Otimização

INF1608 – Análise Numérica

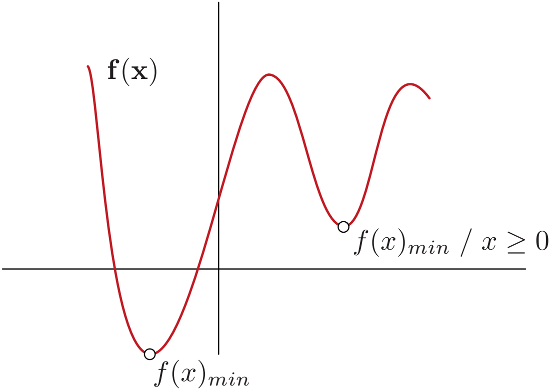
Waldemar Celes
celes@inf.puc-rio.br

Departamento de Informática, PUC-Rio

Problema de otimização

Achar o *mínimo* (ou o *máximo*) de uma **função objetiva**

- Pode ou não estar submetido a restrições



- Note que achar $\max f(x)$ equivale a achar $\min -f(x)$

Nosso foco

Otimização sem restrição

- Métodos que não fazem uso de gradientes
  - Empregados quando gradientes não estão disponíveis
- Métodos que fazem uso do gradiente da função ($\nabla f(x)$)
  - Tendem a convergir mais rápido

Otimização sem gradientes

Método da Seção Áurea

Determinar $\min f(x)$ no intervalo $[a, b]$, onde $f(x)$ é unimodal

Otimização sem gradientes

Método da Seção Áurea

Determinar min $f(x)$ no intervalo $[a, b]$, onde $f(x)$ é unimodal

- Inspirado no método da bisseção para determinação de raízes
- Parte-se de dois valores x_1 e x_2

$$a < x_1 < x_2 < b$$

Otimização sem gradientes

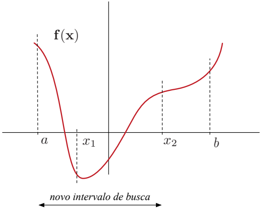
Método da Seção Áurea

Determinar min $f(x)$ no intervalo $[a, b]$, onde $f(x)$ é unimodal

- Inspirado no método da bisseção para determinação de raízes
- Parte-se de dois valores x_1 e x_2

$$a < x_1 < x_2 < b$$

- Se $f(x_1) \leq f(x_2)$, ajusta intervalo para $[a, x_2]$
- Senão, ajusta intervalo para $[x_1, b]$



Método da Seção Áurea

Como escolher os valores x_1 e x_2?

- Assumindo $[a, b] = [0, 1]$

Método da Seção Áurea

Como escolher os valores x_1 e x_2?

- Assumindo $[a, b] = [0, 1]$

Intuitivamente:

- Serem simétricos, já que não temos conhecimento de $f(x)$

$$x_1 = 1 - x_2$$

Método da Seção Áurea

Como escolher os valores x_1 e x_2?

- Assumindo $[a, b] = [0, 1]$

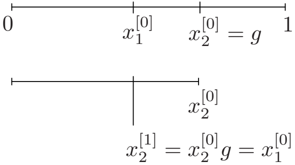
Intuitivamente:

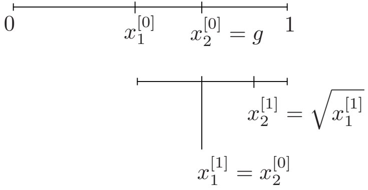
- Serem simétricos, já que não temos conhecimento de $f(x)$

$$x_1 = 1 - x_2$$

- Aproveitar $f(x_i)$ na próxima iteração

$$x_1 = x_2^2$$





Método da Seção Áurea

Combinando os dois critérios intuitivos

$$\begin{cases} x_1 = 1 - x_2 \\ x_1 = x_2^2 \end{cases}$$

$$x_2^2 + x_2 - 1 = 0$$

- Raiz positiva
  - Razão de ouro

$$g = \frac{\sqrt{5} - 1}{2} \approx 0.618$$

Método da Seção Áurea

Intervalo $[0, 1]$

- $x_1 = 1 - g$
- $x_2 = g$

Método da Seção Áurea

Intervalo $[0, 1]$

- $x_1 = 1 - g$
- $x_2 = g$

Generalizando: intervalo $[a, b]$

- $x_1 = a + (1 - g)(b - a)$
- $x_2 = a + g(b - a)$

Método da Seção Áurea

Redução do intevalo de busca por iteração:

