Optimização

Aula 3

ISCTE - IUL, Mestrado Matemática Financeira

Diana Aldea Mendes, Gab 207 AA, http://iscte.pt/~deam

diana.mendes@iscte.pt

11 de Janeiro de 2010

Optimização multi-dimensional

- Métodos gradiente (gradiente descendente, gradiente conjugado)
- Métodos de Newton e quase-Newton
- Método de Powell (non-gradiente search)
- Método dos mínimos quadrados (não-linear) (Levenberg-Marquardt)

Optimização multi-dimensional

Nota-se que:

- Os computadores não podem encontrar todas as soluções
- Os computadores não podem provar a ausência de soluções
- Os computadores não podem provar a existência de soluções
- Os computadores não podem provar a unicidade de uma solução
- http://www2.imm.dtu.dk/~hbn/Software/ (Software by Hans Bruun Nielsen)

Optimização multi-dimensional

$$
\begin{aligned}
\min_{\mathbf{x}\in D} f(\mathbf{x}) \quad &\longleftrightarrow \quad \nabla f(\mathbf{x}) = \mathbf{0},\ \nabla^2 f(\mathbf{x}) \text{ definida positiva} \\
\mathbf{x} \quad &\in \quad \mathbb{R}^n,\ \ f : D \subset \mathbb{R}^n \to \mathbb{R} && (1) \\
& \quad\ \ \textit{onde} && (2) \\
& \quad\ \ f(\mathbf{x}) \text{ função objectivo} && (3) \\
\nabla f(\mathbf{x}) \quad &= \quad \left[\frac{\partial f}{\partial x_i}\right]_{i=1,\dots,n} \text{ vector gradiente} && (4) \\
& \quad\ \ \text{(derivadas parciais de primeira ordem)} && (5) \\
\nabla^2 f(\mathbf{x}) \quad &= \quad H(x) = \left[\frac{\partial^2 f}{\partial x_i \partial x_j}\right]_{i,j=1,\dots,n} \text{ matriz Hessiana} && (6) \\
& \quad\ \ \text{(derivadas parciais de segunda ordem)} && (7)
\end{aligned}
$$

- A formulação do problema de mínimo é muito simples, mas, apesar disto, não existe **um algoritmo perfeito**.
- A arte de minimizar funções não-lineares consta em saber qual é o método à utilizar e como determinar se o método escolhido converge para a solução correcta.
- Em geral, utilizamos métodos iterativos que calculam uma sequência de pontos $x_0, x_1, ... \in \mathbb{R}^n$ tal que $f(x_{k+1}) < f(x_k)$.
- O algoritmo termina quando $\nabla f(x_k) < \varepsilon$ para ε pré-definido

Optimização multi-dimensional

- Classificação dos métodos de optimização não-condicionada multidimensional
  - **Métodos analíticos:** resolver $\nabla f(\mathbf{x}) = \mathbf{0}$ analiticamente
  - **Métodos directos:** precisam só a computação de $f(x)$ em cada passo (**Powell ou DogLeg, Nelder-Mead**)
  - **Métodos gradiente:** requerem a computação de $\nabla f(\mathbf{x})$ e $f(\mathbf{x})$ em cada passo (**gradiente descendente, gradiente conjugado**)
  - **Métodos de segunda ordem:** requerem a computação de $\nabla^2 f(\mathbf{x})$, $\nabla f(\mathbf{x})$ e $f(\mathbf{x})$ em cada passo (**Newton e quase-Newton**)
  - Outros métodos: **arrefecimento simulado, algoritmos genéticos**, etc...

Método de Powell

- Requer apenas o cálculo de $f(x)$. Necessita de um grande número de iterações, o que aumenta o custo computacional
- **Ideia básica**: seja x_0 o ponto que aproxima inicialmente o minimizante de $f(x)$. O ponto x_1, obtem-se procurando sucessivamente um minimizante de f ao longo de cada um dos vectores da base canónica $\{e_1, ..., e_n\}$ de $\mathbb{R}^n$. Este processo gera a sequência de pontos

$$x_0 = P_0, P_1, ..., P_N = x_1.$$

- Ao longo de cada um dos vectores da base canónica, a função f só tem uma variável. Assim, minimizar f signifique aplicar métodos de optimização unidimensional sobre intervalos onde a função é unimodal.
- O ponto x_1 é determinado como sendo o ponto em qual o mínimo de f ocorre ao longo do vector $P_N - P_0$. Como o vector $P_N - P_0$ é considerado ser uma direcção de investigação viável, vai substituir um dos vectores direcção para o próximo iterado.

