USC UNIVERSIDADE DE SANTIAGO DE COMPOSTELA
FACULTADE DE MATEMÁTICAS

Traballo Fin de Grao

Resolución numérica del problema no lineal de mínimos cuadrados. Aplicaciones a la estimación de parámetros de modelos matemáticos.

Dídac Blanco Morros

Curso Académico

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

GRAO DE MATEMÁTICAS

Traballo Fin de Grao

Resolución numérica del problema no lineal de mínimos cuadrados. Aplicaciones a la estimación de parámetros de modelos matemáticos.

Dídac Blanco Morros

Febrero, 2022

UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

Trabajo propuesto

Área de Coñecemento: Matemática Aplicada
Título: Resolución numérica do problema non linear de mínimos cadrados. Aplicacións á estimación de parámetros de modelos matemáticos.
Breve descrición do contido
O problema non linear de mínimos cadrados zurde en moutas aplicacións da ciencia e da enxeñería: no axuste dun conxunto de datos a un modelo matemático, na estimación de parámetros, na aproximación de funcións, etc. O obxectivo do traballo fin de grao é o estudo de métodos numéricos para abordar o problema de minimización resultante. centrándose especialmente no algoritmo de Levenberg-Marquardt. O estudante estudará o método, no marco dos métodos de optimización con rexión de confianza e familiarizarase co seu uso mediante o comando Isqnonlin de Matlab. As metodoloxías estudadas aplicaranse a exemplos académicos e á estimación de parámetros de distintos modelos matemáticos a partir de datos experimentais.
Recomendacións
Outras observacións

Índice

Resumen

Abstract

Introducción

Capítulo 1

Fundamentos de la optimización sin restricciones

Un problema de optimización sin restricciones tiene la forma

$$\min_x f(x), \tag{1.1}$$

donde $x \in \mathbb{R}^n$ y $f : \mathbb{R}^n \to \mathbb{R}$ es continuamente diferenciable, la llamamos **función objetivo**. Notar que podemos usar esta formulación para referirnos tanto a los problemas de minimización como de maximización, basta sustituir f por $-f$.

La dificultad de un problema como este viene de no conocer el comportamiento global de f, normalmente solo disponemos de la evaluación de f en algunos puntos, y a lo mejor de algunas de sus derivadas. El trabajo de los algoritmos de optimización es identificar la solución sin usar demasiado tiempo ni almacenamiento computacional.

Definición 1.1. A una aplicación $\|\cdot\|$ se le llama *norma* si y sólo si cumple:

1. $\|x\| \geq 0$, $\forall \mathbb{R}^n$ y $\|x\| = 0$ si y solo si $x = 0$.
2. $\|\alpha x\| = |\alpha| \|x\|$, $\forall \alpha \in \mathbb{R}$, $x \in \mathbb{R}^n$.
3. $\|x + y\| \leq \|x\| + \|y\|, \forall x, y \in \mathbb{R}^n$.

Un ejemplo muy común es la *norma* l_2, también llamada *norma euclídea*, a la cual nos referiremos cuando no se especifique lo contrario, se define

$$\|x\|_2 = \left(\sum_{i=1}^{n} |x_i|^2 \right)^{\frac{1}{2}}. \tag{1.2}$$

Definición 1.2. Un punto x^* se dice *mínimo local* si existe $\delta > 0$ tal que $f(x^*) \leq f(x)$ para todo $x \in \mathbb{R}^n$ que satisface $\|x - x^*\| < \delta$. Un punto x^* se dice *mínimo local estricto* si existe $\delta > 0$ tal que $f(x^*) < f(x)$ para todo $x \in \mathbb{R}^n$ que satisface $\|x - x^*\| < \delta$ con $x \neq x^*$.

Definición 1.3. Un punto x^* se dice *mínimo global* si $f(x^*) \leq f(x)$ para todo $x \in \mathbb{R}^n$. Un punto x^* se dice *mínimo global estricto* si $f(x^*) < f(x)$ para todo $x \in \mathbb{R}^n$ con $x \neq x^*$.

Como no se suele tener un conocimiento a gran escala de f debido a su coste, la mayoría de algoritmos solo encuentran mínimos locales, lo cual es suficiente para muchos casos prácticos.

Aún así, los algoritmos para encontrar mínimos globales se suelen construír a partir de una secuencia de otros algoritmos de optimización local. También podemos aprovechar características fáciles de detectar en la función objetivo, como la convexidad, que nos asegura que un mínimo local será también global.

Definición 1.4. Sea $f : \mathbb{R}^n \to \mathbb{R}$ diferenciable en $x \in \mathbb{R}^n$ tal que $\langle \nabla f(x), d \rangle < 0$, entonces a d se le llama *dirección descendente* de f en x.