$$l_1 = g\ l_0, \quad \text{onde} \quad l_0 = b - a$$

Método da Seção Áurea

Redução do intevalo de busca por iteração:

$$l_1 = g\ l_0, \quad \text{onde} \quad l_0 = b - a$$

- Após k iterações: $$l_k = g^k(b - a)$$

Método da Seção Áurea

Redução do intevalo de busca por iteração:

$$l_1 = g\ l_0, \quad \text{onde} \quad l_0 = b - a$$

- Após k iterações:

$$l_k = g^k (b - a)$$

- Erro da solução

$$E = \frac{g^k (b - a)}{2}$$

  - Solução representada pelo meio do intervalo após k iterações

Método da Seção Áurea

Redução do intevalo de busca por iteração:

$$l_1 = g\ l_0, \quad \text{onde} \quad l_0 = b - a$$

- Após k iterações:

$$l_k = g^k(b-a)$$

- Erro da solução

$$E = \frac{g^k(b-a)}{2}$$

  - Solução representada pelo meio do intervalo após k iterações

Convergência garantida, linear

Método da Seção Áurea

Redução do intevalo de busca por iteração:

$$l_1 = g\ l_0, \quad \text{onde} \quad l_0 = b - a$$

- Após k iterações:

$$l_k = g^k(b - a)$$

- Erro da solução

$$E = \frac{g^k(b - a)}{2}$$

  - Solução representada pelo meio do intervalo após k iterações

Convergência garantida, linear

- Determinação de raízes via método da bisseção: $\gamma = 0.5$
- Otimização via método da seção áurea: $\gamma = 0.618$
  - Convergência mais lenta que bisseção

Método da Seção Áurea

Dada f unimodal com mínimo em $[a, b]$

$$g = \frac{\sqrt{5} - 1}{2}$$

for $k = 1, 2, 3, \cdots$

  if $f(a + (1 - g)(b - a) < f(a + g(b - a))$

    $b = a + g(b - a)$

  else

    $a = a + (1 - g)(b - a)$

  end

end

return $\dfrac{a + b}{2}$

Método da Seção Áurea

Dada f unimodal com mínimo em $[a, b]$

$$
\begin{array}{l}
g = \dfrac{\sqrt{5}-1}{2} \\
\textbf{for } k = 1, 2, 3, \cdots \\
\quad \textbf{if } f(a + (1-g)(b-a) < f(a + g(b-a)) \\
\quad\quad b = a + g(b-a) \\
\quad \textbf{else} \\
\quad\quad a = a + (1-g)(b-a) \\
\quad \textbf{end} \\
\textbf{end} \\
\textbf{return } \dfrac{a+b}{2}
\end{array}
$$

- Implementação eficiente avalia a função uma vez por iteração

Interpolação Parabólica Sucessiva

Método busca usar os valores de $f(x)$ avaliados nas iterações

- Método da seção áurea só compara valores

Interpolação Parabólica Sucessiva

Método busca usar os valores de $f(x)$ avaliados nas iterações

- Método da seção áurea só compara valores

Ideia

- Criar modelo local e assumir como função objetivo
  - Similar aos métodos *Secante* e *IQI* para determinação de raízes

Modelo local: parábola

- Inicia com 3 estimativas próximas ao mínimo: r, s, t
- Calcula parábola interpolante (por diferenças divididas)

