Subject: Re: Fast Implementation
Posted by Dick Jackson on Fri, 05 Jul 2002 17:48:30 GMT
View Forum Message <> Reply to Message

"Isa Usman" <I.Usman@rl.ac.uk> wrote in message
news:ag4fjt$j6a@newton.cc.rl.ac.uk...
> Hi,
>
> I have the bit of code below which calculates the number of points in
all
> four quandrants of a 2d space. Unfortunately my arrays are very large
and it
> takes quite a while to run.Is there a way of making the code faster.

David's approach may be just what you're looking for, but if you need the results to be identical to what you gave (using GT and LT will not count points that are *equal* to the test value), the code below shows the time taken and checks that the result is still correct! I get better than 2x speedup over your original on arrays up to 5000 items.

I used two tricks here:
- you don't need the indexes returned by Where, so just use Total
- create partial results (binary arrays for X gt x0, etc.) and reuse
them

=====

```
PRO QuadrantCount

n1 = 1000
n2 = 1000
x = RandomU(seed, n1)
y = RandomU(seed, n1)

;;    Method 1

points = FltArr(n1, 4)

Print, 'Starting method 1...'
t0 = SysTime(1)

for j=0L,n1-1 do begin
 x0=X(j)
 y0=Y(j)

 index=where(X gt x0 and Y gt y0,count1)
 index=where(X lt x0 and Y gt y0,count2)
 index=where(X lt x0 and Y lt y0,count3)
```

```
 index=where(X gt x0 and Y lt y0,count4)

 na=count1
 nb=count2
 nc=count3
 nd=count4

 points(j,0:3)=float([na,nb,nc,nd])/n2

endfor

Print, SysTime(1)-t0, ' seconds'


;;    Method 2 - don't use Where, use Total

points2 = FltArr(n1, 4)

Print, 'Starting method 2...'
t0 = SysTime(1)

for j=0L,n1-1 do begin
 x0=X(j)
 y0=Y(j)

 na=Total(X gt x0 and Y gt y0)
 nb=Total(X lt x0 and Y gt y0)
 nc=Total(X lt x0 and Y lt y0)
 nd=Total(X gt x0 and Y lt y0)

 points2(j,0:3)=float([na,nb,nc,nd])/n2

endfor

Print, SysTime(1)-t0, ' seconds'
Print, Total(points2 NE points), ' errors'


;;    Method 3 - create partial results and reuse them

points3 = FltArr(n1, 4)

Print, 'Starting method 3...'
t0 = SysTime(1)

for j=0L,n1-1 do begin
 x0=X(j)
 y0=Y(j)
```

```
 xl = X lt x0
 xg = X gt x0
 yl = Y lt y0
 yg = Y gt y0

 na=Total(xg AND yg)
 nb=Total(xl AND yg)
 nc=Total(xl AND yl)
 nd=Total(xg AND yl)

 points3(j,0:3)=float([na,nb,nc,nd])/n2

endfor

Print, SysTime(1)-t0, ' seconds'
Print, Total(points3 NE points), ' errors'

END
```

=====

Hope this helps!

Cheers,
--
-Dick

Dick Jackson / dick@d-jackson.com
D-Jackson Software Consulting / http://www.d-jackson.com
Calgary, Alberta, Canada / +1-403-242-7398 / Fax: 241-7392