Subject: Re: vector of bin indices using histogram?
Posted by btt on Wed, 18 Oct 2006 13:19:13 GMT
View Forum Message <> Reply to Message

greg michael wrote:
>
> Can anyone suggest a good way to get a vector of bin indices using
> histogram?
>
> IDL> x=randomu(0,10)
> IDL> print,x
> 0.415999 0.0919649 0.756410 0.529700 0.930436
> 0.383502 0.653919 0.0668422 0.722660 0.671149
>
> I make a histogram anyway:
>
> IDL> h=histogram(x,binsize=.1)
>
> And I also want to know which bin each element went into:
>
> i.e. b=[4,0,7,5,9,3,6,0,7,6]
>
> I could calculate that from the original data of course, but I'm sure
> there must be a trick to get it out of the reverse_indices more
> efficiently (when n_elements is huge).
>

Hi,

I think this does the trick by using the LOCATION output keyword to
HISTOGRAM and then VALUE_LOCATE. I had to specify the MINIMUM bin value
to get the indices lined up to match yours.

```
x = [ 0.415999, 0.0919649,  0.756410,     0.529700,     0.930436,$
 0.383502,     0.653919,    0.0668422,     0.722660,     0.671149]

h = HISTOGRAM(x, BINSIZE = 0.1, LOC = loc, MIN = 0.0)

print, VALUE_LOCATE(loc, x)
```

Cheers,
Ben

> ---
>
> While experimenting, I came across this, which is not nice...

>
> IDL> x=randomu(0,100)*1000.
> IDL> print,histogram(x,nbins=4)
>   31   34   34   1
>
> The max value sometimes ends up in a bin of its own (usually this last
> bin is zero - I suppose it's a rounding problem).
>
> ---
>
> And then a question about reverse_indices - (I think it's not touched
> in JD's tutorial):
>
> why are the two parts shoved into a single array? Is there an
> application where this arrangement gives some benefit? Wouldn't the
> first half make more sense indexing a second separate vector without
> the need for this offset?
>
>
> regards,
> Greg
>

Subject: Re: vector of bin indices using histogram?
Posted by greg michael on Wed, 18 Oct 2006 14:00:09 GMT
View Forum Message <> Reply to Message

Thanks Ben - I never met that function before! Unfortunately, it's using a bisection search, and comes out a little slower than the the direct calculation:

```
pro test2,n
 x=randomu(0,n)
 h = HISTOGRAM(x, BINSIZE = 0.1, LOC = loc, MIN = 0.0)

 t=systime(/seconds)
 mx=max(x,min=mn)
 b=fix((x-mn)/(mx-mn)*10)
 print,"direct calc",systime(/seconds)-t

 t=systime(/seconds)
 b=VALUE_LOCATE(loc, x)
 print,"value_locate",systime(/seconds)-t
end

IDL> test2,5e7
direct calc       1.5470002
```

value_locate 2.8750000

Well, maybe the direct calculation isn't so inefficient. But histogram must have known those numbers during its calculation. It's a pity they got thrown away.

Greg

Subject: Re: vector of bin indices using histogram?
Posted by Paolo Grigis on Wed, 18 Oct 2006 14:47:20 GMT
View Forum Message <> Reply to Message

Is it fair to add the time used for computing the min & max of the data into the total time for the direct method? When using real data you might already know them, or else you can get them directly out of the histogram via the omin and omax keywords....

You might also want to change the computation of b with a division into a multiplication by the reciprocal (see example below).

```
x=fltarr(5d7)

t=systime(/seconds)
y=x/2.
print,systime(/seconds)-t
       0.82191920

t=systime(/seconds)
y=x*(1./2)
print,systime(/seconds)-t
      0.33465910
```

Ciao,
Paolo

greg michael wrote:
> Thanks Ben - I never met that function before! Unfortunately, it's
> using a bisection search, and comes out a little slower than the the
> direct calculation:
>
> pro test2,n
> x=randomu(0,n)
> h = HISTOGRAM(x, BINSIZE = 0.1, LOC = loc, MIN = 0.0)
>

> t=systime(/seconds)
> mx=max(x,min=mn)
> b=fix((x-mn)/(mx-mn)*10)
> print,"direct calc",systime(/seconds)-t
>
> t=systime(/seconds)
> b=VALUE_LOCATE(loc, x)
> print,"value_locate",systime(/seconds)-t
> end
>
> IDL> test2,5e7
> direct calc   1.5470002
> value_locate   2.8750000
>
> Well, maybe the direct calculation isn't so inefficient. But histogram
> must have known those numbers during its calculation. It's a pity they
> got thrown away.
>
> Greg
>

