Subject: Re: Help wanted with log-normal distribution.
Posted by thompson on Thu, 27 Jan 2000 08:00:00 GMT
View Forum Message <> Reply to Message

Helen-Louise Windsor <hlw@ic.ac.uk> writes:

> Could anyone tell me whether IDL has a built-in command for generating a
> log-normal distribution, as I have been unable to find one?

> Failing that, would anyone be able to give me a fragment of code which
> will generate a log-normal distribution when given the two parameters?
> The reason I am asking is that I have typed in the pdf for a log-normal
> distribution from http://mathworld.wolfram.com/LogNormalDistribution.html
> but have:
> 1) obtained probabilities greater than 1
> 2) obtained a maximum probability at a value of x much greater than the
> mean.

Helen:

First of all, I'm assuming that you're asking about the Interactive Data
Language package from RSI. The proper newsgroup for that is
comp.lang.idl-pvwave, so I'm cross-posting my answer there. It may be that
somebody on that group has a better answer than I'm able to get you.

First of all, I'm a little confused as to what you're trying to do. The
equations given on the above web site look to be relatively simple to implement
within IDL. For example, the probability density function can be calculated as

 IDL> p=exp(-(alog(x)-M)^2/(2*S^2))/(x*S*sqrt(2*!pi))

with the caveat that x is greater than 0. (For x=0, simply set p=0.)
Similarly, the cumulative distribution function can be implemented as

 IDL> d=0.5*(1+errorf((alog(x)-M)/(S*sqrt(2))))

with the same caveat.

Alternatively, are you trying to generate a series of random numbers with this
particular probability distribution. I haven't thought this through
completely, but it sounds like one could generate a series of random numbers
using RANDOMN and then taking the exponential, e.g.

 IDL> Y = exp(A+B*randomn(seed,N))

with appropriate selections for A and B. On the other hand, I could be wrong,
and maybe somebody can suggest a more appropriate solution.

Another way to generate a series of random numbers with any arbitrary probability distribution between two limits is as follows:

1. Generate pairs of random numbers, labeled X and Y, where X is evenly distributed between Xmin and Xmax, and Y is evenly distributed between 0 and 1.

2. For each random number X, calculate the probability distribution function P(X).

3. Throw out all pairs for which Y is greater than P(X).

This is rather brute force, and more computationally intensive, but it can replicate any probability distribution. It does have the drawback that one has to specify limits, so that the tail of the distribution isn't replicated, but presumably one can set the limits so that this doesn't matter (much). Of course, the larger the limits, the more pairs have to be thrown away.

Hope this helps,

William Thompson