Teorema 1.5 (Teorema de Taylor). *Sea $f : \mathbb{R}^n \to \mathbb{R}$ continuamente diferenciable y sea $p \in \mathbb{R}^n$, tenemos que*

$$f(x+p) = f(x) + \nabla f(x+tp)^T p, \quad t \in (0,1). \tag{1.3}$$

Si además, f es dos veces continuamente diferenciable

$$\nabla f(x+p) = \nabla f(x) + \int_0^1 \nabla^2 f(x+tp)p \, dt, \tag{1.4}$$

y

$$f(x+p) = f(x) + \nabla f(x)^T p + \frac{1}{2} p^T \nabla^2 f(x+tp)p, \; t \in (0,1). \tag{1.5}$$

Proposición 1.6. *Partiendo de la reformulación de (1.5) y teniendo en cuenta que el último término es el error de aproximación $o(t)$*

$$f(x_k + td) = f(x_k) + t\nabla f(x_k)^T d + o(t), \tag{1.6}$$

se cumple que

$$\exists \delta > 0 \text{ tal que } f(x_k + td) < f(x_k) \quad \forall t \in (0, \delta) \tag{1.7}$$

si y solo si d es una dirección descendiente de f en x_k.

Tratemos ahora las condiciones de optimalidad.

Teorema 1.7 (Condición Necesaria de Primer Orden). *Sea $f : D \subset \mathbb{R}^n \to \mathbb{R}$ continuamente diferenciable en un conjunto abierto D. Si x^* es un mínimo local de (1.1), entonces $\nabla f(x^*) = 0$.*

Teorema 1.8. *(Condición Necesaria de Segundo Orden) Sea $f : D \subset \mathbb{R}^n \to \mathbb{R}$ dos veces continuamente diferenciable en un conjunto abierto D. Si x^* es un mínimo local de (1.1), entonces $\nabla f(x^*) = 0$ y $\nabla^2 f(x^*)$ es definida positiva.*

Teorema 1.9 (Condición Suficiente de Segundo Orden)**.** *Sea $f : D \subset \mathbb{R}^n \to \mathbb{R}$ dos veces continuamente diferenciable en un conjunto abierto D. Si $\nabla f(x^*) = 0$ y $\nabla^2 f(x^*)$ es definida positiva, entonces $x^* \in D$ es un mínimo local.*

Para poder dar un último resultado para soluciones óptimas en minimización, veamos unas últimas definiciones.

Definición 1.10. Sea $S \subset \mathbb{R}^n$ y sean $x_1, x_2 \in S$ cualesquiera. Si $\alpha x_1 + (1-\alpha)x_2 \in S$ para todo $\alpha \in [0,1]$, entonces se dice que D es un *conjunto convexo.*

Definición 1.11. Sea $S \subset \mathbb{R}^n$ un conjunto convexo no vacío. Sea $f : S \subset \mathbb{R}^n \to R$. Si para cualquiera $x_1, x_2 \in S$ y $\alpha \in (0,1)$, se cumple que

$$f(\alpha x_1 + (1-\alpha)x_2) \leq \alpha f(x_1) + (1-\alpha) f(x_2), \tag{1.8}$$

se dice que f es una función convexa en S.

Teorema 1.12. *Sea $S \subset \mathbb{R}^n$ un conjunto convexo no vacío y $f : S \subset \mathbb{R}^n \to \mathbb{R}$ una función convexa. Si x^* es mínimo local de (1.1), entonces también es mínimo global.*

Teorema 1.13. *Sea $f : \mathbb{R}^n \to \mathbb{R}$ convexa y diferenciable, entonces x^* es un mínimo global si y solo si $\nabla f(x^*) = 0$.*

Todo algoritmo de optimización sin restricciones comienza con un punto de partida, denotado normalmente como x_0. Aunque generalmente el usuario introduce una estimación razonable, el punto puede ser elegido por el algoritmo, tanto de forma sistemática como aleatoria. El algoritmo itera sobre x_0, creando una sucesión $\{x_k\}_{k=0}^n$ la cual termina cuando no pueda continuar o cuando ya se haya acercado razonablemente a la solución. Para decidir como se avanza de un x_k al siguiente, los algoritmos utilizan información sobre $f(x_k)$ o incluso en los puntos anteriores $x_0, x_1, \ldots, x_{k-1}$ con el objetivo de que $f(x_{k+1}) < f(x_k)$. Hablaremos de las dos estrategias fundamentales que se utilizan para avanzar de x_k a x_{k+1}, *búsqueda de línea* y *región de confianza.*

1.1. Búsqueda de línea

En este caso el algoritmo tiene dos tareas a partir de cada iteración, primero elige una *dirección* d_k y tomando el punto de partida busca en esa dirección el nuevo valor. Es decir, dado x_k

$$x_{k+1} = x_k + \alpha_k d_k \tag{1.9}$$

para un d_k elegido previamente, y un *paso* α_k obtenido solucionando otro problema de minimización más simple por ser unidimensional:

$$\min_{\alpha_k>0} f\left(x_k + \alpha_k d_k\right). \tag{1.10}$$

Si se toma el α_k óptimo se le llama búsqueda de línea *exacta* u *óptima*. Para evitar el gran coste computacional que puede llegar a tomar, lo más común es tomar un α_k que aporte un descenso aceptable, en cuyo caso se le llama búsqueda de línea *inexacta* o *aproximada*. Desde el nuevo punto se busca otra dirección y paso para repetir el proceso. Veamos brevemente cómo se eligen d_k y α_k.

