

## **DESCENSO GRADIENTE**

ALAN REYES-FIGUEROA  
MÉTODOS NUMÉRICOS II

(AULA 18) 19.SEPTIEMBRE.2022

# Algoritmos para Optimización

## Algoritmos para minimización sin restricciones:

Los algoritmos para minimización sin restricciones son métodos iterativos que encuentran una solución aproximada.

Todos los algoritmos para minimización sin restricciones requieren que el usuario proporcione un punto de partida  $\mathbf{x}_0 \in \mathbb{R}^n$ . El usuario con conocimiento sobre la función o el conjunto de datos *input* puede estar en una buena posición para elegir  $\mathbf{x}_0$  como una estimación razonable de la solución.

De lo contrario, el punto inicial  $\mathbf{x}_0$  debe ser elegido por el algoritmo, ya sea mediante un enfoque sistemático o de alguna manera arbitraria (aleatorio dentro de cierto dominio).

- A partir de  $\mathbf{x}_0$ , se genera una secuencia  $\{\mathbf{x}_k\}_{k \geq 0}$  de aproximaciones.
- Para pasar de una iteración  $\mathbf{x}_k$  a la siguiente, los algoritmos usan información sobre la función  $f$  en  $\mathbf{x}_k$ , y posiblemente también información de iteraciones anteriores.
- Con esta información, se espera hallar una nueva iteración  $\mathbf{x}_{k+1}$ , usualmente con la propiedad  $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k)$ .

# Descenso Gradiente

- Sin embargo, existen algoritmos no monótonos en los que  $f$  no disminuye en cada paso, pero  $f$  debería disminuir después de algún número  $m$  de iteraciones es decir,  $f(\mathbf{x}_{k+1}) < f(\mathbf{x}_{k-j})$  para algún  $j \in \{0, 1, \dots, m\}$ .

Por ejemplo, seleccione

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \mathbf{d}_k, \quad \text{donde } \mathbf{d}_k = -\frac{\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|}$$

si

$$f(\mathbf{x}_k + \alpha \mathbf{d}_k) < \max_{0 \leq j \leq m} f(\mathbf{x}_{k-j}) + \gamma \alpha \nabla f(\mathbf{x}_k)^T \mathbf{d}_k.$$

## Framework general:

- Elegir  $\mathbf{x}_0$ ,
- Hallar o establecer un criterio de paro,
- Definir cómo actualizar  $\mathbf{x}_k$ .

# Descenso Gradiente

## ¿Cómo actualizar $\mathbf{x}_k$ ?:

La idea es elegir una dirección  $\mathbf{d}_k$  y buscar a lo largo del semirrayo en esta dirección,  $\mathbf{x}_{k+1} = \mathbf{x}_k + t\mathbf{d}_k$ , para una nueva iteración  $\mathbf{x}_{k+1}$  donde la función reduzca su valor.

## Definición

Dada  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  diferenciable, y un punto  $\mathbf{x}_k \in \mathbb{R}^n$ , una **dirección de descenso** para  $f$  en  $\mathbf{x}_k$  es cualquier vector  $\mathbf{d} \in \mathbb{R}^n$ , tal que

$$f(\mathbf{x}_k + t\mathbf{d}) < f(\mathbf{x}_k), \quad \text{para todo } t \in (0, T). \quad (1)$$

*En el contexto de optimización, una dirección de descenso en  $\mathbf{x}_k$  mueve el punto  $\mathbf{x}_k$  un poco más cerca de un mínimo local.*

Muchos de los métodos de optimización basan su estrategia en hallar una dirección de descenso, por ejemplo: el método de descenso gradiente, el método de gradiente conjugado, ...

# Descenso Gradiente

**Ejemplo:** La dirección de descenso más común para una función es  $\mathbf{u} = -\frac{\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|}$ .

