Subject: Re: Frustrated by 2 Data Plotting problems
Posted by Kenneth P. Bowman on Wed, 25 May 2011 20:36:52 GMT
View Forum Message <> Reply to Message

In article
<709fffde-7bcd-47db-b69c-b5a5ea4fe658@l18g2000yql.googlegroups.com>,
almost_like_a_metaphor <bronze.enigma@gmail.com> wrote:

> First: trying to figure out how to plot data and color code by
> value.
>
> I have x,y, and, z values in 3 columns (actually lons, lats, and
> intensities of a profile)
>
> In my simple minded way I wanted to plot x vs y, and set the color of
> the dot to a value for z.

If I understand correctly, what you want is PLOTS. It takes a COLOR keyword that can be a vector with different colors for every point. You will need to make the color vector yourself.

Setup your map with MAP_SET, then call PLOTS.

Also, with PLOTS you often need to set NOCLIP = 0. That double negative turns on clipping, which is off by default.

If you are a beginner, you might like my book

http://idl.tamu.edu/idl/Home.html

which you can get from Amazon, among other places.

Ken Bowman

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Wed, 25 May 2011 20:57:59 GMT
View Forum Message <> Reply to Message

almost_like_a_metaphor writes:

> I have 2 IDL plotting questions that have been giving me lots of
> frustrations. I just upgraded to 8.1 (Mac), so can use whatever is
> available.
>
> I'm guessing these have both been done a zillion times, but I'm new
> here (sorry, hi), and am having a devil of a time.
>

> First: trying to figure out how to plot data and color code by
> value.
>
> I have x,y, and, z values in 3 columns (actually lons, lats, and
> intensities of a profile)
>
> In my simple minded way I wanted to plot x vs y, and set the color of
> the dot to a value for z.
>
> The only way I can think of doing this right now is to step through
> every row and oplot each pair x[i],y[i] with the color for that pair
> taken from z[i]. This makes for a long executing loop with large
> files.
>
> I'm guessing I'm missing doing something with a mapping procedure
> (which I would be willing to learn), but is there any way to simply
> turn the color value of plotting into a vector the same size as x&y?

If you wanted to download the Coyote Library (highly recommended),
you could do something like this.

```
   data = cgDemoData(14)
   Help, data
   ;  The data will be in three columns, lon, lat, data.
   cgLoadCT, 33
   cgPlot, data[0,*], data[1,*], /YNoZero, /NoData
   cgPlotS, data[0,*], data[1,*], PSym=16, SymSize=2.0, $
      SymColor=BytScl(data[2,*])
```

You can find the Coyote Library here:

   http://www.idlcoyote.com/code_tips/installcoyote.php

> Second:
> I have many files I need to repeat processes with, generating data I
> want to overplot on multiple plots. So far, it looks like I have to
> make a choice: Load in one file and create my different plots for that
> file, or load in all files and extract the data from each file for one
> plot at a time, then going back again through all the files to create
> the next plot.
>
> A more detailed version of the problem.
> From any given Data (D1, D2, D3....D100) I extract parameters P1, P2,
> P3 that I want to plot.
> As of now, I can loop through D1-D100 and overplot derived parameters
> for P1, but in order to get P2, I have to loop through again. I can't
> seem to plot P1, P2, and P3 from D1, then cycle through and overplot

> data from D2 onto the correct plots.
>
> I hope that description makes sense. If not I can elaborate.

Well, this is harder to explain. How much money do you have?
I'm happy to sell you a book that explains everything you ever
wanted to know about IDL graphics. :-)

http://www.idlcoyote.com/books/

Basically, when you create a plot, IDL stores information about that plot (so you can overplot on it, for example) in system variables. These system variables are overwritten each time you create a plot. So, if you wanted to go back and overplot on a *previous* plot (i.e., not the *last* one you drew), you would have to save those system variables that were set when you drew the plot and reset them to overplot.