Método de Powell

Algoritmo

1. $P_0 = x_i$

2. Para $k = 1, ..., n$ encontre o valor de γ_k que minimiza $f\left(P_0 + \gamma_k U_k\right)$ e escolha

$$P_k = P_{k-1} + \gamma_k U_k$$

3. $i = i + 1$

4. $U_j = U_{j+1}$ para $j = 1, ..., n-1$. Seja

$$U_n = P_n - P_0$$

5. Encontre o valor γ que minimiza $f\left(P_0 + \gamma U_n\right)$. Escolha

$$x_i = P_0 + \gamma U_n$$

6. Repetir os passos (1)-(5).

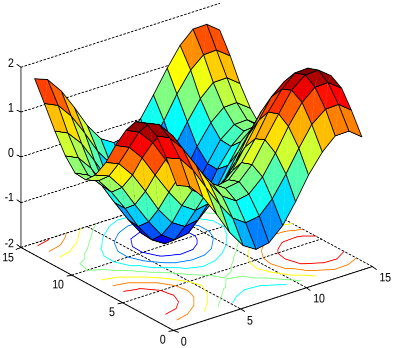
onde

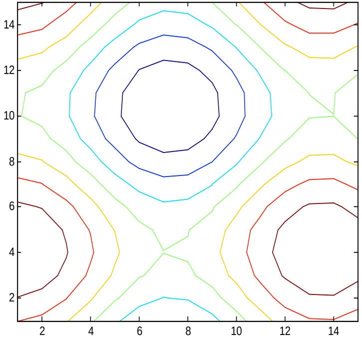
$$U = [U_1^T ... U_n^T] = [e_1^T ... e_n^T].$$

Método de Powell

Example

Seja $f(x,y) = \cos x + \sin y$ e seja $x_0 = (5.5, 2)$. Determine os pontos x_1 e x_2.





Método de Powell

- Seja
$$U = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$
e $P_0 = x_0 = (5.5, 2)$. Para $i = 1$, a função $f(P_0 + \gamma_1 U_1) = f((5.5, 2) + \gamma_1 (1, 0)) = \cos(5.5 + \gamma_1) + \sin 2$ tem um mínimo em $\gamma_1 = -2.3584$. Logo $P_1 = (3.1415, 2)$.
- Para $i = 2$, a função $f(P_1 + \gamma_2 U_2) = f((3.1415, 2) + \gamma_2 (0, 1)) = \cos(3.1415) + \sin(2 + \gamma_2)$ tem um mínimo em $\gamma_2 = 2.7123$. Logo $P_2 = (3.1415, 4.7123)$. Seja agora
$$U_2^T = (P_2 - P_0)^T \text{ e } U = \begin{bmatrix} 0 & -2.3584 \\ 1 & 2.7123 \end{bmatrix}.$$

Método de Powell

- A função

$$\begin{aligned} f\left(P_0+\gamma U_2\right) &= f\left((5.5,2)+\gamma\left(-2.3584,2.7123\right)\right) \\ &= \cos\left(5.5-2.3584\gamma\right)+\sin\left(2+2.7123\gamma\right) \end{aligned}$$

  tem um mínimo em $\gamma = 0.9816$. Logo $x_1 = (3.1848, 4.6626)$.
- Seja $P_0 = x_1$. Quando $i = 1$, a função

$$\begin{aligned} f\left(P_0+\gamma_1 U_1\right) &= f\left((3.1848,4.6626)+\gamma_1\left(0,1\right)\right) \\ &= \cos\left(3.1848\right)+\sin\left(4.6626+\gamma_1\right) \end{aligned}$$

  tem um mínimo em $\gamma_1 = 0.0497$. Logo $P_1 = (3.1484, 4.7123)$.
- Quando $i = 2$, a função

$$\begin{aligned} f\left(P_1+\gamma_2 U_2\right) &= f\left((3.1848,4.7123)+\gamma_2\left(-2.3584,2.7123\right)\right) \\ &= \cos\left(3.1848-2.3584\gamma_2\right)+\sin\left(4.7123+2.7123\gamma_2\right) \end{aligned}$$

  tem um mínimo em $\gamma_2 = 0.0078$. Logo $P_2 = (3.1662, 4.7337)$.