Ya hemos mencionado que  $-\frac{\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|}$  indica la dirección en la cual  $f$  decrece lo más rápido posible en el punto  $\mathbf{x}_k$ . En particular, del Teorema de Taylor, tenemos

$$\begin{aligned} f(\mathbf{x}_k + t\mathbf{u}) &= f(\mathbf{x}_k) + t\nabla f(\mathbf{x}_k)^T \mathbf{u} + o(\|\mathbf{u}\|) \approx f(\mathbf{x}_k) - t\nabla f(\mathbf{x}_k)^T \frac{\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|} \\ &\approx f(\mathbf{x}_k) - t\|\nabla f(\mathbf{x}_k)\| < f(\mathbf{x}_k). \end{aligned} \quad (2)$$

Luego,  $f(\mathbf{x}_k + t\mathbf{u}) < f(\mathbf{x}_k)$ , para  $t \in (0, 1)$  y  $\mathbf{u} = -\frac{\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|}$  es una dirección de descenso.

general, lo anterior vale para cualquier vector  $\mathbf{d}$  tal que  $\nabla f(\mathbf{x}_k)^T \mathbf{d} < 0$ .

## Proposición

*Dada  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  de clase  $C^1$ , y  $\mathbf{x}_k \in \mathbb{R}^n$ . Entonces,  $\mathbf{d} \in \mathbb{R}^n$  es una dirección de descenso para  $f$  en  $\mathbf{x}_k$ , si y sólo si,  $\nabla f(\mathbf{x}_k)^T \mathbf{d} < 0$ .*

Prueba: ( $\Leftarrow$ ) Se deduce directamente de la aproximación de Taylor de  $f(\mathbf{x}_k + t\mathbf{d})$ .

# Descenso Gradiente

$$f(\mathbf{x}_k + t\mathbf{d}) = f(\mathbf{x}_k) + t\nabla f(\mathbf{x}_k)^T \mathbf{d} + o(\|\mathbf{d}\|) \approx f(\mathbf{x}_k) + \underbrace{t \nabla f(\mathbf{x}_k)^T \mathbf{d}}_{<0} < f(\mathbf{x}_k),$$

para  $t \in (0, 1)$ , y  $\mathbf{d}$  es dirección de descenso.

( $\Rightarrow$ ) Si  $\mathbf{d}$  es dirección de descenso de  $f$  en  $\mathbf{x}_k$ , entonces existe  $t_0 \in (0, T)$ , tal que  $f(\mathbf{x}_k + t_0\mathbf{d}) < f(\mathbf{x}_k)$ .

Luego, por continuidad de  $\nabla f$  y la preservación de signo, se tiene que  $\nabla f(\mathbf{x}_k + t\mathbf{d})^T \mathbf{d} < 0$ , para todo  $t \in (0, t_0)$ . Usando Taylor, existe  $h \in (0, 1)$  tal que

$$f(\mathbf{x}_k + t\mathbf{d}) = f(\mathbf{x}_k) + t\nabla f(\mathbf{x}_k + h t\mathbf{d})^T \mathbf{d}.$$

Como  $0 < ht < t < t_0$ , entonces  $\nabla f(\mathbf{x}_k + h t\mathbf{d})^T \mathbf{d} < 0$ , para todo  $h \in (0, 1)$  y por lo tanto,  $f(\mathbf{x}_k + h t\mathbf{d}) < f(\mathbf{x}_k)$ ,  $\forall ht \in (0, t)$ . Esto muestra que  $\mathbf{d}$  es una dirección de descenso.  $\square$

La estrategia anterior ya nos da un algoritmo básico de optimización.

# Descenso Gradiente

**Algoritmo:** (Descenso gradiente, versión naïve)

*Inputs:*  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  función de clase  $C^1$ ,  $\mathbf{x}_0 \in \mathbb{R}^n$ ,  $\alpha > 0$  tamaño de paso.

*Outputs:*  $\mathbf{x}$  punto crítico de  $f$ .

For  $k = 0, 1, 2, \dots$  hasta que se cumpla un criterio de paro:

    Compute  $\mathbf{d}_k$  a descent direction

    (for example, any  $\mathbf{d}_k$  such that  $\angle(-\nabla f(\mathbf{x}_k), \mathbf{d}_k) < |\frac{\pi}{2}|$ ).

    Set  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha \mathbf{d}_k$ .

Return  $\mathbf{x}_{k+1}$ .

En el caso en que  $\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$ , tenemos

**Algoritmo:** (Steepest descent, versión naïve)

*Inputs:*  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  función de clase  $C^1$ ,  $\mathbf{x}_0 \in \mathbb{R}^n$ ,  $\alpha > 0$  tamaño de paso.

