Lecturas de Optimización Numérica:

Métodos de Newton y de Descenso o Búsqueda Lineal

Prof. Dr. Fredy Vides
Scientific Computing Innovation Center, UNAH &
Centre for Analysis of Data-Driven Systems
E-mail: fredy.vides@unah.edu.hn

Índice

Objetivos

1. Estudiar los prinicipios fundamentales de optimización por métodos de Newton.
2. Estudiar los prinicipios fundamentales de optimización por métodos de descenso o búsqueda Lineal.

1. Métodos de Newton

Considerando una función $f \in C^2(\mathbb{R}^n, \mathbb{R})$ cuyas primeras y segundas derivadas parciales podemos calcular. Si se considera ahora el problema determinado por el siguientes sistema de ecuaciones no lineales.

$$\nabla f(x) = 0 \tag{1.1}$$

Es posible aplicar una expansión truncada de Taylor en varias variables para calcular una aproximación de primer orden de $\nabla f(x)$ determinada por la siguiente expresión.

$$\nabla f(x+h) \approx \nabla f(x) + (\mathbb{J}\nabla f(x))h \tag{1.2}$$

Aplicando (1.1) es posible aproximar el vector h en (1.2) a través de la siguiente expresión.

$$h \approx -(\mathbb{H}f(x))^{-1}\nabla f(x) \tag{1.3}$$

Donde se ha aplicado la siguiente identidad considerada en lecturas previas.

$$\mathbb{H}f(x) = \mathbb{J}\nabla f(x) \tag{1.4}$$

Definición 1.1. La expresión (1.3) permite derivar lo que se denomina en este curso un **método de Newton**

Un método de Newton puede aplicarse para resolver el problema determinado por el cómputo de un punto **estacionario** o **crítico** de f, el cual como se estudio en lecturas previas, puede estar relacionado con problemas de la forma.

$$x = \operatorname{argmin}_{y \in \mathcal{S}} f(y) \tag{1.5}$$

donde $\mathcal{S} \subset \mathbb{R}^n$. La notación utilizada en (1.5) permite hacer énfasis en que el problema que se busca resolver es el correspondiente al cálculo de un minimizador de global o local de la función objetivo f.

La ideas previamente consideradas para los métodos de Newton pueden aplicarse para derivar un algoritmo prototípico como el que se muestra a continuación.

Algoritmo 1. **Método de Newton**

> **entradas** x_0, N, ε
> **salidas** k, x_k
> **para** $k = 0, \ldots, N$ **hacer**
> **resolver** $\mathbb{H}f(x_k)h_k = -\nabla f(x_k)$
> $x_{k+1} \leftarrow x_k + h_k$
> **si** $\|x_{k+1} - x_k\|_2 \leq \varepsilon$ **interrumpir**
> **fin**

2. Métodos de Descenso o Búsqueda Lineal

Suposición 2.1. Por simplicidad, en esta sección se considerará que $f \in C^2(\mathbb{R}^n, \mathbb{R})$ está acotada inferiormente.

Un método de descenso o búsqueda lineal es un método iterativo que produce una sucesión $\{x_k\}_{k \geq 0}$ donde cada x_{k+1} depende de:

- x_k
- un vector d_k que depende de $\nabla f(x_k)$
- un parámetro adecuado $\alpha_k \in \mathbb{R}$.

Dado $x_0 \in \mathbb{R}^n$ la forma genérica del método puede abreviarse en el siguiente algoritmo prototípico.

Algoritmo 2. **Método Genérico de Búsqueda Lineal**

> **entradas** x_0, N, ε
> **salidas** k, x_k
> **para** $k = 0, \ldots, N$ **hacer**
> **encontrar dirección** $d_k \in \mathbb{R}^n$
> **calcular** $\alpha_k \in \mathbb{R}$
> $x_{k+1} \leftarrow x_k + \alpha_k d_k$
> **si** $\|x_{k+1} - x_k\|_2 \leq \varepsilon$ **interrumpir**
> **fin**

Observación 2.2. Es importante establecer que el vector d_k debe ser una *dirección de descenso*, en el sentido de que se cumplen las siguientes condiciones:

$$
\begin{aligned}
&d_k^\top \nabla f(x_k) < 0, && \text{si } \nabla f(x_k) \neq 0,\\
&d_k = 0, && \text{si } \nabla f(x_k) = 0.
\end{aligned} \tag{2.1}
$$

2.1. Direcciones de descenso de uso más frecuente.

2.1.1. Direcciones de Newton.

$$d_k = -(\mathbb{H}f(x_k))^{-1}\nabla f(x) \tag{2.2}$$

2.1.2. Direcciones de Casi-Newton.

$$d_k = -H_k^{-1}\nabla f(x) \tag{2.3}$$

donde $H_k \approx \mathbb{H}f(x_k)$.