Método de Powell

- Seja
$$U_2^T = (P_2 - P_0)^T \text{ e } U = \begin{bmatrix} -2.3584 & -0.0185 \\ 2.7123 & 0.0710 \end{bmatrix}.$$
A função
$$\begin{aligned} f(P_0 + \gamma U_2) &= f((3.1848, 4.6626) + \gamma(-0.0185, 0.0710)) \\ &= \cos(3.1848 - 0.0185\gamma) + \sin(4.6626 + 0.0710\gamma) \end{aligned}$$
tem um mínimo em $\gamma = 0.8035$. Logo $x_2 = (3.1698, 4.7197)$.
- A função $f(x, y) = \cos x + \sin y$ tem um mínimo no ponto $P = (\pi, 3\pi/2)$.

Métodos gradiente ou métodos descendentes

Minimização ao longo de uma direcção - Problema geral

- Minimizar uma função regular f, onde o seu gradiente $g = \nabla f$ é conhecido e é dada uma direcção descendente u.
- Uma direcção u diz-se **descendente** para a função f em x se $u^T f'(x) < 0$.
- Consideram-se iterações de forma

$$x_{k+1} = x_k + \lambda_k u_k,$$

  onde u_k é a direcção descendente investigada e λ_k é o comprimento de passo obtido por uma procura unidimensional (faz decrescer a função f).
- Em todos os métodos consideremos que $f(x_{k+1}) < f(x_k)$, (*descending condition*) o que obriga a convergência para um ponto de mínimo, caso existe. O algoritmo termina quando satisfaz o critério de paragem.
- Tem convergência linear e portanto é um algoritmo bastante lento.

Métodos gradiente ou métodos descendentes

Problema: Dada uma função $f(x)$, $x = (x_1, ..., x_n) \in \mathbb{R}^n$, um ponto inicial $x_0 \in \mathbb{R}^n$ e uma direcção $u = (u_1, ..., u_n) \in \mathbb{R}^n$, como minimizar a função f ao longo da direcção u?

- Começamos com um ponto inicial x_0 e procuramos um novo ponto

  $$x = x_0 + \lambda^* u$$

  tal que a função

  $$F(\lambda) = f(x_0 + \lambda u)$$

  é minimizada quando $\lambda = \lambda^*$. Nesta situação $F(\lambda)$ é uma função só de uma variável (λ).

Métodos gradiente ou métodos descendentes

- **Como escolher a direcção descendente:**
- Nos métodos do gradiente conjugado não-linear, a direcção de investigação é de forma

$$u_k = \nabla f\left(x_k\right) + \beta_k u_{k-1}$$

onde o escalar β_k é escolhido de tal maneira que o método reduz-se ao método linear dos gradientes conjugados, quando a função é quadrática e a linha de investigação é exacta.

- Uma outra classe de métodos define a linha (direcção) de investigação por

$$u_k = -B_k^{-1} g_k$$

onde B_k é uma matriz regular ($|B_k| \neq 0$) e simétrica ($b_{ij} = b_{ji}$).

Métodos gradiente ou métodos descendentes

- Casos especiais são dados quando

$$\begin{aligned} B_k &= I \text{ the steepest descent method} \\ B_k &= \nabla^2 f(x_k) = H(f(x_k)) \quad \text{o método de Newton.} \end{aligned}$$

- Os métodos quase-Newton ou métricos variáveis são também deste tipo, mas neste caso B_k depende ainda de B_{k-1} e de x_{k-1}.
- **Matlab**: m-file **opt_steep.m**

[xo,fo] = opt_steep (f, x0, TolX, TolFun, alpha0, MaxIter)

Métodos gradiente ou métodos descendentes

Minimização ao longo de uma direcção - exemplo

Example

Seja $f(x) = f(x_1, x_2) = 1 + x_1 - x_2 + x_1^2 + 2x_2^2$. Minimize f na direcção $u = (-2, 1)$ para o ponto inicial $x_0 = (0, 0)$.