*Outputs:*  $\mathbf{x}$  punto crítico de  $f$ .

For  $k = 0, 1, 2, \dots$  hasta que se cumpla un criterio de paro:

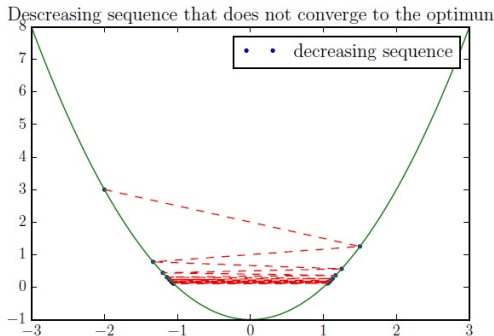
    Set  $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \nabla f(\mathbf{x}_k)$ .

Return  $\mathbf{x}_{k+1}$ .

# Descenso Gradiente

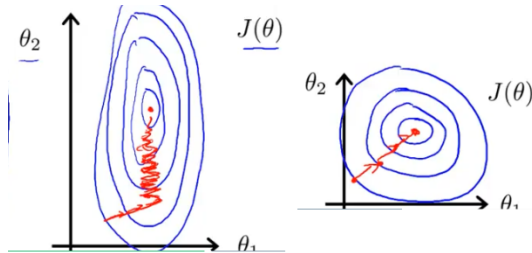
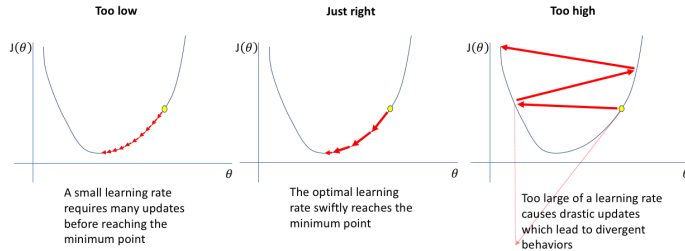
A la constante  $\alpha_k > 0$  se le llama el **tamaño de paso**. Usualmente este tamaño de paso  $\alpha_k$  cambia en cada iteración, y se elige en función de la iteración y del punto,  $\alpha_k$ . El caso más simple se da al elegir  $\alpha_k = \alpha$  constante, como en los algoritmos naïve anteriores.

Elegir el tamaño de paso adecuado es crucial. Si  $\alpha_k$  es demasiado grande, es posible que el algoritmo no detecte las regiones donde se encuentra el mínimo local.





# Descenso Gradiente



# Descenso Gradiente

**Ejemplo:** Considere la función  $f : \mathbb{R} \rightarrow \mathbb{R}, f(\mathbf{x}) = \mathbf{x}^2$ .  $f$  es diferenciable y  $\nabla f(\mathbf{x}) = 2\mathbf{x}$ .

- Tomando  $\alpha = 1$ , obtenemos la iteración de descenso máximo

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \nabla f(\mathbf{x}) = \mathbf{x}_k - 2\mathbf{x}_k = -\mathbf{x}_k,$$

la cual es una secuencia alternante  $\mathbf{x}_0, -\mathbf{x}_0, \mathbf{x}_0, -\mathbf{x}_0, \dots$ , no convergente.

- Tomando  $\alpha = \frac{1}{4}$ , obtenemos la iteración de descenso máximo

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \frac{1}{4}\nabla f(\mathbf{x}) = \mathbf{x}_k - \frac{2}{4}\mathbf{x}_k = \frac{1}{2}\mathbf{x}_k.$$

Esta es una secuencia geométrica convergente  $\mathbf{x}_0, \frac{1}{2}\mathbf{x}_0, \frac{1}{4}\mathbf{x}_0, \frac{1}{8}\mathbf{x}_0, \dots$

Una estrategia empírica muy simple, pero bastante útil, para elegir  $\alpha$  es comenzar con un valor pequeño (e.g.  $\alpha = 0.1$ ). Si con este valor de  $\alpha$  no se observa convergencia del método de descenso gradiente, se prueban valores usando una escala potencial:

- $\alpha = 0.01, \alpha = 0.001; \alpha = 0.0001, \dots$
- $\alpha = \rho^1\alpha_0, \alpha = \rho^2\alpha_0, \alpha = \rho^3\alpha_0, \dots$ , donde  $0 < \rho < 1$  (por ejemplo:  $\rho = \frac{1}{2}, \frac{1}{4}$  ó  $\rho = \frac{1}{10}$ )

# Descenso Gradiente

**Criterios de paro:** Existen muchos criterios de paro que pueden usarse para detener los algoritmos de optimización numérica.

