Subject: Re: Removing equal elements from an array
Posted by JD Smith on Thu, 17 Aug 2006 21:21:13 GMT
View Forum Message <> Reply to Message

On Thu, 17 Aug 2006 14:32:30 -0600, R.G. Stockwell wrote:

>
> "JD Smith" <jdsmith@as.arizona.edu> wrote in message
> news:pan.2006.08.17.17.59.05.354360@as.arizona.edu...
> ...
>
>> [5.5,5.5] => combo= 5505.5
>> [5.2,305.] => combo= 5505.5
>
> Good point, but I thought it was obvious to choose the correct
> multiplication
> factor such that the 2 numbers will not overlap. Six digits of lat
> or lon will give 10 meter resolution, which is probably good enough
> for many geophysical applications.

It's not just a matter of choosing the multiplication factor, it's actually truncating or rounding to integer values, to keep the "low digits" and "high digits" from polluting each other's waters.

> So, to be more precise:
>
> ; make data array
> A = fltarr(2,9)
> A[0,*]=[20.4, 40.3, 50.2, 50.2, 5.5,5.2, .2,.1,.15]
> A[1,*]=[30.2, 60.2, 32.4, 32.4, 5.5, 302,10,110,60]
> original_a = a
>
> factor = 10L^6L ; 6 digits for each value
> intfactor = 1000 ; take 3 decimal places, 10m resolution
> a = round(a * intfactor,/l64)
> combo = A[0,*]*factor + A[1,*]
> result = UNIQ(combo) ; the indices

That's a similar "cast to integers with a given precision" approach as I recommend, only using decimal digits instead of binary bits. Just wanted to make clear that it's dangerous with full precision floating points in general (a point I hadn't appreciated until playing with it a bit).

JD