Solution

Definimos a linha de investigação

$$x = x_0 + \lambda u = (0, 0) + \lambda(-2, 1) = (-2\lambda, \lambda)$$

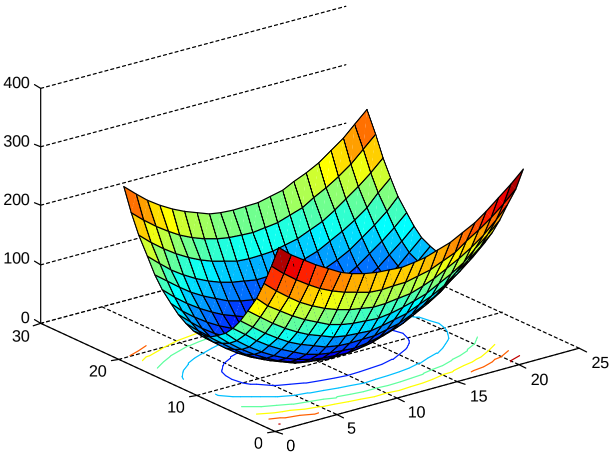
e a nova função F, isto é $F(\lambda) = f(x_0 + \lambda u) = f(-2\lambda, \lambda) = 1 - 3\lambda + 6\lambda^2$. *O minimizante de* $F(\lambda)$ *é obtido de* $F'(\lambda) = 0$, *isto é* $F'(\lambda) = 12\lambda - 3 = 0 \Rightarrow \lambda^* = \frac{1}{4}$, *e como* $F''(\lambda^*) = 12 > 0$, *temos que* $\lambda^* = \frac{1}{4}$ *é um mínimo.*

Solution

Tem-se então que $x^* = x_0 + \lambda^* u = \left(-\frac{1}{2}, \frac{1}{4}\right)$ *é o mínimo de* $f(x)$ *ao longo da linha* $x = x_0 + \lambda u$. *Mais, podemos mostrar que* x^* *é o mínimo global da função* $f(x)$. *Escrevendo*

$$f(x) = f(x_1, x_2) = 1 + x_1 - x_2 + x_1^2 + 2x_2^2 = \left(x_1 + \frac{1}{2}\right)^2 + 2\left(x_2 - \frac{1}{4}\right)^2 + \frac{5}{8}$$

a última função tem curvas de nível (contorno) definidas por elipses centradas no ponto de mínimo $x^* = \left(-\frac{1}{2}, \frac{1}{4}\right)$. *Observa-se que o mínimo ao longo da direcção* $x = x_0 + \lambda u$ *passa pelo centro* $\left(-\frac{1}{2}, \frac{1}{4}\right)$.



Funço $f(x_1, x_2) = 1 + x_1 - x_2 + x_1^2 + 2x_2^2$

Método do gradiente conjugado (Fletcher-Reeves)

- A introdução do método do gradiente conjugado por Fletcher-Reeves conduz a possibilidade de optimizar problemas não lineares de larga escala, pois só são precisos $O(n)$ operações por iteração e pouco espaço para armazenagem.
- São superiores aos métodos que não utilizam informação sobre o gradiente, mas são inferiores aos métodos de tipo Newton.
- Considerando uma equação da forma $f(x) = 0$, é claro que $f(x) = 0$ sse $f(x)f(x) = 0$. Logo, se existirem, os zeros de f, são os pontos de mínimo absoluto de f^2.
- De forma semelhante, no caso de funções com várias variáveis, usando a norma euclidiana, obtemos:

$$F(x) = 0 <=> ||F(x)||^2 = 0 <=> F(x).F(x) = 0$$

e, se existirem, as soluções de $F(x) = 0$ são os pontos de mínimo absoluto de $f(x) = F(x).F(x)$.

Método do gradiente conjugado (Fletcher-Reeves)

- Seja A uma matriz simétrica e definida positiva, e consideremos uma forma quadrática auxiliar:

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

que transforma vectores em números reais.