Here is an article about drawing multi-plots, but the concept is the same.

http://www.idlcoyote.com/tips/oplot_pmulti.html

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Wed, 25 May 2011 21:01:28 GMT

View Forum Message <> Reply to Message

David Fanning writes:

> I'm happy to sell you a book that explains everything you ever
> wanted to know about IDL graphics. :-)
>
> http://www.idlcoyote.com/books/

Well, a book that contains everything *I* know about
graphics, anyway. I don't really know what you want
to know, I guess. :-)

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by Mark Piper on Thu, 26 May 2011 01:41:32 GMT

View Forum Message <> Reply to Message

Here's an example for the first problem using Direct and (New) Graphics. Works in IDL 8.1.

```
; Fake data at lon-lat grid nodes.
n = 100L ; try 1000L
lat = findgen(n)/(n-1)*180.0 - 90.0
lon = findgen(2*n)/(2*n-1)*360.0 - 180.0
z = cos(lat*!dtor) ## (1.0 + 0.05*randomn(1, 2*n))

; Need expansion/linearization of grid to get correct dimensions for PLOT / PLOTS.
glat = reform(lat ## (fltarr(2*n) + 1.0), 2*n^2)
glon = reform((fltarr(n) + 1.0) ## lon, 2*n^2)
gcolors = bytscl(reform(z, 2*n^2))

help, lon, lat, z, glon, glat, gcolors

; (New) Graphics.
m = map('Robinson')
g = plot(glon, glat, $
   linestyle='none', $
   symbol='.', $
   /overplot, $
   rgb_table=5, $
   vert_colors=gcolors)

; Direct Graphics.
loadct, 5
map_set, /robinson
plots, glon, glat, psym=3, color=gcolors
```

Subject: Re: Frustrated by 2 Data Plotting problems

Posted by cgguido on Thu, 26 May 2011 19:54:03 GMT

View Forum Message <> Reply to Message

```
>
>     data = cgDemoData(14)
>     Help, data
>     ;  The data will be in three columns, lon, lat, data.
>     cgLoadCT, 33
>     cgPlot, data[0,*], data[1,*], /YNoZero, /NoData
>     cgPlotS, data[0,*], data[1,*], PSym=16, SymSize=2.0, $
>        SymColor=BytScl(data[2,*])
>
```

So say you wanted to colour the points based on a 2D histogram of the data, so that when the overplotting fills a part of the plot, at least colour gives you an indication that there is a higher density of dots... (see http://tinyurl.com/3kv9kdm and sorry for the partial thread hijack!) Would you go about it in a similar way or is there a faster way?

Here's what I came up with, using sshist_2d.pro (http://tinyurl.com/3on7bzx) that automagically finds bin size:

```
h2=sshist_2d(x,y, re=ri1, cost=co)

col=x*0
nn=n_elements(h2)
for b=0L, nn-1 do begin &$
  w=histobin(ri1,b) &$
  if w[0] ne -1 then col[w]=h2[b] &$
endfor


cgloadct, 33

set_plot, 'z'
device, z_buff=0, set_res=[!D.X_SIZE,!D.Y_SIZE]
cgplot, x, y, /noda, back=cgcolor('black'), $
color=cgcolor('white'), chars=1.5
cgplots, x, y, ps=16, syms=.1, symcol=bytscl(col)
a=tvrd(/tr)
set_plot, 'x'
tv, a, /tr
```

Subject: Re: Frustrated by 2 Data Plotting problems

Posted by Jeremy Bailin on Thu, 26 May 2011 20:57:36 GMT

View Forum Message <> Reply to Message

```
>  col=x*0
>  nn=n_elements(h2)
>  for b=0L, nn-1 do begin &$
>    w=histobin(ri1,b) &$
>    if w[0] ne -1 then col[w]=h2[b] &$
>  endfor
```

Assuming that you can get the minimum x and y values (say minx and miny) and the bin size (say xbin and ybin) out of sshist_2d, then the following should work and be faster:

```
h2size = size(h2, /dimen)
col = h2[ floor((x-xmin)/xbin) + floor((y-ymin)/ybin)*h2size[0] ]
```

-Jeremy.

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by cgguido on Thu, 26 May 2011 21:46:07 GMT

View Forum Message <> Reply to Message

Thanks Jeremy, your code generates the colors much faster indeed, but unfortunately the bottleneck is cgPlotS...

I am wondering if I could batch cgPlotS all points that have the same colour to speed things up...