2.1.3. Direcciones Gradiente.

$$d_k = -\nabla f(x) \tag{2.4}$$

donde $H_k \approx \mathbb{H}f(x_k)$.

2.1.4. Direcciones Gradiente Conjugadas.

$$
\begin{aligned}
&d_0 = -\nabla f(x)\\
&d_{k+1} = -\nabla f(x_{k+1}) - \beta_k d_k, \;\; k \geq 0.
\end{aligned} \tag{2.5}
$$

donde consideraremos que $\beta_k = -\|\nabla f(x_k)\|_2^2 / \|\nabla f(x_{k-1})\|_2^2$.

2.2. Estrategias de selección de la longitud de paso α_k. Dada una dirección de búsqueda d_k, en términos de óptimos α_k debería cumplir de forma ideal la siguiente condición.

$$\alpha_k = \text{argmin}_{\alpha\in\mathbb{R}} f(x_k + \alpha d_k) \tag{2.6}$$

Calculando una expansión truncada de Taylor de segundo orden alrededor de x_k se obtiene la siguiente expresión.

$$f(x_k + \alpha_k d_k) \approx f(x_k) + \alpha d_k^\top \nabla f(x_k) + \frac{\alpha^2}{2} d_k^\top \mathbb{H}f(x_k) d_k \tag{2.7}$$

Aplicando (2.7) en el caso particular de una función cuadrática de la forma

$$f(x) = \frac{1}{2}x^\top A x - x^\top b + c \tag{2.8}$$

con $A \in \mathbb{R}^{n\times n}$ SPD, $b \in \mathbb{R}^n$, y $c \in \mathbb{R}$, La expansión (2.7) es exacta. En este caso $\mathbb{H}f(x_k) = A$, $\nabla f(x_k) = Ax_k - b = -r_k$, para cada $k \geq 0$. Además, en el caso particular de f determinada por (2.8), es posible calcular el valor óptimo de alpha aplicando las técnicas aplicadas en la solución de ejercicios de las lecturas previas, obteniendo la siguiente expresión.

$$\alpha_k = \frac{d_k^\top r_k}{d_k^\top A x_k} \tag{2.9}$$

Condiciones de Wolfe. En general un criterio apropiada para el cómputo de $\alpha_k > 0$ está basado en lo que se conoce como las **condiciones de Wolfe**, determinadas por las siguientes expresiones.

$$\begin{aligned} f(x_k + \alpha_k d_k) &\leq f(x_k) + \sigma \alpha_k d_k^\top \nabla f(x_k) \\ d_k^\top \nabla f(x_k + \alpha_k d_k) &\geq \delta d_k^\top \nabla f(x_k) \end{aligned} \tag{2.10}$$

donde σ, δ son constantes predeterminadas para el problema, que cumplen $0 < \sigma < \delta < 1$.

2.3. Método de Descenso con direcciones de casi Newton. Las aproximaciones $H_k \approx \mathbb{H}f(x_k)$ deben ser calculadas de manera que cumplan las siguientes restricciones.

- $H_{k+1}(x_{k+1} - x_k) = \nabla f(x_{k+1}) - \nabla f(x_k)$
- H_k es SPD
- H_k cumple además la siguiente condición.

$$\lim_{k \to \infty} \frac{\|(H_k - \mathbb{H}f(x_k))d_k\|_2}{\|d_k\|} = 0$$

Es posible calcular las matrices H_k utilizando la estrategia de Broyden, Fletcher, Goldfarb y Shanno (BFGS) basada en la siguiente relación de recurrencia.

$$H_{k+1} = H_k + \frac{1}{y_k^\top s_k} y_k y_k^\top - \frac{1}{s_k^\top H_k s_k} H_k s_k s_k^\top H_k^\top \tag{2.11}$$

donde $s_k = x_{k+1} - x_k$ y $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$.