- Existe apenas um vector que minimiza f e é exactamente o ponto crítico, ou seja, a solução de $f'(x) = 0$, e neste caso,

$$f'(x) = \frac{1}{2}(A^T + A)x - b = Ax - b.$$

Assim, se encontrarmos o ponto de mínimo, ele será solução do sistema linear $Ax = b$.

- Consideramos métodos iterativos de optimização de tipo investigação de linha, ou seja: $x_{n+1} = x_n + a_n h_n$ de forma que haja uma direcção descendente, ou seja, $f(x_{n+1}) < f(x_n)$. O vector h_n define a direcção de investigação descendente.

Método do Gradiente

- No caso do método do gradiente (ou declive máximo - steepest descent), a direcção de descida escolhida é

$$h_n = -\nabla f(x_n) = b - Ax_n,$$

e neste caso designa-se por **resíduo**.

- Resta encontrar o valor a_n que minimiza f, de entre os possíveis valores $x_n + ah_n$.
- Derivando f (nesses pontos) em ordem a a, tem-se,

$$\begin{aligned} \frac{d}{da}\left(f\left(x_n + ah_n\right)\right) &= f'(x_n + ah_n).h_n \\ &= (b - A(x_n + ah_n)).h_n \\ &= h_n.h_n - aAh_n.h_n \end{aligned}$$

- O valor mínimo a_n será obtido como o zero da derivada,

$$h_n.h_n - a_n A\ h_n.h_n = 0 <=> a_n = \frac{h_n.h_n}{A\ h_n.h_n}.$$

Método do Gradiente

- Em conclusão, dado um vector inicial x_0, o método do gradiente resume-se à iteração

$$x_{n+1} = x_n + a_n h_n = x_n + \frac{h_n.h_n}{h_n.A\ h_n} h_n \text{ com } h_n = b - Ax_n$$

- Um critério de paragem consiste em exigir que

$$||h_n||^2 = h_n.h_n < \varepsilon$$

com ε pequeno, notando que isso implica que Ax_n é próximo de b.

Método das Direcções Conjugadas

- Começamos por definir direcções conjugadas: Um conjunto de direcções $\{h_1, h_2, ...\}$ diz-se **conjugado** com respeito a uma matriz A simétrica e definida positiva se

$$h_i^T A h_j = 0 \text{ para qualquer } i \neq j.$$

- Com este produto interno, dois vectores dizem-se A-**ortogonais** se $< h_i, h_j > A = h_i.Ah_j = h_i^T A h_j = 0$, (e as direcções h_i e h_j são conjugadas).
- Seja N a dimensão da matriz A e sejam $h_1, h_2, ..., h_N$ direcções conjugadas, que constituem uma base A-ortogonal em $\mathbb{R}^N$. Se considerarmos h_n como direcções de descida, temos a iteração

$$x_{n+1} = x_n + a_n h_n$$

e queremos agora encontrar o valor a_n que minimiza f, de entre os possíveis valores $x_n + a h_n$.

Método das Direcções Conjugadas

- De forma, semelhante, podemos obter

$$a_n = \frac{r_n.h_n}{h_n.\ A\ h_n}, \text{ com } \ r_n = b - Ax_n$$

e portanto o método de direcções conjugadas é

$$x_{n+1} = x_n + \frac{r_n.h_n}{h_n.\ A\ h_n} h_n$$

Theorem

Um método de direcções conjugadas atinge a solução ao fim de N iterações.

Método dos Gradientes Conjugados

- **Caso particular: método das direcções conjugadas, onde a direcção é dada pelo gradiente** (no caso linear o gradiente é dado pelo resíduo $r_n = b - Ax_n$).
- Através de um processo de ortogonalização de Gram-Schmidt, através dos sucessivos resíduos podemos construir as direcções h_n que serão conjugadas (A-ortogonais).
- Assim, podemos resumir o Método dos Gradientes Conjugados,

1. Dado x_0 definimos $h_0 = r_0 = b - Ax_0$,
2. Definimos $x_{n+1} = x_n + a_n h_n$ com

$$a_n = \frac{r_n.h_n}{h_n.\ A\ h_n}$$

3. Definimos $r_{n+1} = r_n - a_n A h_n$ e

$$h_{n+1} = r_{n+1} + b_n h_n, \quad \text{com} \quad b_n = \frac{r_{n+1}.r_{n+1}}{r_n.r_n}.$$