La mayor parte de algoritmos de este tipo necesitan que d_k sea una dirección descendente, esto es, $d_k^T \nabla f_k < 0$, lo cual asegura que en esa dirección se podrá reducir el valor de f. Esta suele tener la forma

$$d_k = -B_k^{-1} \nabla f_k \tag{1.11}$$

con B_k una aproximación de la matriz Hessiana $\nabla^2 f(x_k)$ simétrica y no singular. Según lo que acabamos de decir, necesitamos que B_k sea definida positiva. En las tres corrientes principales se elige un B_k distinto, en el *método del descenso máximo* o *descenso del gradiente*, se usa la matriz identidad I. En el *método de Newton* se usa la matriz exacta, mientras que en los *métodos Quasi-Newton* la matriz Hessiana es aproximada para cada x_k.

En el caso de la elección de α_k, el caso ideal sería encontrar el óptimo en 1.10, pero esto es en general demasiado costoso. Debido a ese coste, se suelen utilizar búsquedas inexactas probando una serie de puntos hasta que alguno cumpla unas condiciones preestablecidas con las que se acepta el paso dado. Estas condiciones son por ejemplo las condiciones *Wolfe* o las condiciones *Goldstein*. Esta elección se hace en dos fases, primero un proceso elige un intervalo conteniendo los pasos deseables y una segunda fase donde se va reduciendo el intervalo por técnicas de interpolación o bisección.

1.1.1. Método de Newton

Veamos brevemente las ideas detrás del método de Newton, influyentes en los demás planteamientos. La idea principal es usar la aproximación cuadrática $q^{(k)}$ de la función objetivo,

$$q^{(k)}(p) = f(x_k) + \nabla f(x_k)^T p + \frac{1}{2} p^T \nabla^2 f(x_k) p, \tag{1.12}$$

si $f : \mathbb{R}^n \to \mathbb{R}$ dos veces continuamente diferenciable, $x_k \in \mathbb{R}^n$ y la Hessiana $\nabla^2 f(x_k)$ es definida positiva. En tal caso aproximamos $f(x_k + p) \approx q^{(k)}(p)$.

Minimizando $q^{(k)}(p)$ obtenemos la fórmula de Newton, denotando $G_k = \nabla^2 f(x_k)$ y $g_k = \nabla f(x_k)$:

$$x_{k+1} = x_k - G_k^{-1} g_k. \tag{1.13}$$

Teorema 1.14 (Teorema de Convergencia del Método de Newton). *Sea $f \in \mathcal{C}^2$ y x_k lo suficientemente cerca a la solución x^* del problema de minimización con $g(x^*) = 0$. Si la Hessiana $G(x^*)$ es definida positiva y $G(x)$ satisface la condición de Lipschitz*

$$|G_{ij}(x) - G_{ij}(y)| \leq \beta \|x - y\|, \text{ para algún } \beta, \text{ y para todo } i, j, \tag{1.14}$$

siendo $G_{ij}(x)$ el elemento en la posición (i, j) de la matriz $G(x)$, entonces para todo k, la iteración (1.13) está bien definida y la sucesión $\{x_k\}$ generada converge a x^* de forma cuadrática.

1.2. Región de confianza

Esta estrategia enfoca el problema de otro modo, primero se fija una distancia máxima Δ_k para definir una región, generalmente de la forma

$$\Omega_k = \{x : \|x - x_k\| \leq \Delta_k\} \tag{1.15}$$

y luego ya se busca la dirección y paso. A partir de la información conocida de f, para cada x_k se modela una función m_k que se comporte de manera similar a f cerca de este punto. Se suele utilizar el modelo cuadrático $q^{(k)}$ visto anteriormente (1.12), aprovechando la notación usada en el apartado anterior:

$$m_k(p) := q^{(k)}(p) = f(x_k) + g_k^T p + \frac{1}{2} p^T G_k p. \tag{1.16}$$

Como hemos visto, este modelo se utiliza en los métodos de búsqueda de línea para determinar la dirección de búsqueda, mientras que en este caso lo usamos para tener una representación adecuada de la función objetivo y así elegir el mínimo dentro de esta región. Este método nos evita el problema de que la Hessiana no sea definida positiva. En cada iteración, una vez elegido Δ_k se resuelve el siguiente problema:

$$\begin{aligned} \min_p \quad & m_k(p) = f(x_k) + g_k^T p + \frac{1}{2} p^T B_k p \\ \text{s.a.} \quad & \|p\| \leq \Delta_k. \end{aligned} \tag{1.17}$$

Notamos que en el modelo se escribe B_k en lugar de G_k, pues no siempre se usa esta última. Debido al coste computacional, como vimos en la elección de la dirección de búsqueda, a veces se prefiere aproximar de alguna manera más o menos eficiente, e incluso puede ser aceptable tomar la matriz 0.

También se puede elegir qué norma define la región de confianza, cambiando así la forma de esta y ofreciendo distintos resultados, aunque generalmente se utiliza la bola definida por $\|p\|_2 \leq \Delta_k$.

La efectividad de cada iteración depende de la elección del radio Δ_k, es por ello que puede que la primera elección de este no sea la definitiva. Es decir, se toma un radio a raíz de la información que se tenga, esta puede incluír la de pasos anteriores, y luego se decide si este radio nos da un resultado aceptable. Un radio demasiado pequeño nos puede hacer perder la oportunidad de ser mucho más rápidos, pero un paso demasiado grande, el mínimo de la función modelo m_k puede estar lejos del mínimo de la función objetivo. Este último caso es el que se comprueba y se decide si reducir la región de confianza.