Subject: Re: vector of bin indices using histogram?
Posted by David Fanning on Wed, 18 Oct 2006 15:08:59 GMT
View Forum Message <> Reply to Message

Paolo Grigis writes:

> You might also want to change the computation of b with
> a division into a multiplication by the reciprocal (see
> example below).
>
> x=fltarr(5d7)
>
> t=systime(/seconds)
> y=x/2.
> print,systime(/seconds)-t
>   0.82191920
>
> t=systime(/seconds)
> y=x*(1./2)
> print,systime(/seconds)-t
>   0.33465910

Well, that's interesting. Do you have a theory about this? :-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: vector of bin indices using histogram?
Posted by Foldy Lajos on Wed, 18 Oct 2006 15:33:11 GMT
View Forum Message <> Reply to Message

On Wed, 18 Oct 2006, David Fanning wrote:

> Paolo Grigis writes:
>
>> You might also want to change the computation of b with
>> a division into a multiplication by the reciprocal (see
>> example below).
>>
>> x=fltarr(5d7)
>>
>> t=systime(/seconds)
>> y=x/2.
>> print,systime(/seconds)-t
>> 0.82191920
>>
>> t=systime(/seconds)
>> y=x*(1./2)
>> print,systime(/seconds)-t
>> 0.33465910
>
> Well, that's interesting. Do you have a theory about this? :-)
>

CPU latency. Eg. for Pentium 4, the latency is 7 clock cycles for
FMUL, and 43 for FDIV (this is worst case, depends on the data, and
assumes that the data is in the L1 cache). Decent compilers (including
FL :-) replace division by float const by multiplication.

regards,
lajos

Subject: Re: vector of bin indices using histogram?
Posted by greg michael on Wed, 18 Oct 2006 15:38:32 GMT
View Forum Message <> Reply to Message

Yes, you're right - I do have those - thanks.

The second trick is intriguing, but I can't reproduce it:

```
IDL> p=randomu(0,1e8)
IDL> t=systime(1) & b=p*.5 &print,systime(1)-t
      0.56299996
IDL> t=systime(1) & b=p/2. &print,systime(1)-t
      0.54699993
IDL> t=systime(1) & b=p*(1/2.) &print,systime(1)-t
      0.56299996
```

Could it be architecture-dependent?

Greg

Subject: Re: vector of bin indices using histogram?
Posted by Foldy Lajos on Wed, 18 Oct 2006 15:53:21 GMT
View Forum Message <> Reply to Message

> CPU latency. Eg. for Pentium 4, the latency is 7 clock cycles for FMUL, and
> 43 for FDIV (this is worst case, depends on the data, and assumes that the
> data is in the L1 cache). Decent compilers (including
> FL :-) replace division by float const by multiplication.
>
> regards,
> lajos
>

oops, I have to correct myself: FDIV latency is 23 clock cycles for float,
38 for double, and 43 for long double. Anyway, it is greater than 7.

regards,
lajos

Subject: Re: vector of bin indices using histogram?
Posted by David Fanning on Wed, 18 Oct 2006 16:06:08 GMT
View Forum Message <> Reply to Message

=?ISO-8859-2?Q?F=D6LDY_Lajos?= writes:

> oops, I have to correct myself: FDIV latency is 23 clock cycles for float,
> 38 for double, and 43 for long double. Anyway, it is greater than 7.

Well, the multiplication is actually a bit faster on
my machine (Windows) than the division. So I'm not
at all sure how generalized this result is.

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: vector of bin indices using histogram?
Posted by Foldy Lajos on Wed, 18 Oct 2006 16:25:13 GMT
View Forum Message <> Reply to Message

On Wed, 18 Oct 2006, David Fanning wrote:

> =?ISO-8859-2?Q?F=D6LDY_Lajos?= writes:
>
>> oops, I have to correct myself: FDIV latency is 23 clock cycles for float,
>> 38 for double, and 43 for long double. Anyway, it is greater than 7.
>
> Well, the multiplication is actually a bit faster on
> my machine (Windows) than the division. So I'm not
> at all sure how generalized this result is.
>
> Cheers,
> David

A little experiment with a surprising result, on a Pentium D 3.4 GHz with
linux and IDL 6.2. The array size is small to avoid memory access latency.