Interpolação Parabólica Sucessiva

Determinação da parábola

$$\begin{array}{c|ccc}
r & f(r) & & \\
 & & d_1 & \\
s & f(s) & & d_3 \\
 & & d_2 & \\
t & f(t) & &
\end{array}$$

$$d_1 = \frac{f(s) - f(r)}{s - r},\ d_2 = \frac{f(t) - f(s)}{t - s},\ d_3 = \frac{d_2 - d_1}{t - r}$$

$$P(x) = f(r) + d_1(x - r) + d_3(x - r)(x - s)$$

Interpolação Parabólica Sucessiva

P(x) = f(r) + d1(x-r) + d3(x-r)(x-s)

Para achar o mínimo, faz-se $P'(x) = 0$, chegando a:

$$x = \frac{r+s}{2} - \frac{(f(s)-f(r))(t-r)(t-s)}{2[(s-r)(f(t)-f(s))-(f(s)-f(r))(t-s)]}$$

- x substitui o menos recente ou a pior estimativa entre r, s e t.

Interpolação Parabólica Sucessiva

P(x) = f(r) + d1(x-r) + d3(x-r)(x-s)

Para achar o mínimo, faz-se $P'(x) = 0$, chegando a:

$$x = \frac{r+s}{2} - \frac{(f(s)-f(r))(t-r)(t-s)}{2[(s-r)(f(t)-f(s))-(f(s)-f(r))(t-s)]}$$

- x substitui o menos recente ou a pior estimativa entre r, s e t.

Características:

- Convergência não garantida
- Tende a ser mais rápido que seção áurea

Interpolação Parabólica Sucessiva

P(x) = f(r) + d1(x-r) + d3(x-r)(x-s)

Para achar o mínimo, faz-se $P'(x) = 0$, chegando a:

$$x = \frac{r+s}{2} - \frac{(f(s)-f(r))(t-r)(t-s)}{2[(s-r)(f(t)-f(s)) - (f(s)-f(r))(t-s)]}$$

- x substitui o menos recente ou a pior estimativa entre r, s e t.

Características:

- Convergência não garantida
- Tende a ser mais rápido que seção áurea

Nota:

- Como a função é localmente horizontal perto do mínimo, é comum $f(x_k) = f(x_{k+1})$ dentro da precisão de *double*, quando x_k se aproxima da solução: *Erro* $\approx 10^{-7}$.

Interpolação Parabólica Sucessiva

Algoritmo: com x substituindo a estimativa menos recente

- Dadas 3 estimativas iniciais: r, s e t

for $= 1, 2, 3, \cdots$

$$x = \frac{r+s}{2} - \frac{(f(s)-f(r))(t-r)(t-s)}{2[(s-r)(f(t)-f(s))-(f(s)-f(r))(t-s)]}$$

$$r = s$$

$$s = t$$

$$t = x$$

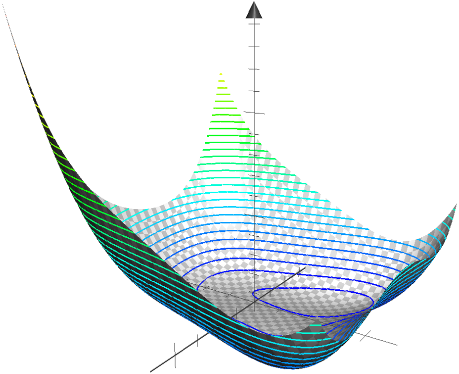
end

Método Nelder-Mead

(*Downhill simplex method*)

Determinação de valor mínimo de função com múltiplas variáveis

$$\min f(\mathbf{x}), \quad \text{onde } \mathbf{x} \text{ é vetor de dimensão } n$$



Método Nelder-Mead

- Estimativas iniciais de $n+1$ vetores

$$\mathbf{x}_0, \mathbf{x}_1 \ldots \mathbf{x}_n$$

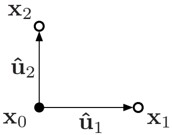
Método Nelder-Mead

- Estimativas iniciais de $n+1$ vetores

$$\mathbf{x}_0, \mathbf{x}_1 \ldots \mathbf{x}_n$$

- Estratégia simples para obtenção de $n+1$ estimativas iniciais
  - $\mathbf{x}_0$ estimativa inicial
  - $\mathbf{x}_i = \mathbf{x}_0 + \delta \hat{\mathbf{u}}_i$