Una vez tomado el radio, encontrar el mínimo es directo en el caso de que B_k sea definida positiva, basta tomar $p_k^B = -B_k^{-1} g_k$, conocido como *paso completo*. En caso contrario tampoco supone una tarea muy costosa ya que sólo se necesita una solución aproximada para garantizar la convergencia.

Veamos ahora de forma detallada como se elige el radio Δ_k en cada iteración. Esta elección se toma según el parecido entre la función f y el modelo m_k tomado en las iteraciones previas. Dado p_k, definimos el ratio

$$\rho_k = \frac{f(x_k) - f(x_k + p_k)}{m_k(0) - m_k(p_k)}, \tag{1.18}$$

donde el numerador es la *reducción real*, mientras que el denominador es la *reducción prevista*. La reducción prevista será positiva por definición, pues p_k es elegido para tener el menor valor posible y el 0 es una posibilidad. Por tanto, si ρ_k es negativo, el nuevo valor $f(x_k + p_k)$ no es menor que $f(x_k)$ y este paso ha de ser rechazado. Por otro lado, si ρ_k es cercano a 1, esto quiere decir que f y m_k se comportan de manera similar en la región tomada en la iteración actual, por tanto podemos agrandar el radio con seguridad. En resumen, nos quedamos con el radio elegido si ρ_k no tiene un valor muy cercano a 0 o a 1. El proceso se describe en el siguiente algoritmo.

Algoritmo 1.15 (Región de confianza)**.**

Dado $\hat{\Delta} > 0, \Delta_0 \in (0, \hat{\Delta})$, *y* $\eta \in [0, \frac{1}{4})$
for $k \leftarrow 0, 1, 2, \ldots$ ***do***
    Obtener p_k *(1.17).*
    Calcular ρ_k *(1.18).*
    if $\rho_k < \frac{1}{4}$ ***then***
        $\Delta_{k+1} \leftarrow \frac{1}{4}\Delta_k$
    else
        if $\rho_k > \frac{3}{4}$ *y* $\|p_k\| = \Delta_k$ ***then***
            $\Delta_{k+1} \leftarrow \text{mín}(2\Delta_k, \hat{\Delta})$
        else

11: $\Delta_{k+1} \leftarrow \Delta_k$
12: ***end if***
13: ***end if***
14: ***if*** $\rho_k > \eta$ ***then***
15: $x_{k+1} \leftarrow x_k + p_k$
16: ***else***
17: $x_{k+1} \leftarrow x_k$
18: ***end if***
19: ***end for***.

Aquí $\hat{\Delta}$ es el máximo radio de la región de cada iteración. Notar que únicamente se aumenta el radio en el caso de que $\|p_k\| = \Delta_k$, ya en caso contrario, se entiende que el Δ_k elegido no influye en la elección de forma negativa.

Para que este algoritmo sea práctico, nos centramos en la resolución del subproblema (1.17). Tomemos una notación más limpia,

$$\begin{aligned} \min_{p} \quad & m(p) = f + g^T p + \frac{1}{2} p^T B p, \\ \text{s.a.} \quad & \|p\| \leq \Delta. \end{aligned} \tag{1.19}$$

Teorema 1.16. *El vector p^* es una solución global de (1.19) si y solo si p^* cumple las condiciones y existe un escalar $\lambda \geq 0$ tal que se cumplen las siguientes condiciones:*

1. $(B + \lambda I)p^* = -g$,
2. $\lambda(\Delta - \|p^*\|) = 0$,
3. $(B + \lambda I)$ *es semidefinida positiva.*

Demostración. página 90 Nocedal. □

Este teorema caracteriza la solución según el primer punto. El segundo punto es una condición complementaria que nos dice que al menos uno de los dos factores es 0. Esto es, si $\|p\| < 0$, λ tendrá que ser 0 y $Bp^* = -g$ con B definida positiva (puntos 1 y 3). En el caso de que $\|p\|$ se maximice, lo cual da a entender que la solución óptima no se encuentra dentro de la región de confianza, λ podrá tomar valores positivos.

Capítulo 2

Mínimos Cuadrados

El problema de mínimos cuadrados surge de la necesidad de ajustar modelos que nos permitan predecir ciertos comportamientos en una amplia variedad de campos. Dados unos datos $(t_1, y_1), (t_2, y_2), \cdots, (t_m, y_m)$, queremos ajustar una función $\phi(t, x)$ de forma que se minimicen los residuos $r_i(x) = \phi(t_i, x) - y_i$ para $i = 1, \cdots, m$

$$\min_{x \in \mathbb{R}^n} f(x) = \frac{1}{2} r(x)^T r(x) = \frac{1}{2} \sum_{i=1}^{m} r_i^2(x), \quad m \geq n, \tag{2.1}$$

donde $r(x) = (r_1(x), r_2(x), \ldots, r_m(x))^T$, con $r_i : \mathbb{R}^n \to \mathbb{R}$ funciones continuamente diferenciables.

