Subject: Re: multi-dimensional SVDFIT
Posted by R.G.S. on Tue, 03 Jul 2001 21:17:21 GMT

Ingo Wardinski <ingo@gfz-potsdam.de> wrote in message
news:ynlelryc66q.fsf@mt35.gfz-potsdam.de...
> Hi,
> does anyone know, if it is possible to make a multi-dimensional SDVFIT
> to a certain dataset? I'm looking for a SDVFIT of y=f(x1,x2), but
> AFAIK the standard idl SDVFIT routine works only for 1-d.
> thanks in advance
> ingo
> --
> ingo wardinski   ingo@gfz-potsdam.de
> GeoForschungsZentrum Potsdam, Telegrafenberg F456, 14473 Potsdam
>
> "Ich war ganz allein in der Maschine und habe dem Piloten gesagt..."
>   (Helmut Kohl in "Am Ende des Jahrhunderts")

Hello,
the SVD solution to a least squares fit (Ax=b) to higher dimensions is
trivial.
Merely add on the other dimensions as columns of A, since there is no
difference
in principal between "x^2" and "xy", since to the SVD, the matrix A is
merely a collection
of numbers.

Cheers,
bob stockwell
stocwkell at co-ra dot com

For instance, here is a smallpiece of code that will compute a fit to
$f(x,y,z) = a0+a1*x+a2*y+a3*z+a4x*y+a5x*z+a6y*z+a7= x^2+a8y^2+a9z^2$

(i.e. 10 parameters second order term polynomial in 3D)


```
; Here zon = data(x,y,z)
m = 10 ; number of terms in equation
n = n_elements(zon) ; n = number of data points
```

```
; create matrix colum by colum
a = dblarr(m,n)
a(0,*) = 1
a(1,*) = x
a(2,*) = y
a(3,*) = z
a(4,*) = x*y
a(5,*) = x*z
a(6,*) = y*z
a(7,*) = x^2
a(8,*) = y^2
a(9,*) = z^2




; Decompose A:
tic = systime(1)
SVDC, A, W, U, V,/double
toc = systime(1)

; Compute the solution and print the result:
result1 = SVSOL(U, W, V, zon,/double)
toc2 = systime(1)
print,result1
```

Subject: Re: multi-dimensional SVDFIT
Posted by ingo on Wed, 04 Jul 2001 11:57:47 GMT
View Forum Message <> Reply to Message

> " " == R G S <rgs1967@hotmail.com> writes:
 > Hello,
 > the SVD solution to a least squares fit (Ax=b) to higher dimensions is
 > trivial.
 > Merely add on the other dimensions as columns of A, since there is no
 > difference
 > in principal between "x^2" and "xy", since to the SVD, the matrix A is
 > merely a collection
 > of numbers.

 > Cheers,
 > bob stockwell
 > stocwkell at co-ra dot com

> For instance, here is a smallpiece of code that will compute a fit to

```
> f(x,y,z)  = a0+a1*x+a2*y+a3*z+a4x*y+a5x*z+a6y*z+a7= x^2+a8y^2+a9z^2

> (i.e. 10 parameters second order term polynomial  in 3D)


> ; Here zon = data(x,y,z)
> m = 10 ; number of terms in equation
> n = n_elements(zon) ; n = number of data points


> ; create matrix colum by colum
> a = dblarr(m,n)
> a(0,*) = 1
> a(1,*) = x
> a(2,*) = y
> a(3,*) = z
> a(4,*) = x*y
> a(5,*) = x*z
> a(6,*) = y*z
> a(7,*) = x^2
> a(8,*) = y^2
> a(9,*) = z^2


> ; Decompose A:
> tic = systime(1)
> SVDC, A, W, U, V,/double
> toc = systime(1)

> ; Compute the solution and print the result:
> result1 = SVSOL(U, W, V, zon,/double)
> toc2 = systime(1)
> print,result1
```

Hi Bob,
thanx for your response, but unfortunately it doesn't work with
SVDFIT.
The error message looks like 'SVDFIT: The input X must be a vector'
SVDFIT does also not accept X as a structur.
here my code:

;declaring x1,x2

x=fltarr(2,length)
x(0,*)=t

```
x(1,*)=aa_ind

;first guess
A=[1,1,1,1,1,1,1,1,1] ; m=N_ELEMENTS(A)
;ts == my data
result=svdfit(x,ts,A=A, MEASURE_ERRORS=0.05*ts, $
         FUNCTION_NAME='myfunct2',SIGMA=sigma,YFIT=YFIT)

window,0,retain=2
!P.MULTI=[0,1,2]
PLOT,YFIT
PLOT,ts

FUNCTION myfunct2,X,M

  RETURN,[[X(1,*)],$
      [SIN(2*!pi*X(0,*))/365.25],[COS(2*!pi*X(0,*))/365.25],$
      [SIN(2*!pi*X(0,*))/183.0], [COS(2*!pi*X(0,*))/183.0],$
      [1.0],[X(0,*)],[X(0,*)^2],[X(0,*)^3]]
END
```

It seems to me that i have to use svdc and svsol. (??)
thanx, ingo

--
ingo wardinski   ingo@gfz-potsdam.de
 GeoForschungsZentrum Potsdam, Telegrafenberg F456, 14473 Potsdam

There was a young lady named Bright,
Whose speed was far faster than light;
She set out one day,
In a relative way,
And returned home at the previous night.   Arthur Buller