Espaço 2D:



$$\hat{\mathbf{u}}_1 = \{1, 0\}$$
$$\hat{\mathbf{u}}_2 = \{0, 1\}$$

Método Nelder-Mead

Dadas as estimativas $\mathbf{x}_i$, com $i = 0, 1, \cdots, n$

1. Ordena estimativas: $f_i = f(\mathbf{x}_i)$

$$f_0 < f_1 < \cdots < f_n$$

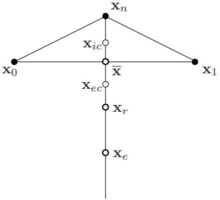
2. Remove $\mathbf{x}_n$ do conjunto de estimativas
   - Determina centróide de pontos restantes: $\bar{\mathbf{x}}$
3. **Reflete** $\mathbf{x}_n$ em relação a centróide $\bar{\mathbf{x}}$

$$\mathbf{x}_r = 2\bar{\mathbf{x}} - \mathbf{x}_n$$

   - Substitui $\mathbf{x}_n$

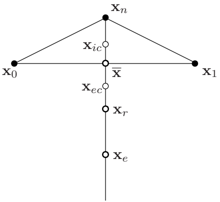
Método Nelder-Mead

- Dadas as estimativas ordenadas, substitui a pior
  - Se $f_0 < f_r < f_{n-1}$
    - Substitui $\mathbf{x}_n$ por $\mathbf{x}_r$

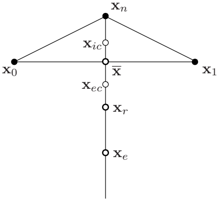


Método Nelder-Mead

- Dadas as estimativas ordenadas, substitui a pior
  - Se $f_0 < f_r < f_{n-1}$
    - Substitui $\mathbf{x}_n$ por $\mathbf{x}_r$
  - Senão, se $f_r < f_0$, faz **expansão**:
    - $\mathbf{x}_e = 3\bar{\mathbf{x}} - 2\mathbf{x}_n$
    - Substitui $\mathbf{x}_n$ pelo menor $f(\mathbf{x})$ entre $\mathbf{x}_r$ e $\mathbf{x}_e$

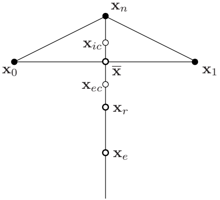


Método Nelder-Mead



- Dadas as estimativas ordenadas, substitui a pior
  - Se $f_0 < f_r < f_{n-1}$
    - Substitui $\mathbf{x}_n$ por $\mathbf{x}_r$
  - Senão, se $f_r < f_0$, faz **expansão**:
    - $\mathbf{x}_e = 3\bar{\mathbf{x}} - 2\mathbf{x}_n$
    - Substitui $\mathbf{x}_n$ pelo menor $f(\mathbf{x})$ entre $\mathbf{x}_r$ e $\mathbf{x}_e$
  - Senão, se $fr > f_{n-1}$, faz **contração** (interna ou externa):
    - $\mathbf{x}_{ic} = 0.5\bar{\mathbf{x}} + 0.5\mathbf{x}_n$
    - $\mathbf{x}_{ec} = 1.5\bar{\mathbf{x}} - 0.5\mathbf{x}_n$
    - Substitui $\mathbf{x}_n$ pelo menor $f(\mathbf{x})$ entre $\mathbf{x}_r$, $\mathbf{x}_{ic}$ e $\mathbf{x}_{ec}$

