

Ajuste no lineal de datos con Matlab

Larotonda
Laboratorio 4 - 2oC 2014

DISTINTAS FORMAS DE MINIMIZACIÓN

La idea es aprender algún método de ajuste no lineal sobre matlab, que no sea tan “caja negra” como los de origen. Algoritmos de minimización hay muchos, algunos son lineales y otros dependen de derivadas parciales. Lo que proponemos es usar uno bastante robusto `fminsearch`, pero que no es capaz de dar intervalos de confianza de los parámetros de ajuste, y combinarlo con otro `nlinfit`, del que se puede obtener la incerteza estadística de los ajustes.

En este ejemplo los datos corresponden a dos picos de forma $\exp\left[-\left(\frac{x-x_1}{t}\right)^2\right]$. La tabla de datos “medidos” está en el archivo `dossgauss.mat`. Lo que hay que armar es una función de ajuste de la forma

$$f(x) = \exp\left[-\left(\frac{x-x_1}{x_2}\right)^2\right] + \exp\left[-\left(\frac{x-x_3}{x_4}\right)^2\right]. \quad (1)$$

Esta función ya está escrita, es `funcfitNpicos.m`. Para la primera parte, hace falta una función adicional, que es la norma de la diferencia entre los puntos medidos y la función de ajuste, evaluada en los mismos valores de la variable independiente (x). A esta la llamamos `estimador.m`; La rutina de Matlab `fminsearch` va a buscar minimizar esa diferencia. Para que la rutina funcione, hay que inicializar los parámetros $\{x_1, \dots, x_4\}$ de forma medianamente razonable como para que encuentre el mínimo.

Una vez que encontramos una solución aproximada (en general está bastante cerca), podemos pasar la solución del primer método como semilla para el segundo. El método es el de regresión no lineal de la rutina `nlinfit` (o una versión modificada para permitir fijar algunos de los parámetros, `mod_nlinfit`). Esto es bastante rápido, y cuando converge a una solución, devuelve los parámetros de ajuste y fundamentalmente, sus tolerancias o incertezas. Esto lo puede calcular usando la matriz de covarianza de los coeficientes ajustados. El programa de Matlab `ejemplo_ajuste.m` hace esto mismo: carga los datos del experimento simulado, ajusta primero con un método heurístico y después con el método de derivadas parciales. Para que funcione hay que bajar las funciones adicionales `estimador.m`, `funcfitNpicos.m` y los datos numéricos que están todos juntos en `datos_simulados.rar`. Todo este paquete de funciones, scripts y datos están en el archivo comprimido `ajuste_matlab.rar`.

Otras series de datos y sus funciones de ajuste

Biexponencial: Una superposición de dos decaimientos con distintos tiempos característicos:

$$f(x) = \frac{9}{10} \exp\left(-\frac{x}{t_1}\right) + \frac{1}{10} \exp\left(-\frac{x}{t_2}\right). \quad (2)$$

Los datos asociados están en `biexponencial.mat`.

Saturación exponencial: La curva típica de carga de un capacitor, con cierto *offset*:

$$f(x) = e_1 - \exp\left(-\frac{x}{e_2}\right) \quad (3)$$

Los datos simulados para ajustar a este modelo están en `capacitor.mat`. También se puede probar a ajustar la primera parte con una función lineal:

$$f(x) \approx e_1 - 1 + x/e_2 - O(x^2) \quad (4)$$

Función error, o saturación: Una expresión que puede describe un comportamiento lineal con saturación; la integral de un pulso gaussiano

$$f(x) = e_0 + \operatorname{erf}\left(\frac{x-e_1}{e_2}\right) \quad (5)$$

Para usar esta función como ajuste no lineal, usar los datos de `erf.mat`.

Decaimiento exponencial: A partir de los valores experimentales encontrar la amplitud inicial A y el tiempo característico t_c :

$$f(x) = A \exp(-x/t_c) \quad (6)$$

Para usar esta función como ajuste no lineal, usar los datos de `expdecay.mat`. Atención a cómo se presentan estos valores en un gráfico. Una representación lineal no siempre es la mejor.

Oscilador amortiguado: La expresión para la respuesta de un oscilador amortiguado sub-crítico, con dos frecuencias de oscilación naturales cercanas que producen batidos:

$$f(x) = \exp\left(-\frac{x}{\tau}\right) \left[\cos\left(\frac{2\pi x}{T_1} + \phi_1\right) + \cos\left(\frac{2\pi x}{T_2} + \phi_2\right) \right] \quad (7)$$

Los datos de la simulación de una medición de un sistema equivalente están en `rlc_nd.mat`.