Veamos las propiedades de este modelo concreto de optimización sin restricciones y cómo se pueden aprovechar para formular algoritmos eficientes y robustos. Sea $J(x)$ la matriz Jacobiana de $r(x)$,

$$J(x) = \begin{bmatrix} \nabla r_1(x)^T \\ \nabla r_2(x)^T \\ \vdots \\ \nabla r_m(x)^T \end{bmatrix} = \begin{bmatrix} \frac{\partial r_1}{\partial x_1}(x) & \frac{\partial r_1}{\partial x_2}(x) & \cdots & \frac{\partial r_1}{\partial x_n}(x) \\ \frac{\partial r_2}{\partial x_1}(x) & \frac{\partial r_2}{\partial x_2}(x) & \cdots & \frac{\partial r_2}{\partial x_n}(x) \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial r_m}{\partial x_1}(x) & \frac{\partial r_m}{\partial x_2}(x) & \cdots & \frac{\partial r_m}{\partial x_n}(x) \end{bmatrix}. \tag{2.2}$$

El gradiente y la Hessiana de f se pueden expresar como sigue:

$$g(x) = \nabla f(x) = \sum_{i=1}^{m} r_i(x) \nabla r_i(x) = J(x)^T r(x), \tag{2.3}$$

$$\begin{aligned} G(x) = \nabla^2 f(x) &= \sum_{i=1}^{m} \nabla r_i(x) \nabla r_i(x)^T + \sum_{i=1}^{m} r_i(x) \nabla^2 r_i(x) \\ &= J(x)^T J(x) + S(x). \end{aligned} \tag{2.4}$$

Si nos fijamos en la formulación de la matriz Hessiana, el cálculo del primer termino es directo gracias a que ya obtenemos $J(x)$ para calcular el gradiente (2.3), así que el coste se reduce al

segundo término, que hemos denotado $S(x)$. Adaptando el modelo cuadrático (1.12)

$$q^{(k)}(x) = f(x_k) + g_k^T(x - x_k) + \frac{1}{2}(x - x_k)^T G_k(x - x_k), \tag{2.5}$$

y sustituyendo según (2.1), (2.3) y (2.4), obtenemos el método de Newton para (2.1),

$$x_{k+1} = x_k - (J(x_k)^T J(x_k) + S(x_k))^{-1} J(x_k)^T r(x_k). \tag{2.6}$$

2.1. El Problema Lineal

El primer caso más sencillo es si $\phi(t, x)$ es una función lineal, en cuyo caso los residuos $r_i(x)$ también serán lineales. Por ser ϕ lineal, se puede representar como Jx, con J una matriz $m \times n$. Realizaremos un estudio del caso lineal para tener un conocimiento de como se enfocan estos problemas, que nos servirá para entender mejor el caso no lineal. Si escribimos el vector residuo como $r(x) = Jx - y$, la función objetivo nos queda de la forma

$$f(x) = \frac{1}{2}\|Jx - y\|^2. \tag{2.7}$$

En consecuencia, tomando como referencia (2.3) y (2.4) y teniendo en cuenta que en este caso particular $\nabla^2 r_i = 0$, nos queda

$$\nabla f(x) = J^T(Jx - y), \qquad \nabla^2 f(x) = J^T J. \tag{2.8}$$

Como f es convexa, dado un punto x^* tal que $\nabla f(x^*) = 0$, este será mínimo global (1.13). Por tanto, x^* satisface el siguiente sistema lineal:

$$J^T J x^* = J^T y. \tag{2.9}$$

Antes de ver como se resuelve numéricamente este sistema de ecuaciones, conocidas como *ecuaciones normales* de (2.7), veamos los conceptos previos necesarios.

Definición 2.1. Un problema numérico se dice *bien condicionado* si su solución no se ve afectada por pequeñas perturbaciones a cualquiera de los datos que definen el problema.

Definición 2.2. Una matriz cuadrada A se dice *definida positiva* si existe un $\alpha \in \mathbb{R}^+$ tal que

$$x^T A x \geq \alpha x^T x, \quad \text{para todo } x \in \mathbb{R}^n. \tag{2.10}$$

Esta es *semidefinida postiva* si

$$x^T A x \geq 0, \quad \text{para todo } x \in \mathbb{R}^n. \tag{2.11}$$

Definición 2.3. Una matriz $n \times n$ cuadrada A se dice *no singular* si para cada $b \in \mathbb{R}^n$, existe $x \in \mathbb{R}^n$ tal que $Ax = b$.

Definición 2.4. Una matriz cuadrada Q se dice *ortogonal* si cumple $QQ^T = Q^TQ = I$

Definición 2.5. Si tomamos los sistemas de vectores de una una matriz $A_{m\times n}$, $\{u_i\}_{i=1}^n$ y $\{v_i\}_{i=1}^m$, al número máximo de vectores linealmente independientes se le llama *rango*, tanto de los sistemas de vectores como de la matriz A. Si $n < m$, se dice que A es de *rango completo* si su rango es n, que es el máximo posible.

Lo más común en este caso para resolver numéricamente es usar distintos tipos de factorización sobre la matriz J^TJ o sobre J, para luego resolver con sustituciones triangulares. El primer algoritmo que se plantea es a partir de la **factorización de Cholesky**, comenzando por computar la matriz de coeficientes J^TJ y el lado derecho J^Ty. Después se computa la factorización de Cholesky

$$J^TJ = \bar{R}^T\bar{R}. \tag{2.12}$$

Para que esta exista, necesitamos que $m \geq n$ y que J sea de rango n, lo que permite que J^TJ sea simétrica y definida positiva. Se termina realizando las dos sustituciones triangulares con los factores de Cholesky para encontrar x^*. La principal desventaja de este método es que el condicionamiento de J^TJ es el cuadrado del condicionamiento de J, y esto puede llevar a errores de aproximación. Además, si J está mal condicionada, ni si quiera se puede llevar a cabo la factorización.

