Subject: xerr
Posted by laxsri on Wed, 17 Dec 2008 20:27:59 GMT
View Forum Message <> Reply to Message

Hi,

I've been using mpfitfun to fit measured values of period (y) and
distances (x) in a linear equation y = a + bx.
I would like to know if we can include the measured uncertainties in x
values too?

Thanks,

Lakshmi

Subject: Re: xerr
Posted by Wout De Nolf on Thu, 18 Dec 2008 10:35:42 GMT
View Forum Message <> Reply to Message

On Wed, 17 Dec 2008 12:27:59 -0800 (PST), lakshmi <laxsri@gmail.com>
wrote:

> I've been using mpfitfun to fit measured values of period (y) and
> distances (x) in a linear equation y = a + bx.
> I would like to know if we can include the measured uncertainties in x
> values too?

What you're looking for is called "Total least squares" or "orthogonal
regression". Here is a reference + code you can use, translate to IDL
and hopefully share it with use :-).

Article:
http://www.iop.org/EJ/abstract/0957-0233/18/11/025
Matlab (off course) code:
http://www.mathworks.com/matlabcentral/fileexchange/17466

It uses the method of the Lagrange Multipliers, so it should be
possible to add constraints to the rico and intercept.

If you're lazy: here's a simple code (not tested and without
calculating errors) using info from
http://mathforum.org/library/drmath/view/63765.html. Change
fixintercept to 1 when you want to fix the intercept.

```
pro odr
n=100
x=findgen(n)+RANDOMN(seed,n)
```

```
rico=1.2
intercept=3
fixintercept=0b
y=rico*x+intercept+RANDOMN(seed,n)

print,'Rico:',rico
print,'Intercept:',intercept

; Centroid: orthogonal distance
; regression line goes through it
n=n_elements(x)
data=transpose([[x],[y]])
centroid=total(data,2)/n

; Optional: Fix intercept
if fixintercept then centroid=[0,intercept]

data[0,*]-=centroid[0]
data[1,*]-=centroid[1]

SVDC, data, W, U, V

smallest_singularvalue=min(W,ind)
normal=reform(V[ind,*])

rico=-normal[0]/normal[1]
intercept=-rico*centroid[0]+centroid[1]

print,'ODR...'
print,'Rico:',rico
print,'Intercept:',intercept

window
plot,x,y,psym=1
oplot,x,rico*x+intercept
end
```

Subject: Re: xerr
Posted by laxsri on Thu, 18 Dec 2008 19:51:12 GMT

View Forum Message <> Reply to Message

Thanks all!

I shall look up the orthogonal regression Wox...I guess, I thought
this would have been implemented in IDL by somebody! (yea...lazy too)!

Good day

Lakshmi

On Dec 18, 9:35 pm, Wox <s...@nomail.com> wrote:

```
>  On Wed, 17 Dec 2008 12:27:59 -0800 (PST), lakshmi <lax...@gmail.com>
>  wrote:
>
>> I've been using mpfitfun to fit measured values of period (y) and
>> distances (x) in a linear equation y = a  + bx.
>> I would like to know if we can include the measured uncertainties in x
>> values too?
>
>  What you're looking for is called "Total least squares" or "orthogonal
>  regression". Here is a reference + code you can use, translate to IDL
>  and hopefully share it with use :-).
>
>  Article:http://www.iop.org/EJ/abstract/0957-0233/18/11/025
>  Matlab (off course)  code:http://www.mathworks.com/matlabcentral/fileexchange/174 66
>
>  It uses the method of the Lagrange Multipliers, so it should be
>  possible to add constraints to the rico and intercept.
>
>  If you're lazy: here's a simple code (not tested and without
>  calculating errors) using info fromhttp://mathforum.org/library/drmath/view/63765.html. Change
>  fixintercept to 1 when you want to fix the intercept.
>
>  pro odr
>  n=100
>  x=findgen(n)+RANDOMN(seed,n)
>  rico=1.2
>  intercept=3
>  fixintercept=0b
>  y=rico*x+intercept+RANDOMN(seed,n)
>
>  print,'Rico:',rico
>  print,'Intercept:',intercept
>
>  ; Centroid: orthogonal distance
>  ; regression line goes through it
>  n=n_elements(x)
>  data=transpose([[x],[y]])
>  centroid=total(data,2)/n
>
>  ; Optional: Fix intercept
>  if fixintercept then centroid=[0,intercept]
>
>  data[0,*]-=centroid[0]
>  data[1,*]-=centroid[1]
```

```
>
> SVDC, data, W, U, V
>
> smallest_singularvalue=min(W,ind)
> normal=reform(V[ind,*])
>
> rico=-normal[0]/normal[1]
> intercept=-rico*centroid[0]+centroid[1]
>
> print,'ODR...'
> print,'Rico:',rico
> print,'Intercept:',intercept
>
> window
> plot,x,y,psym=1
> oplot,x,rico*x+intercept
> end
```

Subject: Re: xerr
Posted by ed.schmahl on Fri, 19 Dec 2008 05:29:33 GMT
View Forum Message <> Reply to Message

On Dec 18, 8:24 pm, Craig Markwardt <cbmarkwa...@gmail.com> wrote:
> On Dec 17, 5:08 pm, Paolo <pgri...@gmail.com> wrote:
>
>> This is discussed for example in
>> section 15.3 in edition 3 of the book
>> "numerical recipes".
>
> I've used the Numerical Recipes hack for X errors successfully before.
>
> As mentioned, orthogonal distance regression is the real way to do
> this, but unfortunately MPFIT does not support this. [It could in
> principle with a lot of work, but doesn't in practice.]
>
> Craig

Subject: Re: xerr
Posted by ed.schmahl on Fri, 19 Dec 2008 05:43:59 GMT
View Forum Message <> Reply to Message

On Dec 18, 8:24 pm, Craig Markwardt <cbmarkwa...@gmail.com> wrote:
> On Dec 17, 5:08 pm, Paolo <pgri...@gmail.com> wrote:
>
>> This is discussed for example in

>> section 15.3 in edition 3 of the book

Orthogonal least squares, as in finding the line with minimum squared distance from a set of points (x,y), where x and y are on an equal footing, is an eigenvalue problem, and therefore not within the province of MPFIT. Just try googling "least square distance" to find oodles of info about this problem.

However, a working IDL program that finds the orthogonal solution a*x +b*y=d, where x and y are on an equal footing, may be found at http://hesperia.gsfc.nasa.gov/~schmahl/pro/lst_sq_dist_line. pro.

The least square plane a*x+b*y+c*z=d is just as readily found using a similar program: http://hesperia.gsfc.nasa.gov/~schmahl/pro/lst_sq_plane.pro

Both of these programs were converted to IDL from a Fortran program so old its origin is lost in the mists of time.

In each case (2D or 3D), the eigenvector with minimum eigenvalue found by this program is perpendicular to the line (or plane) and the eigenvalue is the sum of the squares of the distances of the points from the line.

Adding a subroutine that computes the sigmas for x and y is an exercise for the reader.

Ed Schmahl
CoRA, Boulder, CO

> "numerical recipes".
>
> I've used the Numerical Recipes hack for X errors successfully before.
>
> As mentioned, orthogonal distance regression is the real way to do
> this, but unfortunately MPFIT does not support this. [It could in
> principle with a lot of work, but doesn't in practice.]
>
> Craig