Gianguido

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Thu, 26 May 2011 22:16:14 GMT

View Forum Message <> Reply to Message

Gianguido Cianci writes:

> Thanks Jeremy, your code generates the colors much faster indeed, but unfortunately the bottleneck is cgPlotS...
>
> I am wondering if I could batch cgPlotS all points that have the same colour to speed things up...

Yes, I have run into occasions, mostly in very tight loops,
where the Coyote Graphics routines can be almost as slow
as the equivalent function graphics routines. If you look
at the code, you can see why: there is a lot of overhead
getting the colors right, the color model set up, etc.

Fortunately, there is usually a way around this. These

routines are, after all, simply wrappers to the normal
low-level IDL routines. All you really need to do to
speed everything up is put yourself in a 24-bit decomposed
color environment and use the low-level graphics routines
to do whatever it is you want to do. This will cut out
almost all of the overhead and will be wickedly fast.

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Thu, 26 May 2011 22:21:50 GMT

View Forum Message <> Reply to Message

David Fanning writes:

> Yes, I have run into occasions, mostly in very tight loops,
> where the Coyote Graphics routines can be almost as slow
> as the equivalent function graphics routines. If you look
> at the code, you can see why: there is a lot of overhead
> getting the colors right, the color model set up, etc.
>
> Fortunately, there is usually a way around this. These
> routines are, after all, simply wrappers to the normal
> low-level IDL routines. All you really need to do to
> speed everything up is put yourself in a 24-bit decomposed
> color environment and use the low-level graphics routines
> to do whatever it is you want to do. This will cut out
> almost all of the overhead and will be wickedly fast.

Another alternative, of course, is to write cgPlotS as
an object (Coyote Graphics 2.0). Then you only incur
the overhead once. I've demonstrated how to do this by
writing the plot command as a object. Any takers for
building cgsPlotS?

  http://www.idlcoyote.com/programs/experimental

It's possible, if someone would take this on, that we

could have Coyote Graphics 2.0 finished by the time
I get back from my travels this summer. And I wouldn't
have had to write anything. :-)

Cheers,

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by cgguido on Thu, 26 May 2011 23:02:31 GMT

View Forum Message <> Reply to Message

Given what I just saw on my screen (speedup by x10) without understanding it, I don't think I should NOT pollute the cgWorld! :-(

All I did (on a hunch...) is replace the line with

```
cgplot, /over ... (fast!)
```

for

```
cgplotS, ...  (slowwww!)
```

I am guessing the advantage is that, "cgplotS" (whether I pass COLOR or SYMCOLOR) has to loop over every dot, but "cgplot, /over" does not?, not in the same way...?
I suppose I could further speed things up by replacing the looped WHERE with a histogram... still, I can now pretty plot 100k dots in 4s (with Z buffer, 6s with X) rather than 80-90s!

Here is the code as it stands now:

```
h2=sshist_2d(x,y, re=ri1, cost=co,  outbin = bin)
xmin = min(X) &  ymin = min(y)
h2size = size(h2, /dimen)
col = h2[ floor((x-xmin)/bin[0]) + floor((y-ymin)/bin[1])*h2size[0] ]
cgloadct, ctable

cgplot, x, y, /noda, back=cgcolor('black'), $
color=cgcolor('white'), chars=1.5,  _extra = eee

col = bytscl(col)
cmin = min(fix(col),  max = cmax)
for c=cmax, cmin, -1 do begin
```

```
   w=where(col EQ c)
   if w[0] ne -1 THEN $
   cgplot, x[w], y[w], ps=16, syms=.1, col=c,  /ov
endfor
```

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Thu, 26 May 2011 23:07:19 GMT

View Forum Message <> Reply to Message

Gianguido Cianci writes:

> Given what I just saw on my screen (speedup by x10) without understanding it, I don't think I should NOT pollute the cgWorld! :-(
>
> All I did (on a hunch...) is replace the line with
>
> cgplot, /over ... (fast!)
>
> for
>
> cgplotS, ... (slowwwww!)

Oh, well, you could do that, too. ;-)

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Thu, 26 May 2011 23:11:44 GMT

View Forum Message <> Reply to Message

Gianguido Cianci writes:

> I am guessing the advantage is that, "cgplotS" (whether I pass COLOR or SYMCOLOR) has to loop over every dot, but "cgplot, /over" does not?, not in the same way...?
> I suppose I could further speed things up by replacing the looped WHERE with a histogram...

still, I can now pretty plot 100k dots in 4s (with Z buffer, 6s with X) rather than 80-90s!