- Error absoluto de iteraciones: Se mide el error absoluto entre dos iteraciones consecutivas

$$||\mathbf{x}_{k+1} - \mathbf{x}_k||_{norm} < tol.$$

- Error relativo de iteraciones: Se compara el error relativo entre dos iteraciones consecutivas  $\mathbf{x}_k$  y  $\mathbf{x}_{k+1}$

$$\frac{||\mathbf{x}_{k+1} - \mathbf{x}_k||_{norm}}{||\mathbf{x}_k||_{norm}} < tol.$$

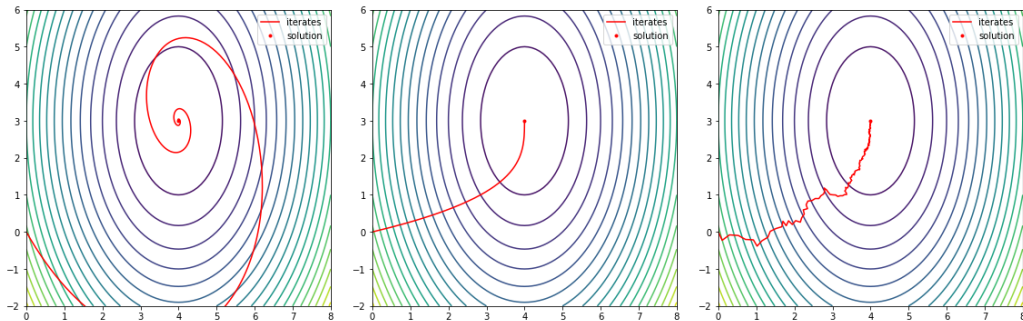
- Error abs/rel del valor de la función: Se mide el error entre dos valores de  $f(\mathbf{x}_k)$  en iteraciones consecutivas. Así

$$|f(\mathbf{x}_{k+1}) - f(\mathbf{x}_k)| < tol.$$

- Norma del gradiente: En un mínimo local, sabemos que  $\nabla f(\mathbf{x}) = \mathbf{0}$ . Se busca entonces que las normas del gradiente sean suficientemente pequeñas

$$||\nabla f(\mathbf{x}_k)||_{norm} < tol.$$

# Descenso Gradiente



Varios métodos gradiente aplicados a una función cuadrática: (a) Descenso gradiente con dirección de descenso con ángulo constante  $\varphi$  con  $\nabla f(\mathbf{x}_k)$ ; (b) Descenso máximo; (c) Descenso gradiente con dirección de descenso aleatoria.

# Descenso Gradiente

Otra estrategia más adecuada para elegir el tamaño de paso es el llamado **esquema de Cauchy**.

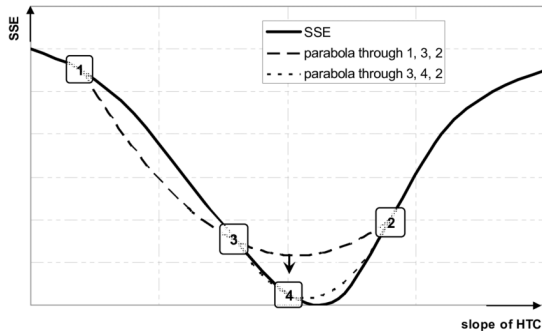
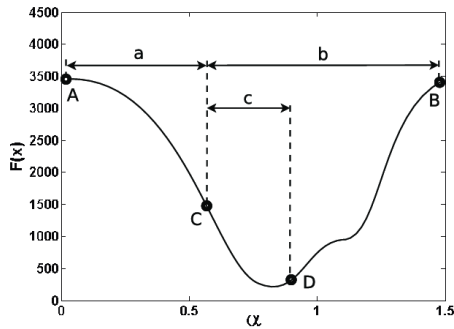
Este consiste en lo siguiente: Dado  $\mathbf{x}_k \in \mathbb{R}^n$ , luego de elegir la dirección de búsqueda  $\mathbf{d}_k$ , buscamos cuál es el valor de  $\alpha_k > 0$  que minimiza la función  $f$ , restringida a la recta  $\mathbf{x}_k + t\mathbf{d}_k$ ,  $t > 0$ . Esto es, definimos