Un algoritmo prototípico de búsqueda lineal con direcciones de casi Newton puede ahora ser derivado.

Algoritmo 3. **Método de Búsqueda Lineal con Direcciones de Casi Newton**

entradas $x_0, H_0 \approx \mathbb{H}f(x_0), N, \varepsilon$
salidas k, x_k
para $k = 0, \ldots, N$ **hacer**
resolver $H_k d_k = -\nabla f(x_k)$
calcular $\alpha_k \in \mathbb{R}$ que cumple (2.10)
$x_{k+1} \leftarrow x_k + \alpha_k d_k$
$s_k \leftarrow x_{k+1} - x_k$
$y_k \leftarrow \nabla f(x_{k+1}) - \nabla f(x_k)$
calcular H_k aplicando (2.11)
si $\|s_k\|_2 \leq \varepsilon$ **interrumpir**
fin

3. Mini Proyecto de Aplicación: Ajuste por Mínimos Cuadrados

Considerando el problema determinado por el ajuste de una colección de datos $\mathcal{D} = \{(x_j, y_j)\}_{j=1}^N \subset \mathbb{R}^2$ a través de una curva determinada por una expresión $y = f(x)$ donde $f(x) = a_0 + a_1 x + a_2 x^2 + \cdots a_n x^n$.

Considerando además la matriz $X \in \mathbb{R}^{N \times (n+1)}$ definida por la expresión.

$$X = \begin{bmatrix} x_1^n & x_1^{n-1} & \cdots & x_1^2 & x_1 & 1 \\ x_2^n & x_2^{n-1} & \cdots & x_2^2 & x_2 & 1 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots \\ x_N^n & x_N^{n-1} & \cdots & x_N^2 & x_N & 1 \end{bmatrix} \tag{3.1}$$

Para este proyecto se hace la siguiente suposición.

Suposición 3.1. Se supone que $N \geq n+1$, y que las columnas de la matriz $X \in \mathbb{R}^{N\times(n+1)}$ definida por (3.1) son linealmente independientes.

Notación 3.2. Sean $a \in \mathbb{R}^{n+1}$ y $y \in \mathbb{R}^n$ los vectores definidos por las expresiones.

$$a = \begin{bmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_1 \\ a_0 \end{bmatrix}, \ y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N-1} \\ y_N \end{bmatrix}, \tag{3.2}$$

Ejercicio para el lector 1. Demostrar que el problema:

$$a = \operatorname{argmin}_{b\in\mathbb{R}^{n+1}} \frac{1}{2}\|Xb - y\|_2^2 \tag{3.3}$$

es equivalente a un programa cuadrático de la forma (1.5), para f determinado por una expresión de la forma (2.8).

Ejercicio para el lector 2. Desarrollar un programa Octave que resuelva el problema (3.3) aplicando un método de búsqueda lineal con direcciones de Newton o casi Newton.

Ejercicio para el lector 3. Dado un número entero $n > 0$ (determinado por el usuario). Desarrollar un programa Octave que genere conjuntos de datos $\mathcal{D} = \{(x_j, y_j)\}_{j=1}^N \subset \mathbb{R}^2$ tales que la suposición 3.1 se cumple. Aplicar el programa desarrollado como parte del ejercicio para el lector 2 para calcular el polinomio que mejor ajusta los datos en $\mathcal{D}$ en el sentido de los mínimos cuadrados.

Referencias

[1] R. L. Burden, D. J. Faires, A. M. Burden. (2017). Análisis Numérico. 10a Ed. Cengage Learning Editores.
[2] Golub, G. H., Van Loan C. F (1996). Matrix Computations (3aEd.). The Johns Hopkins University Press.
[3] Quarteroni A., Saleri F., Gervasio P. (2014). Scientific computing with MATLAB and Octave (Textbook).
[4] D. G. Luenberger, Y. Ye. (2016). Linear and Nonlinear Programming. 4a Ed. Springer International Publishing Switzerland.