Una segunda posibilidad es basarse en la **factorización QR**, que evita el problema de depender del cuadrado del condicionamiento de J, ya que aplicaremos la factorización directamente a J. Se aprovecha que la norma euclídea no se ve afectada por trasformaciones ortogonales para partir de la igualdad

$$\|Jx - y\| = \|Q^T(Jx - y)\|, \tag{2.13}$$

siendo Q una matriz ortogonal $m \times m$. Factorizando con una matriz pivote Π, la solución es

$$x^* = \Pi R^{-1}Q_1^Ty. \tag{2.14}$$

Donde R es una matriz $n \times n$ triangular superior con elementos positivos en la diagonal y Q_1 son las primeras n columnas de Q, ambos producto de la factorización QR.

En este caso, el error relativo es proporcional al condicionamiento de J y no de su cuadrado. Aún así, hay situaciones en las que necesitamos asegurar que la obtención sea de algún modo más robusta o en las que queremos más información acerca de la sensibilidad de la solución a perturbaciones en J o en y. Esto es, queremos asegurarnos que pequeños cambios en J o y no afecten significativamente a la solución obtenida, lo cual pondría en duda nuestro método ya que estas perturbaciones se pueden dar durante la computación.

Nos basaremos ahora en la **descomposición de valores singulares (DVS)**, cuya resolución una vez realizada la descomposición se enfoca de forma similar a la anterior. Primero se realiza el algoritmo (DVS) para obtener $J = USV^T$, con U matriz $m \times m$, V matriz $n \times n$, ambas ortogonales y S matriz $n \times n$ de elementos diagonales $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$. Aprovechamos estas propiedades para obtener

$$x^* = VS^{-1}U_1^T y. \tag{2.15}$$

Denotando por $u_i \in \mathbb{R}^m$ y $v_i \in \mathbb{R}^n$ las columnas de U y V respectivamente, escribimos

$$x^* = \sum_{i=1}^{n} \frac{i_i^T y}{\sigma_i} v_i. \tag{2.16}$$

Fórmula de donde obtenemos información útil como la sensibilidad al aproximar x^*.

Las 3 opciones son buenas según las condiciones en las que nos encontremos. La resolución basada en Cholesky es útil cuando $m \gg n$ y es práctico trabajar almacenando $J^T J$ en lugar de J, siempre y cuando J sea de rango completo y bien condicionada. Si esto último no se cumple, la factorización QR es un enfoque más equilibrado, mientras que DVS es el más costoso a cambio de ser el más fiable.

Por último, mencionar que existen métodos para problemas muy grandes, en los que se usan técnicas iterativas como el método de gradientes conjugados para resolver el sistema.

2.2. El método de Gauss-Newton

Comenzamos los métodos de minimización del problema no lineal (2.1) con el método de Gauss-Newton. La forma más sencilla de abordar el término de segundo orden $S(x)$ de G_k en el modelo cuadrático (2.5) es obviarlo. Así, resulta

$$\begin{aligned} \bar{q}^{(k)}(x) =& \frac{1}{2} r(x_k)^T r(x_k) + (J(x_k)^T r(x_k))^T (x - x_k) + \\ &+ \frac{1}{2}(x - x_k)^T (J(x_k)^T J(x_k))(x - x_k), \end{aligned} \tag{2.17}$$

y por tanto,

$$x_{k+1} = x_k + p_k = x_k - (J(x_k)^T J(x_k))^{-1} J(x_k)^T r(x_k). \tag{2.18}$$

Notar que para que esté bien definido, la matriz Jacobiana $J(x)$ tiene que ser de rango completo. Gracias a la aproximación $\nabla^2 f(x_k) \approx J(x_k)^T J(x_k)$, hacemos que la única dificultad del algoritmo sea resolver un sistema lineal, ya que evitamos computar $\nabla^2 r_j$, $j = 1, 2, \ldots, m$. En algunas situaciones, cuando nos vamos acercando a la solución x^*, esta aproximación suele ser más precisa, ya sea porque los residuos r_i o $\left\|\nabla^2 r_i\right\|$ es cercano a cero. La eficacia del método dependerá por tanto de lo buena que sea esta aproximación.