Gianguido, would you like to send me the data you
are using? This might make a very nice article,
and I would love to know more about that
automatic bin sizing program! :-)

Don't reply to this posting, but I can be found in
the usual place. :-)

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by almost_like_a_metapho on Fri, 27 May 2011 16:54:07 GMT
View Forum Message <> Reply to Message

My sincere thanks for all the helpful replies!

I've tried a PLOTS solution - my issue right now, is that I'm loading
multiple data sets in sequence and using the /CONTINUE argument,
which, when I get beyond a few tends of thousand points seems to draw
IDL to a crawl. I'm working currently on a LIST or HASH solution to my
data to avoid that.

THe CGplot solutoion also works quite well, but slow. I currently have
~400,000 points to plot, and will end up with probably an order of
magnitude more before this is done with.

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Fri, 27 May 2011 17:06:39 GMT
View Forum Message <> Reply to Message

almost_like_a_metaphor writes:

> THe CGplot solutoion also works quite well, but slow. I currently have

> ~400,000 points to plot, and will end up with probably an order of
> magnitude more before this is done with.

I hate to be a spoil sport, but what is the point
of putting 4 million points on a plot!? Don't some
of them, uh, overlap? Think "visualization" rather
than "By God I have the data and I'm gonna plot it!".

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by cgguido on Fri, 27 May 2011 17:08:47 GMT

View Forum Message <> Reply to Message

I think in your case (sorry again for the hijack), you would benefit from histogramming your z-values in 256 bins, and then plotting each set in one go. Assuming you are happy with 256 colours...

G

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by Paul Van Delst[1] on Fri, 27 May 2011 18:14:09 GMT

View Forum Message <> Reply to Message

David Fanning wrote:
> almost_like_a_metaphor writes:
>
>> THe CGplot solutoion also works quite well, but slow. I currently have
>> ~400,000 points to plot, and will end up with probably an order of
>> magnitude more before this is done with.
>
> I hate to be a spoil sport, but what is the point
> of putting 4 million points on a plot!? Don't some
> of them, uh, overlap? Think "visualization" rather
> than "By God I have the data and I'm gonna plot it!".

Dunno about the OP, but plotting lots and lots of points (i.e. scatter plot) can tell you a lot about the relationships
in, and between, datasets. Especially if datasets derived using different algorithms/input-data/whatever are
scatter-plotted with different colours. (a meaningless scatterplot scenario: red points show a linear dependency with a
negative bias, the blue quadratic with a positive bias, and the green linear/+ve bias for low wind speeds, but inverted
quadratic for higher windspeeds)

I could also see plotting individual points using a color gradient to include, say, time information in said
scatter-y-type plot.

It wouldn't be the only way I would look at a dataset, but it is still a useful visualisation of the data.

cheers,

paulv

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Fri, 27 May 2011 18:28:43 GMT

View Forum Message <> Reply to Message

Paul van Delst writes:

> Dunno about the OP, but plotting lots and lots of points (i.e. scatter plot) can tell you a lot about the relationships
> in, and between, datasets. Especially if datasets derived using different algorithms/input-data/whatever are
> scatter-plotted with different colours. (a meaningless scatterplot scenario: red points show a linear dependency with a
> negative bias, the blue quadratic with a positive bias, and the green linear/+ve bias for low wind speeds, but inverted
> quadratic for higher windspeeds)
>
> I could also see plotting individual points using a color gradient to include, say, time information in said
> scatter-y-type plot.
>
> It wouldn't be the only way I would look at a dataset, but it is still a useful visualisation of the data.

I don't have any problem with scatterplots. I'm
just saying that you can't realistically "see"
4 million points on a line plot unless your

monitor is the size of, say, the Vietnam
Memorial wall!

