Identificación estática 1 salida, 2 entradas

Una presentación/screencast de los contenidos de este archivo aparece en: http://personales.upv.es/asala/videos/id2.html

Vamos a identificar un modelo y=Cu, con y vector 1x1, C **matriz 1x2**, u vector 2x1.

```
C=[2 -1]; %este vector es "desconocido", queremos "adivinarlo"
```

Conocemos, u, y... varias muestras de ellos.

```
entradasexperim1=[ 1 -1 4 -5 6 7 9 -3 -8; ...
           2  4 -3 3 1 0 5 2 3]'
```

```
entradasexperim1 =
     1     2
    -1     4
     4    -3
    -5     3
     6     1
     7     0
     9     5
    -3     2
    -8     3
```

*Cada muestra está en una FILA de la matrix entradasexperim1...

```
info1=svd(entradasexperim1)
```

```
info1 =
   16.8021
    8.7572
```

```
entradas2=[1 1.1 1.05 0.1 -1;...
         2 2.1 2.15  0.2 -1.9]';
info2=svd(entradas2)
```

```
info2 =
    4.5827
    0.0587
```

```
[U,S,V]=svd(entradas2)
```

```
U =
   -0.4879    0.2692   -0.8087   -0.0580    0.1791
   -0.5173   -0.4760    0.2968   -0.0017    0.6463
   -0.5220    0.6688    0.5048   -0.0269   -0.1571
   -0.0488    0.0269   -0.0334    0.9979   -0.0013
    0.4685    0.5029    0.0445    0.0117    0.7249
S =
    4.5827         0
```

```
         0    0.0587
         0         0
         0         0
         0         0
V =
   -0.4535   -0.8912
   -0.8912    0.4535
```

Para el parámetro (en comp. principal, esto es, una cierta combinación de parámetros [c1 c2]) más difícil de identificar, tenemos que

```
min(info1)/min(info2)
```

```
ans = 149.0977
```

Este número es cómo de "más informativo" es el experimento "entradasexperim1" que "entradas2". **150 veces!**

```
%entradas=entradasexperim1; para otras pruebas
entradas=randn(40,2);
svd(entradas)
```

```
ans =
    7.2015
    6.2022
```

```
Nmuestras=size(entradas,1)
```

```
Nmuestras = 40
```

```
Nvariables=size(entradas,2)
```

```
Nvariables = 2
```

Generemos las salidas con una cierta matriz C, definida arriba.

Generamos salida "ruidosa": como en el modelo original y=Cu cada dato es columna, pero está en las matrices en "fila", entonces vamos a identificar C' en y'=u'*C'.

```
%rng(3241) si queremos comparar con la misma secuencia de núms. aleatorios.
salidalimpia=entradas*C';% inaccesible
size(salidalimpia)
```

```
ans =
    40     1
```

```
salidas= salidalimpia+randn(size(salidalimpia))*.25;
yy=salidas;
X=[entradas];
svd(X)
```

```
ans =
    7.2015
    6.2022
```

```
%X=[entradas entradas.^2 entradas.^3];
%svd(X)
```

```
ans =
   36.7585
   18.9292
   11.9105
    8.9673
    4.0002
    2.8923
```

```
th=pinv(X)*yy;
Cestim=th'
```

```
Cestim =
    2.0050   -1.0694    0.0173   -0.0069    0.0094    0.0059
```

Comprobemos el ajuste:

```
error=yy-X*th
```

```
error =
   -0.1830
    0.2447
    0.1252
   -0.0522
   -0.1657
    0.0282
   -0.1730
    0.2270
   -0.2439
    0.0962
      ⋮
```

Calculamos la varianza y desviación típica del error:

```
    Varerror=(error'*error)/(Nmuestras-2); %2 params por columna...
    std_error=sqrt(Varerror);
std_error
```

```
std_error = 0.2447
```

Ese **std_error** será una estimación del ruido de medida en la variable de salida... (si es verdad que ha sido generado con un ruido de medida aleatorio, claro... si no, incluye ruido sistemático --bias--).

Como el ruido de medida se multiplica por pinv(X), se multiplicará por la inversa de la información para producir parámetros estimados:

```
pi_svd=svd(pinv(X))
```

```
pi_svd =
    0.1775
    0.1463
```

La matriz de varianzas covarianzas de [c1 c2] es:

```
Var_th=Varerror(1)*inv(X'*X) %VC param estimados
```

```
Var_th =
    0.0017   -0.0003
   -0.0003    0.0015
```

Lo cual producirá, en cada uno de los errores una posible corrección si tuviéramos más datos, de varianza, para la medida "k", dada por:

$$\nu_k = varianza(X_k\theta) = E(X_k\theta\theta^T X_k^T) = X_k E(\theta\theta^T) X_k^T$$

y desv. típica dada por la raíz cuadrada. Por ejemplo, para la fila 7 tendríamos:

```
correccion_error7=sqrt(X(7,:)*Var_th*X(7,:)');
[error(7) correccion_error7*2]
```

```
ans =
   -0.1730    0.0149
```

Calculando la media de los cuadrados de esas correcciones de error para todos los errores:

```
desviacion_media_por_vzparametros=sqrt(trace(X*Var_th*X'/Nmuestras))
```

```
desviacion_media_por_vzparametros = 0.0547
```

```
%divido por Nmuestras porque es calculo teórico de vza
```

Eso tiene que ser la desviacion típica "en media" que podría añadirse, por la incertidumbre en θ, a la salida del modelo, al tomar más datos.

La desviación típica de cada parámetro es la raíz cuadrada de la diagonal de la matriz de varianzas-covarianzas:

```
VMat=[diag(Var_th)];
std_C=sqrt(VMat)'
```

```
std_C =
    0.0408    0.0388
```

Intervalos de confianza parámetros estimados

Tenemos un 99% de confianza que si el modelo ha sido originado por lineal+"ruido de medida dist normal", el proceso "real" esté entre:

```
fprintf('Valores mínimos 99%% [%g %g]',Cestim-3*std_C)
```

```
Valores mínimos 99% [1.84661 -1.09227]
```

```
fprintf('Valores máximos 99%% [%g %g]',Cestim+3*std_C)
```

```
Valores máximos 99% [2.09112 -0.859304]
```

Conclusiones

El experimento identifica un estimado de la matriz C

```
Cestim
```

```
Cestim =
    1.9689   -0.9758
```

Con una desviación típica en cada coeficiente dada por:

```
std_C
```

```
std_C =
    0.0408    0.0388
```

y una desviación típica del error de predicción en la variable de salida dada por:

```
std_error
```

```
std_error = 0.2447
```

Que podría reducirse con más datos hasta, aproximadamente, como mucho:

```
std_no_se_explicaconmasdatos=max(0,std_error-desviacion_media_por_vzparametros)
```

```
std_no_se_explicaconmasdatos = 0.1900
```