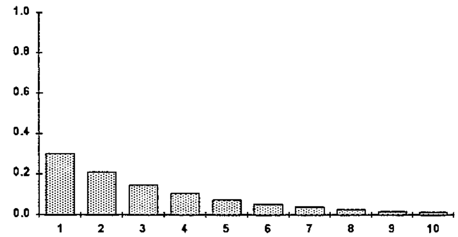
Figure 9 *Geometric Distribution*

`s x=$$geometri^sample(p)`

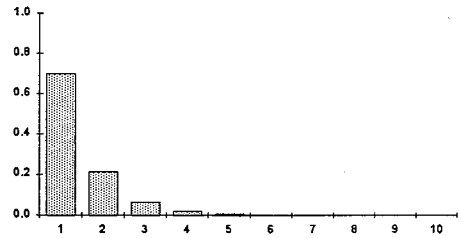
p = *probability of success, (p>0)&(p<1)*

Mean: 1/p
Variance: (1-p)/(p**2)
Range: (x>0)&(x\1=x)

Plots of 100,000 geometric variates; y-axis = sample density; x-axis = value of variate, truncated above 10; width of interval = 1.



geometric (0.3)



geometric (0.7)

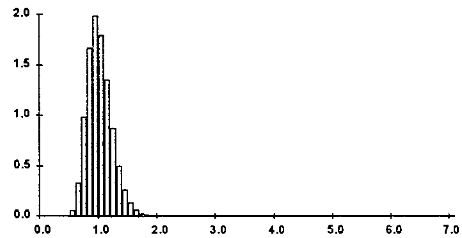
Figure 10 *Lognormal Distribution*

`s x=$$lognorml^sample(mean,sd)`

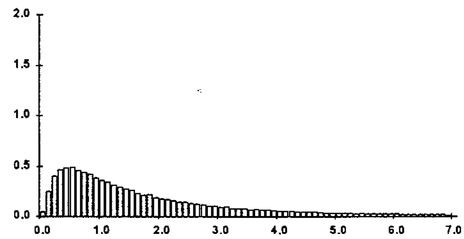
mean = *mean of $$log^math(x), mean'<0*
sd = *standard deviation of $$log^math(x), sd>0*

Mean: $$exp^math(mean+(sd**2)/2)
Variance: $$exp^math(2*mean+(sd**2))
*($$exp^math(sd**2)-1)
Range: x'<0

Plots of 100,000 lognormal variates; y-axis = sample density; x-axis = value of variate, truncated above 7.0; width of interval = 0.1.



lognormal (0,0.2)



lognormal (0.3,1)

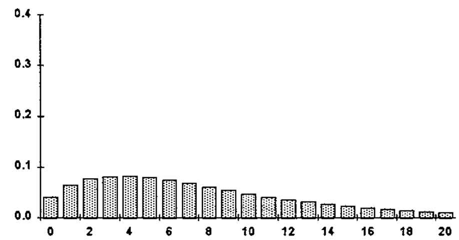
Figure 11 *Negative Binomial Distribution*

`s x=$$negbinom^sample(p,m)`

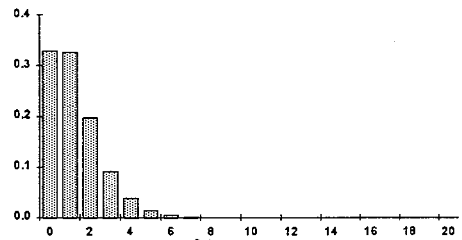
p = *probability of success, (p>0)&(p<1)*
m = *number of success, (m>0)&(m\1=m)*

Mean: m*(1-p)/p
Variance: m*(1-p)/(p**2)
Range: (x>0)&(x\1=x)

Plots of 100,000 negative binomial variates; y-axis = sample density; x-axis = value of variate, truncated above 20; width of interval = 1.



negative binomial (0.2,2)



negative binomial (0.8,5)