$$\alpha_k = \operatorname{argmin}_{t \in \mathbb{R}} f(\mathbf{x}_k + t\mathbf{d}_k). \quad (3)$$

Observe que (3) corresponde a un problema de minimización 1-dimensional. Es posible aplicar aquí las técnicas de optimización que aprendieron en Métodos Numéricos I.

- Método de búsqueda de Fibonacci (*Fibonacci search*),
- Método de la razón áurea (*golden ration search*),
- Interpolación parabólica (*quadratic interpolation*),
- Método de Newton,
- ...

# Descenso Gradiente



Optimización 1-dimensional: (a) *Golden-search*, (b) interpolación parabólica.

Ver <https://web2.qatar.cmu.edu/~gdicaro/15382/additional/one-dimensional-search-methods.pdf>

# Descenso Gradiente

**Algoritmo:** (*Descenso gradiente*, versión esquema de Cauchy)

*Inputs:*  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  función de clase  $C^1$ ,  $\mathbf{x}_0 \in \mathbb{R}^n$ .

*Outputs:*  $\mathbf{x}$  punto crítico de  $f$ .

For  $k = 0, 1, 2, \dots$  hasta que se cumpla un criterio de paro:

    Define  $\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$ , or any other descent direction.

    Compute  $\alpha_k$  such that

$$\alpha_k = \operatorname{argmin}_{t \in \mathbb{R}} f(\mathbf{x}_k + t\mathbf{d}_k),$$

    by any 1-dimensional optimization method,

    Set  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ .

Return  $\mathbf{x}_{k+1}$ .

# Descenso Gradiente

Otra dirección de búsqueda importante es la **dirección de Newton**. Ésta se deriva de la aproximación de Taylor de segundo orden

$$\begin{aligned} f(\mathbf{x}_k + \mathbf{d}) &= f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T D^2 f(\mathbf{x}_k) \mathbf{d} + o(\|\mathbf{d}\|^2). \\ &\approx \underbrace{f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{d} + \frac{1}{2} \mathbf{d}^T D^2 f(\mathbf{x}_k) \mathbf{d}}_{m_k(\mathbf{d})}. \end{aligned} \quad (4)$$

Observe que  $m_k(\mathbf{d})$  es una función cuadrática en  $\mathbb{R}^n$ . Si  $D^2 f(\mathbf{x}_k)$  es positiva definida, entonces  $m_k$  es convexa, y encontramos la dirección de Newton hallando el vector  $\mathbf{d} \in \mathbb{R}^n$  como el mínimo global de esta función cuadrática. Esto es

$$\nabla m_k(\mathbf{d}) = \nabla f(\mathbf{x}_k) + D^2 f(\mathbf{x}_k) \mathbf{d} = \mathbf{0} \implies \mathbf{d}_{Newton} = -(D^2 f(\mathbf{x}_k))^{-1} \nabla f(\mathbf{x}_k).$$

- Podemos usar la dirección de Newton en un método de descenso gradiente siempre que  $D^2 f \succ \mathbf{0}$ .
- Usamos tamaño de paso  $\alpha = 1$  con la dirección de Newton. Sin embargo,  $\alpha$  puede ajustarse cuando los resultados no son satisfactorios.



# Descenso Gradiente

**Algoritmo:** (Descenso gradiente, versión Newton)

*Inputs:*  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  función de clase  $C^2$ , con Hessiana  $D^2f$  positiva definida en cada punto;  $\mathbf{x}_0 \in \mathbb{R}^n$ ,  $\alpha_k > 0$  tamaño de paso (usualmente  $\alpha_k = 1$ ).

*Outputs:*  $\mathbf{x}$  punto crítico de  $f$ .

