Subject: random numbers with gamma distribution
Posted by f055 on Wed, 04 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Dear All

I'm trying to generate a random series that has a gamma distribution, using
the randomu() function with the gamma keyword. The online help says to set
gamma to an integer > 0. WHY DOES IT HAVE TO BE AN INTEGER? The gamma
distribution is in fact defined for all gamma > 0, not just integers.
I want to set gamma=0.5, which currently fails as it converts it to the
integer 0, which is not > 0, and so it fails.

Is this a bug, and is there any work-around?

Cheers

Tim

..... Dr Tim Osborn . t.osborn@uea.ac.uk
.... ___/.. __ /.. /.. /. Senior Research Associate . phone:01603 592089
... /..... /. /.. /.. /.. Climatic Research Unit . fax: 01603 507784
.. /..... __/.. /.. /... School of Environmental Sciences.
. /..... /\ ... /.. /.... University of East Anglia .
_____/.._/.._.._____/..... Norwich NR4 7TJ .
..... UK .

Subject: Re: Random Numbers
Posted by Craig Markwardt on Wed, 24 Oct 2012 23:34:39 GMT
View Forum Message <> Reply to Message

On Wednesday, October 24, 2012 5:38:34 PM UTC-4, John O'Neill wrote:
> Hello Everyone,
>
>
>
> I am trying to create a set of random numbers using an Inverse Gaussian Distribution (Wald distribution) but randomu doesn't seem able to do this. Is there anything more general then randomu, or something where I can define what function I want to use to create random numbers?

http://en.wikipedia.org/wiki/Inverse_transform_sampling

I'm not aware of any IDL routines to do this though, but it's not hard with a little integrating and interpolating.

The difficulty of implementing the method depends on the distribution to some degree. For example, power law distributions can be hard to do because of the dynamic range required.

Craig

Subject: Re: Random Numbers
Posted by Yngvar Larsen on Thu, 25 Oct 2012 07:14:15 GMT
View Forum Message <> Reply to Message

On Wednesday, 24 October 2012 23:38:34 UTC+2, John O'Neill wrote:
> Hello Everyone,
>
> I am trying to create a set of random numbers using an Inverse Gaussian Distribution (Wald distribution) but randomu doesn't seem able to do this. Is there anything more general then randomu, or something where I can define what function I want to use to create random numbers?

Google and Wikipedia are your friends.

http://en.wikipedia.org/wiki/Inverse_Gaussian_distribution#G
enerating_random_variates_from_an_inverse-Gaussian_distribut ion

```
IDL> N = 100
IDL> mu = 1d0 & lambda = 1d0
IDL> nu = randomn(seed, N)
IDL> z = randomu(seed, N)
IDL> igvariates = dblarr(N)
IDL> y = nu^2
IDL> x = mu + mu^2*y/(2*lambda) - mu/2/lambda*sqrt(4*mu*lambda*y + mu^2*y^2)
IDL> ind = where(z le mu/(mu+x), complement=cind)
IDL> igvariates[ind] = x[ind]
IDL> igvariates[cind] = mu^2/x[cind]
```

Include checking for empty index arrays IND and/or CIND if you use IDL version < 8.0.

--
Yngvar

Subject: Re: Random Numbers
Posted by David on Thu, 25 Oct 2012 14:37:59 GMT
View Forum Message <> Reply to Message

Yngvar Larsen writes:

>

> On Wednesday, 24 October 2012 23:38:34 UTC+2, John O'Neill wrote:
>> Hello Everyone,
>>
>> I am trying to create a set of random numbers using an Inverse Gaussian Distribution (Wald distribution) but randomu doesn't seem able to do this. Is there anything more general then randomu, or something where I can define what function I want to use to create random numbers?
>
>
> Google and Wikipedia are your friends.
>
> http://en.wikipedia.org/wiki/Inverse_Gaussian_distribution#G
enerating_random_variates_from_an_inverse-Gaussian_distribut ion
>
> IDL> N = 100
> IDL> mu = 1d0 & lambda = 1d0
> IDL> nu = randomn(seed, N)
> IDL> z = randomu(seed, N)
> IDL> igvariates = dblarr(N)
> IDL> y = nu^2
> IDL> x = mu + mu^2*y/(2*lambda) - mu/2/lambda*sqrt(4*mu*lambda*y + mu^2*y^2)
> IDL> ind = where(z le mu/(mu+x), complement=cind)
> IDL> igvariates[ind] = x[ind]
> IDL> igvariates[cind] = mu^2/x[cind]
>
> Include checking for empty index arrays IND and/or CIND if you use IDL version < 8.0.

This code is easy enough to implement that I just wrote
a function, cgRandomWald to do it this morning. You can
find it here:

http://www.idlcoyote.com/programs/cgrandomwald.pro

It works very much like the other RandomX functions in IDL.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thue. ("Perhaps thos speakest truth.")

Subject: Re: Random Numbers
Posted by John O'Neill on Thu, 25 Oct 2012 15:06:46 GMT

On Thursday, October 25, 2012 10:37:58 AM UTC-4, David wrote:
> Yngvar Larsen writes:
>
>
>
>>
>
>> On Wednesday, 24 October 2012 23:38:34 UTC+2, John O'Neill wrote:
>
>>> Hello Everyone,
>
>>>
>
>>> I am trying to create a set of random numbers using an Inverse Gaussian Distribution (Wald distribution) but randomu doesn't seem able to do this. Is there anything more general then randomu, or something where I can define what function I want to use to create random numbers?
>
>>
>
>>
>
>> Google and Wikipedia are your friends.
>
>>
>
>> http://en.wikipedia.org/wiki/Inverse_Gaussian_distribution#G
enerating_random_variates_from_an_inverse-Gaussian_distribut ion
>
>>
>
>> IDL> N = 100
>
>> IDL> mu = 1d0 & lambda = 1d0
>
>> IDL> nu = randomn(seed, N)
>
>> IDL> z = randomu(seed, N)
>
>> IDL> igvariates = dblarr(N)
>
>> IDL> y = nu^2
>
>> IDL> x = mu + mu^2*y/(2*lambda) - mu/2/lambda*sqrt(4*mu*lambda*y + mu^2*y^2)
>
>> IDL> ind = where(z le mu/(mu+x), complement=cind)
>
>> IDL> igvariates[ind] = x[ind]

>
>> IDL> igvariates[cind] = mu^2/x[cind]
>
>>
>
>> Include checking for empty index arrays IND and/or CIND if you use IDL version < 8.0.
>
>
>
> This code is easy enough to implement that I just wrote
>
> a function, cgRandomWald to do it this morning. You can
>
> find it here:
>
>
>
> http://www.idlcoyote.com/programs/cgrandomwald.pro
>
>
>
> It works very much like the other RandomX functions in IDL.
>
>
>
> Cheers,
>
>
>
> David
>
>
>
> --
>
> David Fanning, Ph.D.
>
> Fanning Software Consulting, Inc.
>
> Coyote's Guide to IDL Programming: http://www.dfanning.com/
>
> Sepore ma de ni thue. ("Perhaps thos speakest truth.")

Thank you very much David, Yngvar, and Craig for your help. This is exactly what I was looking for.

John