Algoritmo 2.6 (Método de Gauss-Newton)**.**

Paso 1. x_0 *y* $\epsilon > 0$ *dados,* $k := 0$

Paso 2. Si $\|g_k\| \leq \epsilon$*, STOP.*

Paso 3. Obtener el paso p_k *resolviendo*

$$J(x_k)^T J(x_k) p_k = -J(x_k)^T r(x_k) \tag{2.19}$$

Paso 4. Definimos $x_{k+1} = x_k + p_k$ *y actualizamos* $k = k + 1$*. Ir a Paso 2.* □

Siempre y cuando J tenga rango completo y el gradiente $\nabla f_k = J(x_k)^T r(x_k)$ sea no nulo, la dirección p_k es una dirección descendente. Como vemos en el *Paso 3* resolvemos un caso análogo al problema lineal. Debido a esto, p_k es también la solución de

$$\min_{p_k} \frac{1}{2} \|J(x_k)p_k + r_k\|^2, \tag{2.20}$$

y por eso decimos que el método de Gauss-Newton es en realidad una linealización del problema no lineal de mínimos cuadrados y, como el método de Newton, resultará localmente convergente de manera cuadrática bajo las condiciones de este. Por tanto, el error de aproximación de la solución dependerá también de como resolvamos el problema lineal, y aplican los casos vistos en el apartado anterior.

Teorema 2.7. *Sea* $f : \mathbb{R}^n \to \mathbb{R}$ *y* $f \in \mathcal{C}^2$*. Supongamos que* x^* *es mínimo local del problema (2.1),* $J(x^*)^T J(x^*)$ *es definida positiva y la sucesión* $\{x_k\}$ *generada por el algoritmo (2.6) converge a* x^**. Si* $G(x)$ *y* $(J(x)^T J(x))^{-1}$ *son lipschitzianas en una vecinidad de* x^**, entonces*

$$\|x_{k+1} - x^*\| \leq \|(J(x^*)^T J(x^*))^{-1}\| \, \|S(x^*)\| \, \|x_k - x^*\| + O(\|x_k - x^*\|). \tag{2.21}$$

Este teorema nos dice esencialmente que la convergencia del método depende de $S(x^*)$. Cuando $S(x^*) = 0$ este converge de forma cuadrática y, según aumente, la convergencia es menor.

Teorema 2.8. *Sea* $f : D \subset \mathbb{R}^n \to \mathbb{R}$ *y* $f \in \mathcal{C}^2(D)$*, con* D *conjunto abierto convexo. Sea* $J(x)$ *lipschitziana en* D *y* $\|J(x)\| \leq \alpha, \forall x \in D$*. Supongamos que existe* $x^* \in D$ *y* $\lambda, \sigma \geq 0$ *tal que* $J(x^*)^T r(x^*) = 0$*,* λ *es el autovalor más pequeño de* $J(x^*)^T J(x^*)$*, y*

$$\left\|(J(x) - J(x^*))^T r(x^*)\right\| \leq \sigma \|x - x_k\|, \forall x \in D. \tag{2.22}$$

Si $\sigma < \lambda$*, para cualquier* $c \in (1, \lambda/\sigma)$*, existe* $\epsilon > 0$ *tal que para todo* $x_0 \in N(x^*, \epsilon)$*, la sucesión resultante* $\{x_k\}$ *del algoritmo 2.6 está bien definida, converge a* x^* *y satisface*

$$\|x_{k+1} - x^*\| \leq \frac{c\sigma}{\lambda} \|x_k - x^*\| + \frac{c\alpha\sigma}{2\lambda} \|x_k - x^*\|^2 \tag{2.23}$$

y

$$\|x_{k+1} - x^*\| \leq \frac{c\alpha + \lambda}{2\lambda} \|x_k - x^*\| < \|x_k - x^*\|. \tag{2.24}$$

Teorema 2.9. *Manteniendo las suposiciones de los dos teoremas 2.7 y 2.8, si* $r(x^*) = 0$*, entonces existe* $\epsilon > 0$ *tal que para cualquier* $x_0 \in N(x^*, \epsilon)$*, la sucesión* $\{x_k\}$ *obtenida del método de Gauss-Newton converge a* x^* *con orden cuadrático.*

Para concluir observamos que el método encaja dentro de los métodos de búsqueda de línea, tomando $d_k = p_k$ en (1.9) y calculando una longitud de paso α_k,

$$x_{k+1} = x_k - \alpha_k (J(x_k)^T J(x_k))^{-1} J(x_k)^T r(x_k). \tag{2.25}$$

Capítulo 3

El método de Levenberg-Marquardt

El método de Levenberg-Marquardt soluciona la necesidad de que J sea de rano completo cambiando el enfoque de búsqueda de línea por el de región de confianza. Lo hace manteniendo la raíz del método de Gauss-Newton, la linealización del problema obviando el término cuadrático, esto es, usando la aproximación $\nabla^2 f(x_k) \approx J(x_k)^T J(x_k)$. Este cambio de enfoque surge de que la linealización pierde efectividad según nos alejamos de x_k, por lo que conviene restringir el tamaño de $p = (x - x_k)$. Consideramos el problema:

$$\min_p \quad \frac{1}{2} \|J_k p + r_k\|^2, \quad \text{s.a. } \|p\| \leq \Delta_k, \tag{3.1}$$

Donde $\Delta_k > 0$ es el radio de la región de confianza. De hecho, se cree que esta solución característica de los métodos de región de confianza nace con este método en concreto. Otra forma de escribir el modelo es

$$m_k(p) = \frac{1}{2} \|r_k\|^2 + p^T J_k^T r_k + \frac{1}{2} p^T J_k^T J_k p, \quad \text{s.a. } \|p\| \leq \Delta_k. \tag{3.2}$$

