

Re: Generate a sequence of random numbers that sum up to 1?

Re: Generate a sequence of random numbers that sum up to 1?

Source: <http://coding.derkeiler.com/Archive/Python/comp.lang.python/2006-04/msg03594.html>

- *From:* Edward Elliott <nobody@xxxxxxxx>
 - *Date:* Sun, 23 Apr 2006 22:55:05 GMT
-

Alex Martelli wrote:

Such total disuniformity, where the very distribution of each value is skewed by the preceding one, may still be "random" for some sufficiently vague meaning of "random", but my intuition suggests it's unlikely to prove satisfactory for the OP's purposes.

It does seem very odd. If you could restrict the range, you could get an unskewed distribution. Set range = (0, 2*sum/cnt) and you can generate cnt numbers whose sum will tend towards sum (because the average value will be sum/cnt):

```
target_sum = 1
cnt = 100
max = 2.0 * target_sum / cnt # 0.02
nums = [random.uniform(0,max) for x in range(0,cnt)]
real_sum = sum(nums) # 0.975... in one sample run
```

If the sum has to be exact, you can set the last value to reach it:

```
nums[-1] = target_sum - sum(nums[:-1])
print sum(nums) # 1.0
```

which skews the sample ever so slightly. And check for negatives in case the sum exceeded the target.

If the exact count doesn't matter, just generate random nums until you're within some delta of the target sum.

Basically, there usually better options to the problem as originally posed. Actually, now that I reread it the OP never said the range had be [0,1). So maybe we read too much into the original phrasing. If you need anything resembling a uniform distribution, scaling the results afterward is not the way to go.

.