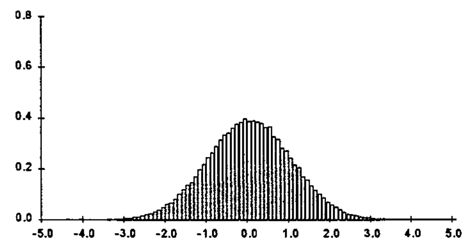
Figure 12 *Normal Distribution*

`s x=$$normal^sample(mean,sd)`

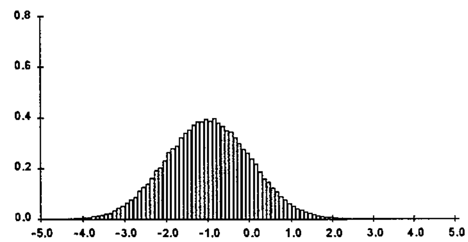
mean = *mean*
sd = *standard deviation, sd>0*

Mean: mean
Variance: sd**2
Range: *unbounded*

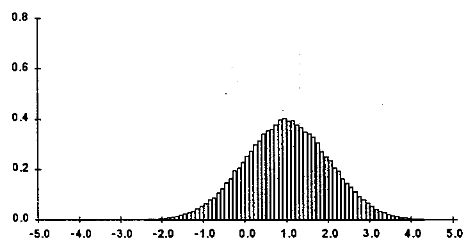
Plots of 100,000 normal variates; y-axis = sample density; x-axis = value of variate, truncated below -5.0 and above 5.0; width of interval = 0.01.



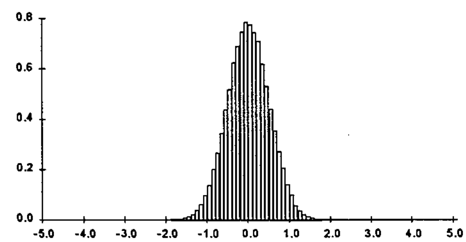
normal (0,1)



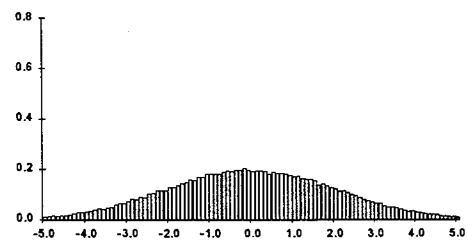
normal (-1,1)



normal (1,1)



normal (0,0.5)



normal (0,2)

Attachment B

Calling Parameters for MUMPS Extrinsic Sampling Functions and Plots of Typical Distributions (continued)

Figure 13 *Pareto Distribution*

`s x=$$pareto^sample(alpha)`

`alpha` = *shape parameter, alpha>0*

Mean: `alpha/(alpha-1)` *for alpha > 1*
Variance: `alpha/(((alpha-1)**2)*(alpha-2))` *for alpha > 2*
Range: `x'<1`

Plots of 100,000 Pareto variates; y-axis = sample density; x-axis = value of variate, truncated above 10.0; width of interval = 0.1.

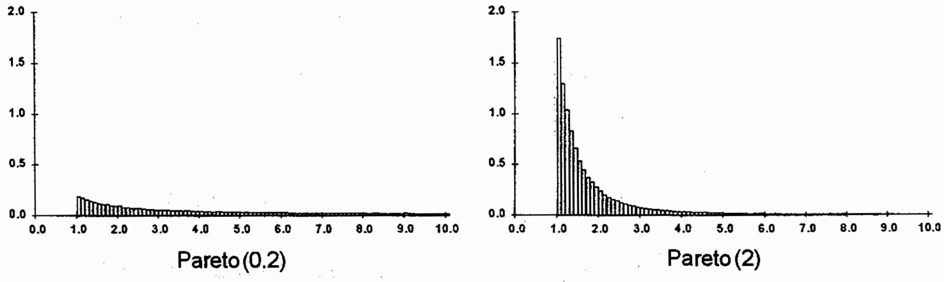


Figure 14 *Pascal Distribution*

`s x=$$pascal^sample(p,m)`

`p` = *probability of success, (p>0)&(p<1)*
`m` = *number of successes, (m>0)&(m\1=m)*

Mean: `m/p`
Variance: `m*(1-p)/(p**2)`
Range: `(x'<m)&(x\1=x)`

Plots of 100,000 Pascal variates; y-axis = sample density; x-axis = value of variate, truncated above 20; width of interval = 1.

