Subject: Re: How to do surface fit?
Posted by Gang Chen on Wed, 28 Jan 1998 08:00:00 GMT
View Forum Message <> Reply to Message

> Use the KX keyword to SFIT, it will return an array of coefficients, in
> your case:
>
> IDL> zfit = SFIT(z,2,KX=k)>
> IDL> help,k
> COEFF FLOAT = Array[3, 3]
>
> They are calculated with the convention X= 0,...,NX and Y=0,...,NY
> (so you must to make the needed conversion).

Hi Evilio,

I really appreciate your help on this. It seems that it almost works the way Iwanted. But I have another question: Since the default grids are set withX= 0,...,NX and Y=0,...,NY in the source code for SFIT, how can I do for anon-uniform grids (e.g., X=0, 1, 4, 5, 9, ..., and Y=3, 4, 7, 11, ..)? Do I have to resort to interpolation so that values at uniform grids can be distributed?

Many thanks,
Gang Chen == gang@cochise.biosci.arizona.edu

Subject: Re: How to do surface fit?
Posted by Evilio del Rio on Wed, 28 Jan 1998 08:00:00 GMT
View Forum Message <> Reply to Message

On Tue, 27 Jan 1998, Gang Chen wrote:

> Hi,
>
> I am trying to fit a surface (probably 2nd degree polynomial:
> z=a+b*x+c*y+d*x*x+e*xy+f*y*y) on a two dimensional grids with IDL. But it
> seems to me that SFIT and SVDFIT would not serve my purpose: SFIT only
> generates the values on those grids without giving the coefficients, and
> it uses different format of polynomial (z=sum of k_i,j * x^i * y^j); While
> SVDFIT works only for curve fit. Does anybody have any idea on how to do
> this by simply calling some existing functions instead of writing more
> complicate code?
>
Use the KX keyword to SFIT, it will return an array of coefficients, in
your case:

```
IDL> zfit = SFIT(z,2,KX=k)
IDL> help,k
COEFF           FLOAT     = Array[3, 3]
```

They are calculated with the convention X= 0,...,NX and Y=0,...,NY
(so you must to make the needed conversion). Also it seems that they are
reversed and the degree of X increases with the 2nd dimension on k and
the degree of Y with the first (so the actual matrix should be
TRANSPOSE(k), or so I see it):

```
IDL> help,x,y,z
X               DOUBLE    = Array[5, 6]
Y               DOUBLE    = Array[5, 6]
Z               DOUBLE    = Array[5, 6]
IDL> print,x,F='(5(F4.1))'
 0.0 1.0 2.0 3.0 4.0
 0.0 1.0 2.0 3.0 4.0
(...)
IDL> print,y,F='(5(F4.1))'
 0.0 0.0 0.0 0.0 0.0
 1.0 1.0 1.0 1.0 1.0
(...)
 5.0 5.0 5.0 5.0 5.0
IDL> z = x + 3.0*y*y + 5.0*x*y
IDL> zfit = SFIT(z,2L,KX=k)
IDL> help,k
K               FLOAT     = Array[3, 3]
IDL> print,k,F='(3(F4.1))'
 0.0 0.0 3.0
 1.0 5.0 0.0
 0.0 0.0 0.0
IDL> print,k[0,1] ; This should be coefficient for the term x^0*y^1 => 0.0
    ; in our case
  1.0
IDL> print,k[1,0]
  0.0
```

> Many thanks,
> Gang Chen == gang@cochise.biosci.arizona.edu
>

Cheers,

Evilio Jose del Rio Silvan Institut de Ciencies del Mar
E-mail: edelrio@icm.csic.es URL: http://www.ieec.fcr.es/~evilio/
"Anywhere you choose,/ Anyway, you're gonna lose"- Mike Oldfield