Tarea 10 - Optimización

Giovanni Gamaliel López Padilla

Índice

1. Introducción

El método del CG fue presentado por Hestenes y Stiefel[1] como un método directo, mas tarde Reid[2] y Concus[3] le descubrieron su verdadero potencial al mirarlo como un método iterativo bastante adecuado para resolver sistemas de la forma Ax = b, donde A es simétrica y definida positiva. El método de CG está basado en un principio de conjugación teniendo un almacenamiento modesto y siempre converge.

El problema de minimización a tratar será el caso de una función cuadrática, la tiene asociada una matriz simétrica y definida positiva. Esta definición se encuentra escrita en la ecuación 1.

$$f(x) = \frac{1}{2}x^T Ax - x^T b \qquad A \in \mathbb{R}^{n\times n}, b \in \mathbb{R}^n \tag{1}$$

La solución análitica de la ecuación 1 es $Ax = b$, el cual se trata de un sistema de n ecuaciones. El método de GC compromete a una clase de funciones sin restricciones. El método se caracteriza por la baja memoria que requiere para adquirir propiedades de una convergencia local o global. El método de GC genera una secuencia de pasos x_k para $k > 1$ a partir de la ecuación 2. El primer paso (x_0) siempre es dado.

$$x_{k+1} = x_k + \alpha_k d_k \tag{2}$$

donde α_k es un parámetro positivo determinado por una búsqueda en linea y d_k son las direcciones de descenso. La secuencia d_k es obtenida con la ecuación 3.

$$d_{k+1} = \beta_k d_k - g_{k+1} \qquad d_0 = -g_0 \tag{3}$$

En la ecuación 3, el término β_k es un parámetro que es obtenido por el método GC. En este trabajo nos centraremos en optimizar una función usando el método GC con β_k obtenidas a partir de los algoritmos de Fletcher-Reeves (FR), Polak-Ribiere (PR), Hestenes-Stiefel (HS) y una combinación de Fletcher-Reeves con Polak-Ribiere (FR-PR).

2. Métodos

2.1. Gradiente Conjugado

De manera general, el algortimo para el método de Gradiente Conjugado se encuentra descrito en .

Input: x_0, d_0
Output: x^*
$d_0 \leftarrow -\nabla f(x_0)$
while $||g(x_k)|| < \tau$ **do**
  $\alpha_k \leftarrow$ line search
  Calcular $\nabla f(x_{k+1})$
  Calcular β_{k+1}
  $d_{k+1} = \beta_{k+1} d_k - \nabla f(x_{k+1})$

La razón por la que α_k es calculado usando una búsqueda en linea es debido a que asi se asegura que la dirección d_{k+1} es de descenso. Esto es debido a que $d_{k+1}^T d_k = 0$, entonces

$$\begin{aligned} d_{k+1}^T d_{k+1} &= -||g_{k+1}||^2 + \beta_{k+1}^{FR} g_{k+1}^T d_k \\ &= -||g_{k+1}||^2 \\ &< 0 \end{aligned}$$

Si α_{k+1} no es un tamaño de paso exacto. El término $beta_{k+1}^{FR} g_{k+1}^T d_k$ podria dominar provocando que no se garantize el descenso.

2.2. Fletcher-Reeves

El algortimo de Fletcher-Reeves[4] se basa en encontrar una secuencia de β_k descritas en la ecuación 4.

$$\beta_{k+1}^{FR} = \frac{\nabla^T f(x_{k+1}) \nabla f(x_{k+1})}{\nabla^T f(x_k) \nabla f(x_k)} \tag{4}$$

2.3. Polak-Ribiere

El algortimo de Polak-Ribiere[5] se basa en encontrar una secuencia de β_k descritas en la ecuación 5.

$$\beta_{k+1}^{PR} = \frac{\nabla^T f(x_{k+1})(\nabla f(x_{k+1}) - \nabla f(x_k))}{\nabla^T f(x_k) \nabla f(x_k)} \tag{5}$$

Es conveniente elegir el valor máximo entre β_{k+1}^{PR} y 0. Esto es debido a que existen funciones las cuales pueden ciclarse de forma indefinida.

2.4. Hestenes-Stiefel

El algortimo de Hestenes-Stiefel[6] se basa en encontrar una secuencia de β_k descritas en la ecuación 6.

$$\beta_{k+1}^{HS} = \frac{\nabla^T f(x_{k+1})(\nabla f(x_{k+1}) - \nabla f(x_k))}{(\nabla f(x_{k+1}) - \nabla f(x_k))^T d_k} \tag{6}$$

2.5. Fletcher-Reeves Polak-Ribiere

El algortimo de Fletcher-Reeves en combinación con Polak-Ribiere[7] se basa en encontrar una secuencia de β_k descritas en la ecuación 7.

$$\beta_{k+1}^{FR-PR} = \begin{cases} -\beta_k^{FR} & \text{si } \beta_k^{PF} < -\beta_k^{FR} \\ \beta_k^{PR} & \text{si } |\beta_k^{PF}| \leq \beta_k^{FR} \\ \beta_k^{FR} & \text{si } \beta_k^{PF} > \beta_k^{FR} \end{cases} \tag{7}$$

La secuencia β_{k+1}^{FR-PR} garantiza la convergencia, debido a que $\beta_k \leq \beta_k^{FR}$ para $k > 2$.

3. Referencias

[1] Hestenes MR, Stiefel E. Methods of conjugate gradients for solving linear systems. J RES NATL BUR STAN. 1952 dec;49(6):409. Available from: https://doi.org/10.6028%2Fjres.049.044.

[2] Reid JK. The Use of Conjugate Gradients for Systems of Linear Equations Possessing "Property A". SIAM Journal on Numerical Analysis. 1972;9(2):325–332. Available from: http://www.jstor.org/stable/2156406.

[3] Concus P, Golub GH, O'Leary DP. A generalized conjugate gradient method for the numerical solution for elliptic partial differential equations. In: Sparse Matrix Computations. Elsevier; 1976. p. 309–332. Available from: https://doi.org/10.1016%2Fb978-0-12-141050-6.50023-4.

[4] Fletcher R, Reeves CM. Function minimization by conjugate gradients. The computer journal. 1964;7(2):149–154.

[5] Polak E, Ribiere G. Note sur la convergence de méthodes de directions conjuguées. ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique. 1969;3(R1):35–43.

[6] Hestenes MR. Conjugate direction methods in optimization. In: Optimization Techniques Part 1. Springer; 1978. p. 8–27.

[7] Babaie-Kafaki S, Ghanbari R. A hybridization of the Polak-Ribière-Polyak and Fletcher-Reeves conjugate gradient methods. Numer Algor. 2014 may;68(3):481–495. Available from: https://doi.org/10.1007%2Fs11075-014-9856-6.