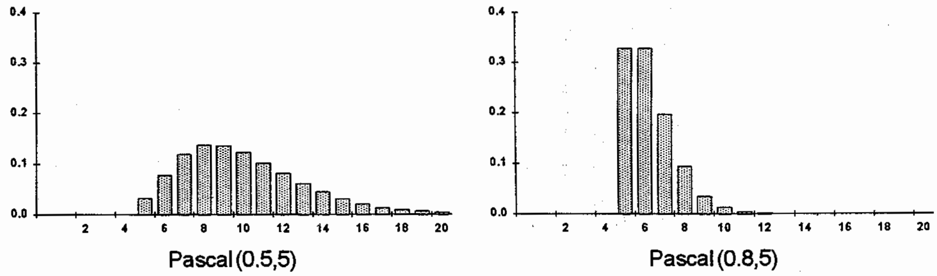


Figure 15 *Poisson Distribution*

`s x=$$poisson^sample(lambda)`

`lambda` = *mean, lambda>0*

Mean: `lambda`
Variance: `lambda`
Range: `(x'<0)&(x\1=x)`

Plots of 100,000 Poisson variates; y-axis = sample density; x-axis = value of variate, truncated above 25; width of interval = 1.

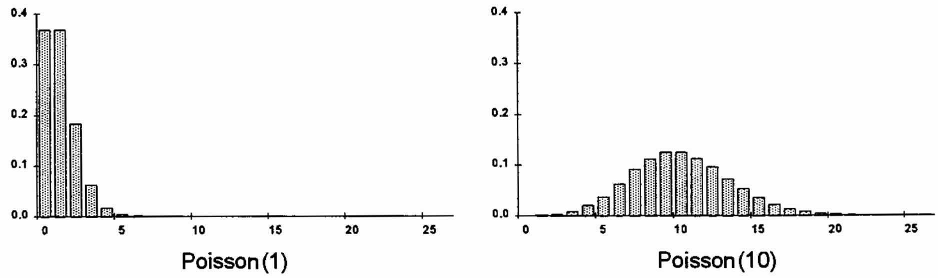


Figure 16 *Student's t Distribution*

`s x=$$t^sample(df)`

`df` = *degrees of freedom, (df>0)&(df\1=df)*

Mean: 0
Variance: `df/(df-2)`
Range: *unbounded*

Plots of 100,000 t variates; y-axis = sample density; x-axis = value of variate, truncated below -5.0 and above 5.0; width of interval = 0.1.