I wonder how your visualization would differ
if you randomly selected one percent of those
points and plotted those. I would guess the
plot would not look materially different,
although the rendering speed might improve
dramatically. :-)

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Fri, 27 May 2011 18:45:58 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> I don't have any problem with scatterplots. I'm
> just saying that you can't realistically "see"
> 4 million points on a line plot unless your
> monitor is the size of, say, the Vietnam
> Memorial wall!
>
> I wonder how your visualization would differ
> if you randomly selected one percent of those
> points and plotted those. I would guess the
> plot would not look materially different,
> although the rendering speed might improve
> dramatically. :-)

Maybe you can tell this is one of my pet peeves. :-)

The people who want to display 4 million points on
a line plot are the same people who think a line
plot *is* their data. I would encourage them to
read and understand the central principle behind

Oliver Sack's essay The Man Who Mistook His Wife
For His Hat. We *visualize* or *represent* our
data to learn something about it. I would encourage
anyone who wants to plot 4 million points to use
a 3D printer to visualize it. Then maybe it would
actually mean something.

http://en.wikipedia.org/wiki/3D_printing

For a cheap 3D printer, consider the one mentioned
in this article:

http://www.nytimes.com/2010/09/14/technology/14print.html

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Fri, 27 May 2011 18:55:52 GMT

View Forum Message <> Reply to Message

David Fanning writes:

> For a cheap 3D printer, consider the one mentioned
> in this article:
>
> http://www.nytimes.com/2010/09/14/technology/14print.html

Of, of course, if you are too cheap to buy a 3D printer,
you could always use Gianguido's suggestion of a 2D
histogram (the poor man's 3D printer). That sounds like
a winner to me. :-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by Kenneth P. Bowman on Fri, 27 May 2011 22:12:54 GMT

View Forum Message <> Reply to Message

In article <MPG.28499f5233c19a5b9898e0@news.giganews.com>,
David Fanning <news@idlcoyote.com> wrote:

> The people who want to display 4 million points on
> a line plot are the same people who think a line
> plot *is* their data.

Plotting 4M points is not necessarily a dumb thing. A 1000 x 1000 pixel window (quite reasonable on current displays) is 1M pixels. So plotting 4M points will result in some overlap, but it might also reveal patterns in data that are difficult to see with binned data. In some cases, binning can cause its own set of perceptual problems. In my experience, contour plots are much more likely to fool the viewer than scatter plots.

You do always need to have an awareness of what your data is and what a particular visualization is showing you (or hiding from you).

Cheers, Ken

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Fri, 27 May 2011 23:51:35 GMT

View Forum Message <> Reply to Message

Kenneth P. Bowman writes:

> Plotting 4M points is not necessarily a dumb thing. A
> 1000 x 1000 pixel window (quite reasonable on current
> displays) is 1M pixels. So plotting 4M points will
> result in some overlap, but it might also reveal
> patterns in data that are difficult to see with
> binned data

I'd say if you data was spread out evenly on
a 1000x1000 grid, you would be better off

forgetting about the plot and going to get
a beer. :-)

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by Kenneth P. Bowman on Sat, 28 May 2011 13:30:17 GMT
View Forum Message <> Reply to Message

In article <MPG.2849e6f29887ef3c9898e2@news.giganews.com>,
David Fanning <news@idlcoyote.com> wrote:

> I'd say if you data was spread out evenly on
> a 1000x1000 grid, you would be better off
> forgetting about the plot and going to get
> a beer. :-)

I couldn't drink a beer that fast even in my college days.

```
PRO TEST_SCATTER
;  Plot a scatterplot with a lot of points
t0 = SYSTIME(/SECONDS)
n = 4000000
x = RANDOMU(seed, n)
y = SIN(2.0*!PI*x) + 0.3*RANDOMN(seed, n)
WINDOW, XSIZE = 1000, YSIZE = 1000
PLOT, x, y, PSYM = 3
PRINT, 'Elapsed time : ', SYSTIME(/SECONDS) - t0
END
```

```
IDL> .r test_scatter
% Compiled module: TEST_SCATTER.
IDL> test_scatter
Elapsed time :       5.5195148
```

After you making this plot, you might want to 2-D bin the data and
replot it, or you might want to do some other analysis entirely.
I think that this is a quick and easy way to get
an idea of what your data looks like, but I should know
better than to expect to get the last word.