For  $k = 0, 1, 2, \dots$  hasta que se cumpla un criterio de paro:

    Define  $\mathbf{d}_k = -(D^2f(\mathbf{x}_k))^{-1} \nabla f(\mathbf{x}_k)$ ,

    Set  $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k$ .

Return  $\mathbf{x}_{k+1}$ .

**Obs:**

- Cuando  $D^2f(\mathbf{x}_k)$  no es positiva definida en alguno de los puntos iterados  $\mathbf{x}_k$ , el método aún se puede utilizar. En este caso, se reemplaza el hessiano por su aproximación simétrica  $A \in \mathbb{R}^{n \times n}$ , más cercana, que sea positiva definida.
- Esto puede hacerse hallando la descomposición espectral  $D^2f(\mathbf{x}_k) = U\Lambda U^T$ , y reemplazando todos los autovalores negativos de  $\Lambda$  por  $\varepsilon > 0$ ;  $A = U\Lambda_\varepsilon U^T$ .

# Descenso Gradiente

- El cálculo de la hessiana  $D^2f(\mathbf{x}_k)$  en cada iteración, consume mucho costo computacional (sobre todo en altas dimensiones).

Existen otros métodos de tipo gradiente que, en lugar de calcular exactamente el hessiano  $D^2f(\mathbf{x}_k)$ , utilizan una aproximación  $B_k$ , que se actualiza en cada paso.

De la aproximación de Taylor

$$\begin{aligned}\nabla f(\mathbf{x}_k + \mathbf{d}) &= \nabla f(\mathbf{x}_k) + \int_0^1 D^2f(\mathbf{x}_k + t\mathbf{d}) \mathbf{d} dt \\ &= \nabla f(\mathbf{x}_k) + D^2f(\mathbf{x}_k) \mathbf{d} + \underbrace{\int_0^1 [D^2f(\mathbf{x}_k + t\mathbf{d}) - D^2f(\mathbf{x}_k)] \mathbf{d} dt}_{o(\|\mathbf{d}\|)}.\end{aligned}$$

Haciendo  $\mathbf{d} = \mathbf{x}_{k+1} - \mathbf{x}_k \Rightarrow \nabla f(\mathbf{x}_{k+1}) = \nabla f(\mathbf{x}_k) + D^2f(\mathbf{x}_k) (\mathbf{x}_{k+1} - \mathbf{x}_k) + o(\|\mathbf{d}\|)$ .  
Cuando  $\mathbf{x}_k, \mathbf{x}_{k+1}$  están en una región cercana al mínimo  $\mathbf{x}^*$ , donde  $D^2f(\mathbf{x}_k) \succ 0$ , resulta

$$D^2f(\mathbf{x}_k) \mathbf{d} \approx \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k). \quad (5)$$

# Descenso Gradiente

Así, elegimos la aproximación de  $B_{k+1}$  de modo que imite la propiedad (5) anterior. Así, requerimos que  $B_{k+1}$  cumpla la **ecuación secante**:

$$B_{k+1}\mathbf{s}_k = \mathbf{y}_k, \quad (6)$$

donde  $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$ , y  $\mathbf{y}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k)$ . Además, requerimos que  $B_{k+1}$  sea simétrica, y que la diferencia  $B_{k+1} - B_k$  sea de bajo rango.

Estos son los métodos llamados **métodos quasi-Newton**. Dos de las fórmulas más populares para actualizar el hessiano son

- el método **simétrico de rango 1 (SR1)**:

$$B_{k+1} = B_k + \frac{(\mathbf{y}_k - B_k\mathbf{s}_k)(\mathbf{y}_k - B_k\mathbf{s}_k)^T}{(\mathbf{y}_k - B_k\mathbf{s}_k)^T\mathbf{s}_k}.$$

- el método **BFGS (Broyden-Fletcher-Goldfarb-Shanno)**:

$$B_{k+1} = B_k - \frac{B_k\mathbf{s}_k\mathbf{s}_k^TB_k}{\mathbf{s}_k^TB_k\mathbf{s}_k} + \frac{\mathbf{y}_k\mathbf{y}_k^T}{\mathbf{y}_k^T\mathbf{s}_k}.$$