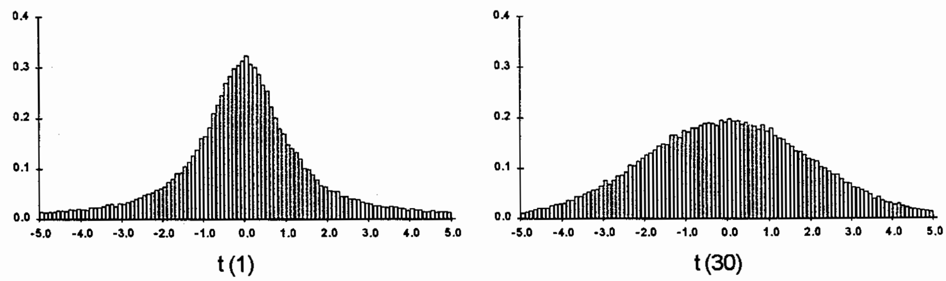


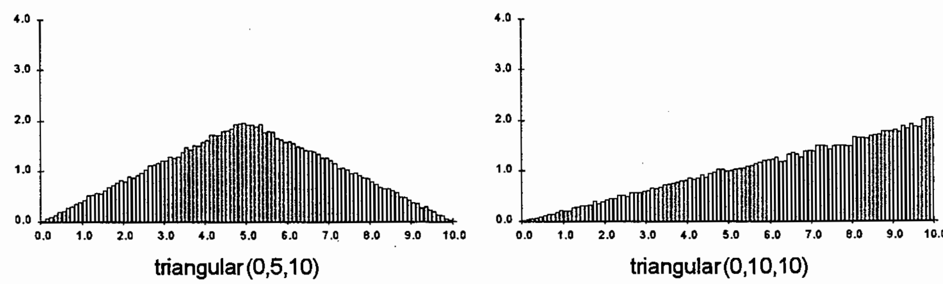
Figure 17 *Triangular Distribution*

`s x=$$triangle^sample(min,mode,max)`

`min` = *lower limit*
`mode` = *modal value, (mode'<min)&(mode'>max)*
`max` = *upper limit, max>min*

Mean: `(min+mode+max)/3`
Variance: `((min*(min-mode))+(max*(max-min))+(mode*(mode-max)))/18`
Range: `(x'<min)&(x'>max)`

Plots of 100,000 triangular variates; y-axis = proportion of variates; x-axis = value of variate; width of interval = 0.1.



Attachment B

Calling Parameters for MUMPS Extrinsic Sampling Functions and Plots of Typical Distributions (continued)

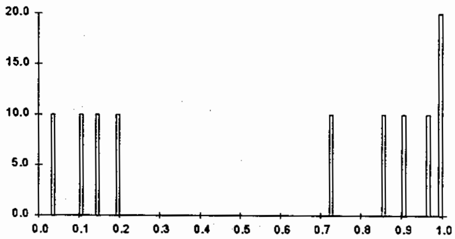
Figure 18 *Uniform Distribution (Continuous)*

```
s x=$$uniformc^sample(min,max)
```

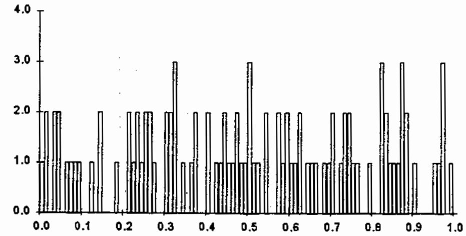
`min` = *lower limit*
`max` = *upper limit, max>min*

Mean: `(max+min)/2`
Variance: `((max-min+1)**2)/12`
Range: `(x'<min)&(x'>max)`

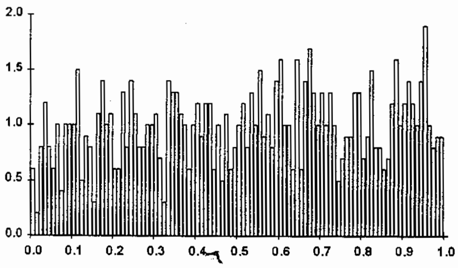
Plots of 10, 100, 1000, 10,000 and 100,000 uniform continuous variates; y-axis = sample density; x-axis = value of variate; width of interval = 0.01.



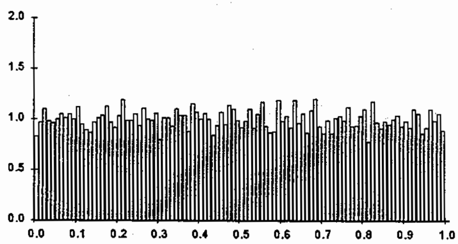
uniform (0,1) - 10 variates



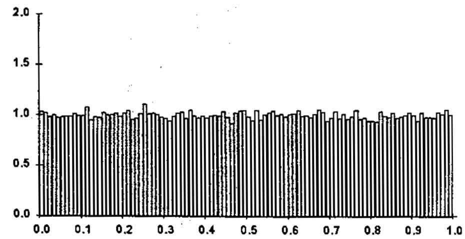
uniform (0,1) - 100 variates



uniform (0,1) - 1000 variates



uniform (0,1) - 10,000 variates



uniform (0,1) - 100,000 variates

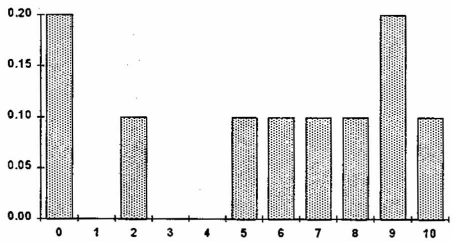
Figure 19 *Uniform Distribution (Discrete)*

```
s x=$$uniformd^sample(min,max)
```