Ken

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Sat, 28 May 2011 15:21:27 GMT

View Forum Message <> Reply to Message

Kenneth P. Bowman writes:

```
>  I couldn't drink a beer that fast even in my college days.
>
>  PRO TEST_SCATTER
>  ;  Plot a scatterplot with a lot of points
>  t0 = SYSTIME(/SECONDS)
>  n = 4000000
>  x = RANDOMU(seed, n)
>  y = SIN(2.0*!PI*x) + 0.3*RANDOMN(seed, n)
>  WINDOW, XSIZE = 1000, YSIZE = 1000
>  PLOT, x, y, PSYM = 3
>  PRINT, 'Elapsed time : ', SYSTIME(/SECONDS) - t0
>  END
>
>
>  IDL> .r test_scatter
>  % Compiled module: TEST_SCATTER.
>  IDL> test_scatter
>  Elapsed time :       5.5195148
```

I guess my machine is a LOT slower. Which confuses me,
because I spent good money on this darn machine! :-(

Anyway, when I run your program, it takes about 42
seconds. After three sets of tennis, I have been
known to drink a beer in about that amount of time!

```
IDL> test_scatter
% Compiled module: TEST_SCATTER.
Elapsed time :       42.018000
```

But this sort of proves my point. If I run your program
with 1 percent of the points, the "visualization" doesn't
change in any material way, but the time is reduced by

a factor of 1000.

```
PRO TEST_SCATTER
;  Plot a scatterplot with a lot of points
n = 4000000L
x = RANDOMU(seed, n)
y = SIN(2.0*!PI*x) + 0.3*RANDOMN(seed, n)
indices = Round(randomu(seed,40000L)*4000000L)
WINDOW, XSIZE = 1000, YSIZE = 1000, 1
t0 = SYSTIME(/SECONDS)
PLOT, x[indices], y[indices], PSYM = 3
PRINT, 'Elapsed time : ', SYSTIME(/SECONDS) - t0
END
```

```
IDL> test_scatter
% Compiled module: TEST_SCATTER.
Elapsed time :       0.43099999
```

> I think that this is a quick and easy way to get
> an idea of what your data looks like, but I should know
> better than to expect to get the last word.

Gianguido was pointing out to me yesterday that the top three contributors to the IDL newsgroup for all time are:

davidf@dfanning.com
david@dfanning.com
news@dfanning.com

You don't get these kinds of records by letting someone else have the last word! ;-)

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems

Posted by David Fanning on Sat, 28 May 2011 15:50:19 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> But this sort of proves my point. If I run your program
> with 1 percent of the points, the "visualization" doesn't
> change in any material way, but the time is reduced by
> a factor of 1000.

Sorry. Factor of 100. While I was writing this I was
momentarily distracted by both a Lazuli Bunting and
a Western Tanager showing up at the backyard feeder
at the same time! Two rare and beautiful birds on the
same day is unbelievable, but two on the same feeder
is a miracle!

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by penteado on Sat, 28 May 2011 19:29:06 GMT
View Forum Message <> Reply to Message

On May 28, 12:21 pm, David Fanning <n...@idlcoyote.com> wrote:
> But this sort of proves my point. If I run your program
> with 1 percent of the points, the "visualization" doesn't
> change in any material way, but the time is reduced by
> a factor of 1000.

It does not change in that case, but it can easily not be the case. I
have one particular application where I can have millions of points to
plot, and the visualization would change substantially if I took a
random subsample.

All it takes is for the distribution of points to be very non-uniform
along it. Then the random subsample might (in some cases probably
would) miss those few points that have very different characteristics

(because, say, nearly all points fall in the same region, with a lot of overlap, but only one in a 1000 will fall in a distinct region in the plot). A common situation, for instance, when one works with the spatial distribution of observations, where some regions, due to geometry / instrument constraints, are only observed rarely.

The plot may have a lot of overlapping points, but still be interesting. As long as the overlapping points do not cover everything, there is room to have the different (frequently the most interesting) points falling in other regions. And this may not show well in 2D histograms, which may not resolve well those few odd points. That is the reason why in some visualizations I used both a scatterplot and a 2D histogram: the histogram shows the distribution well where there is a lot of overlap, while the scatterplot shows well the uncommon points.

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by David Fanning on Sat, 28 May 2011 21:45:41 GMT

View Forum Message <> Reply to Message

Paulo Penteado writes:

> It does not change in that case, but it can easily not be the case. I
> have one particular application where I can have millions of points to
> plot, and the visualization would change substantially if I took a
> random subsample.

