Subject: Re: averaging over same index
Posted by K. Bowman on Tue, 23 Aug 2005 16:41:46 GMT
View Forum Message <> Reply to Message

In article <1124814140.855435.78130@f14g2000cwb.googlegroups.com>,
 andi.walther@web.de wrote:

> Hello,
>
> I have an array V(alues) and an array S(ubscripts of a target array)
> and I want to
> extract the mean of all values with the same index in order to put them
> into the new array.
>
> As a simple example:
> V = [3, 7, 99, 5, 2 , 10]
> S = [1, 3, 3, 2 , 0 ,1]
>
> new vector should be --> new = [2 , 6.5 , 5, 53]
>
> slow way would be: for n = 0 , max(S)-1 do new[n]=mean(v[where(S eq
> n)])
>
> a bit faster without WHERE in the loop:
>
> VSorted = v[sort(S)]
> SSorted = S[sort(S)]
> uu = uniq(iSorted)
> for n = 0 , n_elements(uu)-1 do new[iSorted[uu[n]]] =
> mean(vSorted[n:(n-1)>0])
>
> But is there a way to do what I want without resorting to
> a loop? In my real-world problem I have vectors with 500000 elements
> and the number of the occurence of indices
> are quite irragular and can exceed 30 times.
>
> Thanks Andi

Use HISTOGRAM with reverse indices to get lists of each s. Then average using
array subscripting for each list.

See JD's HISTOGRAM tutorial on David's web site.

http://dfanning.com/tips/histogram_tutorial.html

Ken Bowman