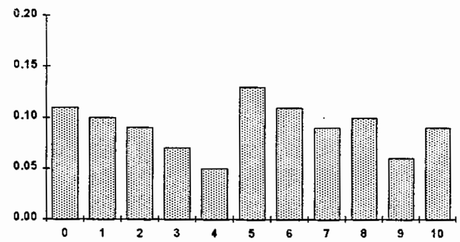
`min` = *lower limit, min\1=min*
`max` = *upper limit, (max>min)&(max\1=max)*

Mean: `(max+min)/2`
Variance: `((max-min+1)**2-1)/12`
Range: `(x'<min)&(x'>max)&(x\1=x)`

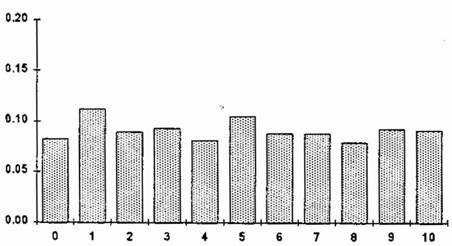
Plots of 10, 100, 1000, 10,000 and 100,000 uniform discrete variates; y-axis = sample density; x-axis = value of variate; width of interval = 1.



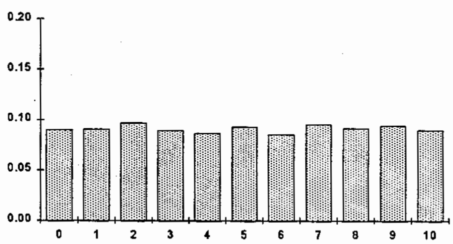
uniform (0,10) - 10 variates



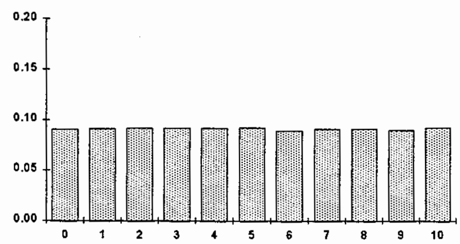
uniform (0,10) - 100 variates



uniform (0,10) - 1000 variates



uniform (0,10) - 10,000 variates



uniform (0,10) - 100,000 variates

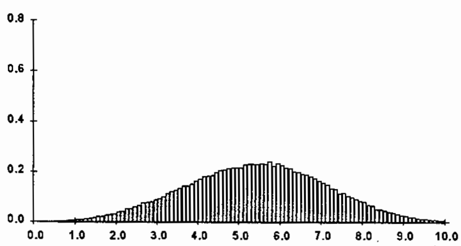
Figure 20 *Weibull Distribution*

```
s x=$$weibull^sample(alpha,beta)
```

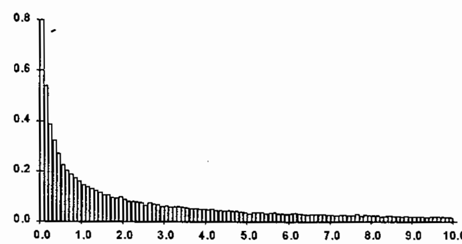
`alpha` = *scale parameter, alpha>0*
`beta` = *shape parameter, beta>0*

Mean: `(alpha/beta)*$$gamma^math(1/beta)`
Variance: `(alpha**2)(beta**2)*(2*beta*$$gamma^math(2/beta)-($$gamma^math(1/beta))**2)`
Range: `x'<0`

Plots of 100,000 Weibull variates; y-axis = sample density, truncated above 0.8; x-axis = value of variate, truncated above 10.0; width of interval = 0.1.



Weibull (6,3.602)



Weibull (2,4.602)