Alright, I'll give you the last word on the subject. ;-)

Cheers,

David

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

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by Carsten Lechte on Tue, 31 May 2011 08:46:58 GMT

View Forum Message <> Reply to Message

On 28/05/11 21:29, Paulo Penteado wrote:
> It does not change in that case, but it can easily not be the case. I
> have one particular application where I can have millions of points to
> plot, and the visualization would change substantially if I took a
> random subsample.

Long ago, when computers were much slower, I had the problem that scatter plots with millions of points would produce huge ps files and take forever to render on screen or to print. I applied poor man's binning by doing the scatter plot into a bitmap graphic. That way, I had all the points, but there was an upper bound to the size of the plot and the resources it took to render it.

chl

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by Kenneth P. Bowman on Tue, 31 May 2011 14:36:42 GMT

View Forum Message <> Reply to Message

In article <MPG.284ac0e2722642419898e3@news.giganews.com>,
 David Fanning <news@idlcoyote.com> wrote:

```
> Kenneth P. Bowman writes:
>
>> IDL> .r test_scatter
>> % Compiled module: TEST_SCATTER.
>> IDL> test_scatter
>> Elapsed time :       5.5195148
>
> I guess my machine is a LOT slower. Which confuses me,
> because I spent good money on this darn machine! :-(
```

This is running on a Mac laptop from a couple of years ago.
(Sorry, I don't mean to embarrass your computer.)

> But this sort of proves my point. If I run your program
> with 1 percent of the points, the "visualization" doesn't
> change in any material way, but the time is reduced by
> a factor of 1000.

Actually, I think it proves *my* point. You plot all of the data. You see that there is a lot of overlap. You decimate the data and plot it again. If the results are qualitatively the same, you can continue with the decimated data, but you don't want to *start* by decimating the data. You might miss something important (like outliers).

Cheers, Ken

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by pgrigis on Tue, 31 May 2011 16:28:15 GMT

View Forum Message <> Reply to Message

Here's an example where mindlessly plotting
points may lead to wrong conclusions:

```
x=randomn(seed,4*10.0^6)
y=randomn(seed,4*10.0^6)
plot,x,y,xrange=[-8,8],yrange=[-8,8],/iso,psym=3,/xst,/yst
```

The plot seems to indicate that the distribution
of the points within 3 units from (0,0) is uniform,
which is not the case as the points are drawn from
a normal distribution - this is just an artifact from
the overlap of the points.

Ciao,
Paolo

On May 28, 11:50 am, David Fanning <n...@idlcoyote.com> wrote:
> David Fanning writes:
>> But this sort of proves my point. If I run your program
>> with 1 percent of the points, the "visualization" doesn't
>> change in any material way, but the time is reduced by
>> a factor of 1000.
>
> Sorry. Factor of 100. While I was writing this I was
> momentarily distracted by both a Lazuli Bunting and
> a Western Tanager showing up at the backyard feeder
> at the same time! Two rare and beautiful birds on the
> same day is unbelievable, but two on the same feeder
> is a miracle!
>
> Cheers,
>
> David
>
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming:http://www.idlcoyote.com/
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Frustrated by 2 Data Plotting problems
Posted by almost_like_a_metapho on Tue, 31 May 2011 20:30:07 GMT
View Forum Message <> Reply to Message

On May 27, 2:28 pm, David Fanning <n...@idlcoyote.com> wrote:
> I don't have any problem with scatterplots. I'm
> just saying that you can't realistically "see"
> 4 million points on a line plot unless your
> monitor is the size of, say, the Vietnam
> Memorial wall!

Well, a cinema display would go most of the way...

I'll grant you that some of this data does indeed overlap, and it is indeed at least part of the point to identify some of the overlapping locations. Additionally, I do intend to slice the data into smaller sections and mask it in various ways.

This is meant to be a quick and somewhat dirty display of a large data set. the worry about taking a percentage, or an 'every x points' is that some variability that I want to identify makes x occasionally equal to 1. 1% would also significantly under-represent my data. Though I'm going to experiment with 1/2 and 1/4. It really turns out that the limiting factor in this particular problem is file i/o as opposed to rendering speed.

On the other hand, I have finer displays in mind for data subsets.

N