regards,
lajos

```
; md.pro  <- cut here
a=sin(findgen(1000))*1e38
nrep=1000000l

t=systime(1)
for j=1l,nrep do b=a/2.
print, 'DIV: ', systime(1)-t

t=systime(1)
```

```
for j=1l,nrep do b=a*0.5
print, 'MUL: ', systime(1)-t

end
; md.pro  <- cut here


IDL> .ru md
% Compiled module: $MAIN$.
DIV:       13.824564
MUL:        2.1084599
IDL> .ru md
% Compiled module: $MAIN$.
DIV:       13.793007
MUL:        2.0625601
IDL> .ru md
% Compiled module: $MAIN$.
DIV:       13.829693
MUL:        2.1155751
IDL>
```

Subject: Re: vector of bin indices using histogram?
Posted by Paolo Grigis on Wed, 18 Oct 2006 16:27:17 GMT
View Forum Message <> Reply to Message

Yes, I expect this to depend strongly on the arcitecture
(all under LINUX or UNIX).

Compare (I had to change the sizes a bit to make the
arrays fit in real memory) these 3 different architectures:

intel pentium 4 (idl 6.2):

```
p=fltarr(5d7)
t=systime(1) & b=p*.5 &print,systime(1)-t
      0.33394289
t=systime(1) & b=p/2. &print,systime(1)-t
      0.82017112
```

Ratio: 2.5

intel xeon dual core (idl 6.3):

```
p=fltarr(1d8)

IDL> t=systime(1) & b=p*0.5 &print,systime(1)-t
      0.50077415
```

IDL> t=systime(1) & b=p/2. &print,systime(1)-t
      0.61366391
Ratio: 1.2

sun sparc (a bit older, idl 5.4):

p=fltarr(1d7)

t=systime(1) & b=p*0.5 &print,systime(1)-t
      0.29103994
t=systime(1) & b=p/2. &print,systime(1)-t
      0.43296313

Ratio: 1.5

Ciao,
Paolo

greg michael wrote:
> Yes, you're right - I do have those - thanks.
>
> The second trick is intriguing, but I can't reproduce it:
>
> IDL> p=randomu(0,1e8)
> IDL> t=systime(1) & b=p*.5 &print,systime(1)-t
>   0.56299996
> IDL> t=systime(1) & b=p/2. &print,systime(1)-t
>   0.54699993
> IDL> t=systime(1) & b=p*(1/2.) &print,systime(1)-t
>   0.56299996
>
> Could it be architecture-dependent?
>
> Greg
>

Subject: Re: vector of bin indices using histogram?
Posted by Paolo Grigis on Wed, 18 Oct 2006 16:56:27 GMT
View Forum Message <> Reply to Message

Yes, you're right, we should use smaller arrays such
that everything fits into the cache.

Using your md.pro, the ratio DIV/MUL in my systems

are about:

pentium 4 : 6.6
xeon dual : 5.0
sparc : 2.9

Anybody with an AMD processor around?

Ciao,
Paolo

Fï¿½LDY Lajos wrote:
>
> On Wed, 18 Oct 2006, David Fanning wrote:
>
>> =?ISO-8859-2?Q?F=D6LDY_Lajos?= writes:
>>
>>> oops, I have to correct myself: FDIV latency is 23 clock cycles for
>>> float,
>>> 38 for double, and 43 for long double. Anyway, it is greater than 7.
>>
>>
>> Well, the multiplication is actually a bit faster on
>> my machine (Windows) than the division. So I'm not
>> at all sure how generalized this result is.
>>
>> Cheers,
>> David
>
>
> A little experiment with a surprising result, on a Pentium D 3.4 GHz
> with linux and IDL 6.2. The array size is small to avoid memory access
> latency.
>
> regards,
> lajos
>
>
> ; md.pro <- cut here
> a=sin(findgen(1000))*1e38
> nrep=1000000l
>
> t=systime(1)
> for j=1l,nrep do b=a/2.
> print, 'DIV: ', systime(1)-t
>
> t=systime(1)
> for j=1l,nrep do b=a*0.5

> print, 'MUL: ', systime(1)-t
>
> end
> ; md.pro <- cut here
>
>
> IDL> .ru md
> % Compiled module: $MAIN$.
> DIV: 13.824564
> MUL: 2.1084599
> IDL> .ru md
> % Compiled module: $MAIN$.
> DIV: 13.793007
> MUL: 2.0625601
> IDL> .ru md
> % Compiled module: $MAIN$.
> DIV: 13.829693
> MUL: 2.1155751
> IDL>
>

Subject: Re: vector of bin indices using histogram?
Posted by Jean H. on Wed, 18 Oct 2006 17:09:58 GMT
View Forum Message <> Reply to Message

> intel pentium 4 (idl 6.2):
>
> p=fltarr(5d7)
> t=systime(1) & b=p*.5 &print,systime(1)-t
> 0.33394289
> t=systime(1) & b=p/2. &print,systime(1)-t
> 0.82017112
>
> Ratio: 2.5

IDL> p=fltarr(5d7)
IDL> t=systime(1) & b=p*.5 &print,systime(1)-t
0.23399997
IDL> t=systime(1) & b=p/2. &print,systime(1)-t
0.25000000
.... ratio = 1.06 here... Pentium 4, windows XP pro, idl 6.3
I guess the operating system has some influence too?!

