

---

Subject: Re: Unique combinations from a 1d array  
Posted by [Chris Lee](#) on Fri, 16 Jan 2004 09:43:47 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

In article <1b881b7a.0401150857.37594317@posting.google.com>, "Darren"  
<dapoulio@sympatico.ca> wrote:

> <big snip>

>> return, val[where(y gt 0)]

<snip>

> The total combinations can be found using:  $n!/(n-p)!*p!$

> Where n is the total number of array elements and p is the size of the  
> desired combination in my case  $p = 2$ . Both of the code examples given  
> by Chris do this much more efficiently than what I posted. For 3000  
> cases, the matrix approach came in at 0.614s and the loop approach at  
> 2.414s on my 2.4 GHz Pentium. However, I believe there was a typo for  
> the matrix approach on the last line which should read  $i_{j+1/2}$   
> return, val[where(mask gt 0)]' to give the array of paired differences.  
> Thanks again,  
> Darren

Ah,

For the total combinations I just used  $\text{sum}(1..N)$ , which could be replaced  
with... (doh, hindsight)  $N(N-1)/2$ . Which is the same as yours for  $p=2$  (but  
without the 3000 ! )

The typo in the matrix approach....that's what happens when I use  
different variables in my IDL code to my newsgroup post :)

Chris.

---