Método Nelder-Mead



- Dadas as estimativas ordenadas, substitui a pior
  - Se $f_0 < f_r < f_{n-1}$
    - Substitui $\mathbf{x}_n$ por $\mathbf{x}_r$
  - Senão, se $f_r < f_0$, faz **expansão**:
    - $\mathbf{x}_e = 3\bar{\mathbf{x}} - 2\mathbf{x}_n$
    - Substitui $\mathbf{x}_n$ pelo menor $f(\mathbf{x})$ entre $\mathbf{x}_r$ e $\mathbf{x}_e$
  - Senão, se $fr > f_{n-1}$, faz **contração** (interna ou externa):
    - $\mathbf{x}_{ic} = 0.5\bar{\mathbf{x}} + 0.5\mathbf{x}_n$
    - $\mathbf{x}_{ec} = 1.5\bar{\mathbf{x}} - 0.5\mathbf{x}_n$
    - Substitui $\mathbf{x}_n$ pelo menor $f(\mathbf{x})$ entre $\mathbf{x}_r$, $\mathbf{x}_{ic}$ e $\mathbf{x}_{ec}$
  - Se ainda não melhorar, faz **encolhimento** em torno de $\mathbf{x}_0$
    - Altera todos os pontos com exceção do $\mathbf{x}_0$
    - $\mathbf{x}_i = \mathbf{x}_0 + 0.5(\mathbf{x}_i - \mathbf{x}_0)$
- Repete processo até convergência
  - Ou até alcançar um número máximo de iterações

Método Nelder-Mead

Critério de parada:

- Número fixo de iterações: k_{max}
- Dentro de uma tolerância:
  - Redução da amplitude da função

$$\|f_0 - f_n\| < \epsilon$$

  - Redução do desvio padrão dos valores da função

$$v_i = f(\mathbf{x}_i)$$

$$\sigma = \sqrt{\frac{\sum_{i=0}^{n}(v_i - \overline{v})^2}{n}}$$

  - Redução do "volume" do simplexo
    - Em 2D, área do triângulo: $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2$

$$Vol = \left| \frac{1}{n!} det \begin{bmatrix} \mathbf{x}_1 - \mathbf{x}_0 & \mathbf{x}_2 - \mathbf{x}_0 & \cdots & \mathbf{x}_n - \mathbf{x}_0 \end{bmatrix} \right|$$

Otimização sem restrições **com gradiente**

Método de Newton

- Mínimo da função $f(x)$ ocorre em $f'(x) = 0$
  - Aplica Newton-Raphson para determinação de raízes em $f'(x)$

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$$

Método de Newton

- Funções com múltiplas variáveis: $f(\mathbf{x})$

- Gradiente

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$

- Matriz Hessian

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \ddots & \\ \vdots & & \end{bmatrix}$$

Método de Newton

- Funções com múltiplas variáveis: $f(\mathbf{x})$

- Gradiente

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{bmatrix}$$

- Matriz Hessian

$$H = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \ddots & \\ \vdots & & \end{bmatrix}$$

Iterações de Newton-Raphson

$$\begin{cases} H(\mathbf{x}_k)\mathbf{v} = -\nabla f(\mathbf{x}_k) \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{v} \end{cases}$$

- Se H estiver disponível, esse é o método geralmente preferível

Método do Máximo Declive

(Método do Gradiente)

Avançar sempre na direção contrária ao gradiente

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

- Como determinar α?

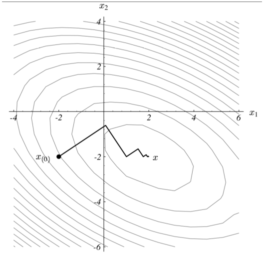
Método do Máximo Declive

(Método do Gradiente)

Avançar sempre na direção contrária ao gradiente

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

- Como determinar α?
  - Busca valor mínimo de $f(\mathbf{x})$ na direção $-\nabla f(\mathbf{x})$
  - Emprega um método de uma variável
    - Por exemplo, Interpolação Parabólica Sucessiva



Método do Máximo Declive

Algoritmo

- Dada estimativa inicial $\mathbf{x}_0$

$$
\begin{aligned}
&\textbf{for } k = 0, 1, 2, 3, \cdots \\
&\qquad \mathbf{v} = \nabla f(\mathbf{x}_k) \\
&\qquad \text{Minimize} \quad f(\mathbf{x} - \alpha \mathbf{v}) \quad \text{for scalar} \quad \alpha = \alpha^* \\
&\qquad \mathbf{x}_{k+1} = \mathbf{x}_k - \alpha^* \mathbf{v}
\end{aligned}
$$

