Subject: Re: Generating N random numbers that add to a TOTAL
Posted by Russell Ryan on Thu, 07 Aug 2014 12:58:46 GMT
View Forum Message <> Reply to Message

So, the problem with the integers is that you recast floats as integers post facto. If you want integers and you want them to sum to a total, then you need to draw integers up front --- otherwise you're noe ensuring that the round(res) is always summing to total. But this is where you're going to run into trouble...

At each step you're drawing a random number between 0 and the requested total *MINUS* the running total. If you plot the random number as a function of iteration variable, you'll see that the typical value is going down. In fact, near the end of your run, the value will be very small --- because you're converging to the requested total. Therefore, that random variable will often be zero (or 1 and rarely higher). That doesn't seem like a good thing, but maybe it is...

Russell

On Wednesday, August 6, 2014 11:52:47 PM UTC-4, Gianguido Cianci wrote:
> Hi all,
>
>
>
> I am wondering if anybody has suggestions on how to improve the function below. It seems ok for floating precision numbers.
>
>
>
> For integers it's a different story:
>
> It works great if N<<TOTAL. When N approaches TOTAL I get a few numbers and then a bunch of zeros... Also, setting /DIFFERENT makes it run for ever if N is large. Also, the sum of res adds up TOTAL=/-1, not always to TOTAL exactly...
>
>
>
> Suggestions?
>
>
>
> Thanks,
>
> Gianguido
>
>
>
>
>

```
>
>
> FUNCTION nrndaddto,  n,  total,  integers = integers,  different = different
>
>
>
> compile_opt idl2
>
>
>
> res = dblarr(n)
>
> res[0] = randomu(seed, 1,  /double)*(total)
>
>
>
> FOR i = 1,  n-2 DO BEGIN
>
>    res[i] = randomu(seed, 1,  /double)*(total-total(res[0:i-1],  /double))
>
> ENDFOR
>
> res[n-1] = total-total(res[0:n-2],  /double)
>
>
>
> IF ~keyword_set(integers) THEN integers = 0
>
>
>
> IF keyword_set(integers) THEN res = round(res)
>
> IF keyword_set(different) THEN BEGIN
>
>    IF n_elements(res) NE n_elements(unique(res,  /sort)) THEN res = $
>
>    nrndaddto(n, total, integers = integers, different = 1)
>
> ENDIF
>
>
>
>
>
> RETURN,  res
>
> END
```