Subject: Re: L-moments
Posted by rogass on Mon, 28 Feb 2011 13:25:02 GMT
View Forum Message <> Reply to Message

On 28 Feb., 14:22, chris <rog...@googlemail.com> wrote:
> On 28 Feb., 13:13, Gray <grayliketheco...@gmail.com> wrote:
>
>
>
>
>
>
>
>
>
>
>> On Feb 28, 6:43 am, chris <rog...@googlemail.com> wrote:
>
>>> On 25 Feb., 16:25, Mark Shephard <mark.w.sheph...@gmail.com> wrote:
>
>>>> Hi,
>
>>>> I was wondering if anyone has anyone develope IDL routines for the
>>>> method of L-moments?
>
>>>> Thanks,
>>>> Mark
>
>>> Hi Mark,
>>> something like this?
>
>>> function cr_binomial,n,m
>>> n1=1d & m1=1d & n1m1=1d
>>> for i=1d,n do n1*=i
>>> for i=1d,m do m1*=i
>>> for i=1d,(n-m) do n1m1*=i
>>> return,n1/(m1*n1m1)
>>> end
>
>>> function cr_l_moment,dat
>>> n=n_elements(dat)
>>> l1 = total(dat,/double)/cr_binomial(n,1)
>>> l2=0d &l3=0d &l4=0d
>>> for i=1d,n do begin
>>> b1 = cr_binomial(i-1,1d)
>>> b2 = cr_binomial(n-i,1d)
>>> b3 = cr_binomial(i-1,2d)
>>> b4 = cr_binomial(n-i,2d)
>>> b5 = cr_binomial(i-1,3d)

```
>>>         b6      =       cr_binomial(n-i,3d)
>>>         l2+=(b1-b2)*dat[i-1l]
>>>         l3+=(b3-2*b1*b2+b4)*dat[i-1l]
>>>         l4+=(b5-3*b3*b2+3*b1*b4+b6)*dat[i-1l]
>>>  endfor
>>>  l2*=0.5d /cr_binomial(n,2d )
>>>  l3*=(1d / 3d )/cr_binomial(n,3d )
>>>  l4*=(1d / 4d )/cr_binomial(n,4d )
>>>  return,{l1:l2,l2:l2,l3:l3,l4:l4}
>>>  end
>
>>>  IDL> r=randomu(seed,5,5)
>>>  IDL> inf=cr_l_moment(r)
>>>  IDL> print,float(inf)
>>>  { -0.000558181 -0.000558181   -0.0111168     0.212071}
>
>>>  Cheers
>
>>>  CR
>
>>  I have no idea what L-moments are, but do you really need to use all
>>  the FOR-loops?
>
>>  function cr_binomial,n,m
>>    n1 = product(dindgen(n)+1)
>>    m1 = product(dindgen(m)+1)
>>    n1m1 = product(dindgen(n-m)+1)
>>    return, n1/(m1*n1m1)
>>  end
>
>  No, it is not necessary, so your suggestion reduces it to:
>
>  function cr_l_moment,dat
>  n=double(n_elements(dat))
>  bin=product(dindgen(n)+1d,/c)
>  l1 = total(dat)/bin[n-1l]
>  l2=0. &l3=0. &l4=0.
>  for i=1l,n do begin
>         b1      =       bin[i-1]
>         b2      =       bin[n-i]
>         b3      =       b1/2d
>         b4      =       b2/2d
>         b5      =       b1/6d
>         b6      =       b2/6d
>         l2+=(b1-b2)*dat[i-1l]
>         l3+=(b3-2*b1*b2+b4)*dat[i-1l]
>         l4+=(b5-3*b3*b2+3*b1*b4+b6)*dat[i-1l]
>  endfor
```

> l2*=bin[n-1]
> l3*=(1./3.)/(bin[n-1]/6d)
> l4*=(1./4.)/(bin[n-1]/24d)
> return,{l1:l2,l2:l2,l3:l3,l4:l4}
> end
>
> Cheers
>
> CR

The type of variables(double, long, float) is inconsistent. You have to change this to double.

Cheers

CR