Jean

Subject: Re: vector of bin indices using histogram?
Posted by C. E. Ordonez on Wed, 18 Oct 2006 17:57:12 GMT
View Forum Message <> Reply to Message

Paolo Grigis wrote:
> Yes, you're right, we should use smaller arrays such
> that everything fits into the cache.
>
> Using your md.pro, the ratio DIV/MUL in my systems
> are about:
>
> pentium 4 : 6.6
> xeon dual : 5.0
> sparc : 2.9
>
> Anybody with an AMD processor around?
>

model name : AMD Athlon(tm) 64 X2 Dual Core Processor 3800+

IDL> print, !VERSION
{ x86 linux unix linux 6.1.1 Oct 11 2004 32 64}
IDL> .run md
% Compiled module: MD.
IDL> md
DIV: 4.1426389
MUL: 1.9575839

Subject: Re: vector of bin indices using histogram?
Posted by Foldy Lajos on Wed, 18 Oct 2006 18:12:18 GMT
View Forum Message <> Reply to Message

On Wed, 18 Oct 2006, Paolo Grigis wrote:

> Yes, you're right, we should use smaller arrays such
> that everything fits into the cache.
>
> Using your md.pro, the ratio DIV/MUL in my systems
> are about:
>
> pentium 4 : 6.6
> xeon dual : 5.0
> sparc : 2.9
>
> Anybody with an AMD processor around?
>
> Ciao,

> Paolo
>

Opteron 142, linux, IDL 6.2:

IDL> .ru md
% Compiled module: $MAIN$.
DIV: 5.9537580
MUL: 3.1484690

FL 0.60j without SSE:

FL> .ru md
% Compiled routine: $MAIN$
DIV: 2.1138449
MUL: 2.1111529

FL 0.60j with SSE (default):

FL> .ru md
% Compiled routine: $MAIN$
DIV: 1.2792230
MUL: 1.2820880

Subject: Re: vector of bin indices using histogram?
Posted by greg michael on Wed, 18 Oct 2006 21:07:11 GMT
View Forum Message <> Reply to Message

Athlon XP 2600+

DIV: 3.8590002
MUL: 4.2659998

Seems to be unique in dividing faster than multiplying...

Greg

Subject: Re: vector of bin indices using histogram?
Posted by greg michael on Wed, 18 Oct 2006 21:42:35 GMT
View Forum Message <> Reply to Message

The last odd result prompted me to fix a problem with the FSB clock

speed (for some peculiar reason it resets to 11.5x100 MHz once in a
while), so now back at 16*166 (or something like that), I get:

DIV: 2.3130000
MUL: 2.3429999

DIV still winning marginally, but with a different factor...

greg michael wrote:
> Athlon XP 2600+
>
> DIV: 3.8590002
> MUL: 4.2659998
>
> Seems to be unique in dividing faster than multiplying...
>
> Greg

Subject: Re: vector of bin indices using histogram?
Posted by Paolo Grigis on Thu, 19 Oct 2006 07:45:20 GMT
View Forum Message <> Reply to Message

Jean H. wrote:
>> intel pentium 4 (idl 6.2):
>>
>> p=fltarr(5d7)
>> t=systime(1) & b=p*.5 &print,systime(1)-t
>>   0.33394289
>> t=systime(1) & b=p/2. &print,systime(1)-t
>>   0.82017112
>>
>> Ratio: 2.5
>
>
> IDL> p=fltarr(5d7)
> IDL> t=systime(1) & b=p*.5 &print,systime(1)-t
>   0.23399997
> IDL> t=systime(1) & b=p/2. &print,systime(1)-t
>   0.25000000
> .... ratio = 1.06 here... Pentium 4, windows XP pro, idl 6.3
> I guess the operating system has some influence too?!

Somehow it looks like that under windows the division
is optimized away, whereas under linux it is not...
I guess that this is one of the reasons why IDL is faster
under windows than linux.

Ciao,
Paolo

>
> Jean