Figura extraída de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

Método do Máximo Declive

Algoritmo

- Dada estimativa inicial $\mathbf{x}_0$

$$
\begin{aligned}
&\textbf{for } k = 0, 1, 2, 3, \cdots \\
&\quad \mathbf{v} = \nabla f(\mathbf{x}_k) \\
&\quad \text{Minimize} \quad f(\mathbf{x} - \alpha\mathbf{v}) \quad \text{for scalar} \quad \alpha = \alpha^* \\
&\quad \mathbf{x}_{k+1} = \mathbf{x}_k - \alpha^* \mathbf{v}
\end{aligned}
$$

- Avanço em zig-zag, pouco eficiente

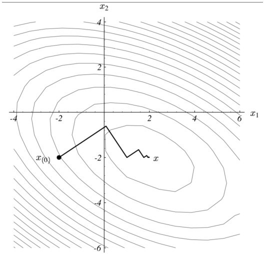


Figura extraída de "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain", Shewchuk, 1994

Busca por Gradiente Conjugado

Método dos Gradientes Conjugados para Sistemas Lineares

- Resolver

$$A\mathbf{x} = \mathbf{b}$$

- Equivale a minimizar a forma quadrática

$$f(x) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x} + c$$

Busca por Gradiente Conjugado

Método dos Gradientes Conjugados para Sistemas Lineares

- Resolver

$$A\mathbf{x} = \mathbf{b}$$

- Equivale a minimizar a forma quadrática

$$f(x) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x} + c$$

Observação

- Gradiente

$$\nabla f(\mathbf{x}) = A\mathbf{x} - \mathbf{b}$$

Busca por Gradiente Conjugado

Método dos Gradientes Conjugados para Sistemas Lineares

- Resolver

$$A\mathbf{x} = \mathbf{b}$$

- Equivale a minimizar a forma quadrática

$$f(x) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x} + c$$

Observação

- Gradiente

$$\nabla f(\mathbf{x}) = A\mathbf{x} - \mathbf{b}$$

- Resíduo expresso pelo gradiente

$$\mathbf{r} = \mathbf{b} - A\mathbf{x} = -\nabla f(\mathbf{x})$$

Busca por Gradiente Conjugado

Generalização de gradientes conjugados para qualquer $f(\mathbf{x})$

- $\mathbf{d}_k = -\nabla f(\mathbf{x})$
- $\alpha_k = \alpha$ que minimiza $f(\mathbf{x}_k + \alpha \mathbf{d}_k)$

Busca por Gradiente Conjugado

Generalização de gradientes conjugados para qualquer $f(\mathbf{x})$

- $\mathbf{d}_k = -\nabla f(\mathbf{x})$
- $\alpha_k = \alpha$ que minimiza $f(\mathbf{x}_k + \alpha \mathbf{d}_k)$

Algoritmo

$$
\begin{array}{l}
\mathbf{x}_0 = \text{estimativa inicial} \\
\mathbf{d}_0 = \mathbf{r}_0 = -\nabla f(\mathbf{x}_0) /* \; CG : \mathbf{d}_0 = \mathbf{r}_0 = \mathbf{b} - A\mathbf{x} */ \\
\textbf{for } k = 0, 1, \cdots \\
\quad \textbf{if } ||\mathbf{r}_k||_2 < tol \\
\qquad \textbf{return} \\
\quad \alpha_k = \alpha \quad \text{que minimiza} \quad f(\mathbf{x}_k + \alpha \mathbf{d}_k) \\
\quad \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \\
\quad \mathbf{r}_{k+1} = -\nabla f(\mathbf{x}_{k+1}) \\
\quad \beta_k = \dfrac{\mathbf{r}_{k+1}^T \mathbf{r}_{k+1}}{\mathbf{r}_k^T \mathbf{r}_k} \\
\quad \mathbf{d}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{d}_k
\end{array}
$$