La solución de este subproblema queda caracterizada por el sistema

$$(J^T J + \lambda I)p = -J^T r. \tag{3.3}$$

Lema 3.1. *El vector p es solución del subproblema (3.1) si y solo si p es factible y existe un $\lambda \geq 0$ tal que*

$$(J^T J + \lambda I)p = -J^T r, \tag{3.4}$$

$$\lambda(\Delta - \|p\|) = 0. \tag{3.5}$$

Demostración. Es consecuencia del teorema 1.16. Solo hay que ver que se cumplen las 3 condiciones, las dos primeras se siguen del propio lema. La tercera pide que $(J^T J + \lambda I)$ sea semidefinida positiva, y lo es por serlo $J^T J$ y por ser $\lambda > 0$ □

En concreto, como $(J^TJ + \lambda I)$ es definida positiva, la solución de (**??**) es una dirección descendente. Lo que nos quiere decir este lema es que si la solución obtenida por el método de Gauss-Newton cae estrictamente dentro de la región de confianza, esta solucionará también el subproblema (3.1). En otro caso, existe un $\lambda > 0$ que permite encontrar una solución a (3.3) con $\|p\| = \Delta$. Si $\lambda = 0$, la solución del problema es la de Gauss-Newton, y según aumenta λ, la solución se acerca a la del método de máximo descenso. Veamos una serie de propiedades del método de Levenberg-Marquardt y de p en función de λ.

Teorema 3.2. *Si λ aumenta desde cero monótonamente, $\|p(\lambda)\|$ decrece de forma estrictamente monótona.*

Demostración. Por un lado,

$$\frac{\mathrm{d}}{\mathrm{d}\lambda}\|p\| = \frac{\mathrm{d}}{\mathrm{d}\lambda}(p^Tp)^{\frac{1}{2}} + \frac{p^T\frac{\mathrm{d}p}{\mathrm{d}\lambda}}{\|p\|}. \tag{3.6}$$

Derivando (3.3) respecto a λ, obtenemos

$$p + (J^TJ + \lambda I)\frac{\mathrm{d}p}{\mathrm{d}\lambda} = 0, \tag{3.7}$$

de esta y (3.3) resulta

$$\frac{\mathrm{d}p}{\mathrm{d}\lambda} = (J^TJ + \lambda I)^{-2}g, \tag{3.8}$$

con $g = J^Tr$. Sustituyendo en (3.6) y usando (3.3), nos queda

$$\frac{\mathrm{d}}{\mathrm{d}\lambda}\|p\| = -\frac{g^T(J^TJ + \lambda I)^{-3}g}{\|p\|}. \tag{3.9}$$

Cuando $\lambda \geq 0$, $J^TJ + \lambda I$ es definida positiva. Por tanto $\|p(\lambda)\|$ de forma estrictamente monótona. □

Teorema 3.3. *Sea $\lambda_k > 0$, si p_k es solución de (3.3), entonces p_k es solución global de el subproblema*

$$\min_p \quad m_k(p) = \frac{1}{2}\|J_kp + r_k\|^2, \quad s.a.\ \|p\| \leq \|p_k\|, \tag{3.10}$$

Demostración. Como p_k es solución de (3.3), entonces

$$\begin{aligned} m_k(p_k) &= \frac{1}{2}r_k^Tr_k + r_k^TJ_kp_k + \frac{1}{2}p_k^TJ_k^TJ_kp_k \\ &= \frac{1}{2}r_k^Tr_k - p_k^T(J_k^TJ_k + \lambda_k I)p_k + \frac{1}{2}p_k^TJ_k^TJ_kp_k \\ &= \frac{1}{2}r_k^Tr_k - \lambda_k p_k^Tp_k - \frac{1}{2}p_k^TJ_k^TJ_kp_k. \end{aligned} \tag{3.11}$$

Por otro lado, para un p cualquiera, tenemos

$$\begin{aligned} m_k(p) &= \frac{1}{2} r_k^T r_k + p^T J_k^T r_k + \frac{1}{2} p^T J_k^T J_k p \\ &= \frac{1}{2} r_k^T r_k - p^T (J_k^T J_k + \lambda_k I) p_k + \frac{1}{2} p^T J_k^T J_k p \\ &= \frac{1}{2} r_k^T r_k - \lambda_k p^T p_k - p^T J_k^T J_k p_k + \frac{1}{2} p_k^T J_k^T J_k p_k. \end{aligned} \tag{3.12}$$

Por tanto, para cualquier p tal que $\|p\| \leq \|p_k\|$, se cumple

$$\begin{aligned} m_k(p) - m_k(p_k) &= \frac{1}{2} (p_k - p)^T J_k^T J_k (p_k - p) + \lambda_k (p_k^T p_k - p^T p_k) \\ &\geq \frac{1}{2} (p_k - p)^T J_k^T J_k (p_k - p) + \lambda_k \|p_k\| (\|p_k\| - \|p\|) \\ &\geq 0, \end{aligned} \tag{3.13}$$

por lo que p_k es una solución global óptima del problema (3.10). □

Anexo I

Título del Anexo I

Anexo II

Título del Anexo II

Bibliografía

[1] Nocedal, J., & Wright, S. (2006). *Numerical Optimization* (2nd ed.). Springer.

[2] Sun, W., & Yuan, Y.-X. (2006). *Optimization theory and methods: Nonlinear programming* (2006th ed.). Springer.