4. Regressamos ao 2^o passo, até que sejam efectuados N passos.

Método dos Gradientes Conjugados

(a). Método de **Fletcher-Reeves** (1964)

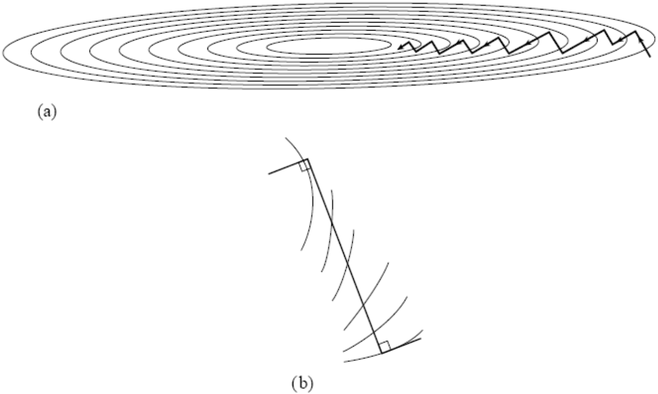
$$b_n = \frac{r_{n+1}.r_{n+1}}{r_n.r_n} = \frac{f'\left(x_{n+1}\right)^T f'\left(x_{n+1}\right)}{f'\left(x_n\right)^T f'\left(x_n\right)}$$

onde x_{prev} é a iterada prévia.
(b). Método de **Polak-Ribière** (1971)

$$b_n = \frac{\left(r_{n+1} - r_n\right).r_{n+1}}{r_n.r_n} = \frac{\left(f'\left(x_{n+1}\right) - f'\left(x_n\right)\right)^T f'\left(x_{n+1}\right)}{f'\left(x_n\right)^T f'\left(x_n\right)}$$

Para uma função quadrática os dois métodos são identicos.

Método dos Gradientes Conjugados



Mtodo do gradiente conjugado

Método dos Gradientes Conjugados

- Em Matlab o programa **mincg.m** está baseado neste método de optimização.

  $$\boxed{\textbf{[res, noiter]=mincg(f, derf, ftau, x, tol)}}$$

  onde **f** é a função a optimizar, **derf** são as derivadas paciais de 1^a ordem da função (gradiente), **ftau** represente o método de investigação de linhas, **x** a condição inicial e **tol** a tolerância permitida.

- De forma alternativa tem-se o m-file **opt_conjg.m**

  $$\boxed{\text{[xo,fo] = opt\_conjg(f,x0,TolX,TolFun,alpha0,MaxIter,KC)}}$$

Método dos Gradientes Conjugados

Example

Seja

$$A = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}.$$

Determine os vectores próprios de A e mostra que são A-conjugados.

Solution

Primeira vez determinam-se os valores próprios, isto é
$|A - \lambda I| = 0 \Longleftrightarrow \lambda_1 = 1$ *e* $\lambda_2 = 3$. *Logo*

Método dos Gradientes Conjugados

$$
\begin{aligned}
(A-\lambda I)\,v &= 0 \Longleftrightarrow \begin{bmatrix} 2-1 & -1 \\ -1 & 2-1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
&\Longleftrightarrow v^1 = \left(v_1^1, v_2^1\right) = \left(v_1^1, v_1^1\right) = v_1^1\,(1,1)\,,\ \forall v_1^1 \in \mathbb{R} \\
&\Longleftrightarrow \begin{bmatrix} 2-3 & -1 \\ -1 & 2-3 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
&\Longleftrightarrow v^2 = \left(v_1^2, v_2^2\right) = \left(v_1^2, -v_1^2\right) = v_1^2\,(1,-1)\,,\ \forall v_1^2 \in \mathbb{R}
\end{aligned}
$$

Agora

$$
\left(v^1\right)^T A v^2 = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = 0
$$

Método dos Gradientes Conjugados

Example

Aplicando o método das direcções conjugadas, minimiza a seguinte função quadrática $\min_{x_1,x_2 \in R} \; f(x_1, x_2) = \frac{1}{2}x^T A x - b^T x + c$, onde

$$A = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}, \; b = \begin{bmatrix} 3 \\ 8 \end{bmatrix}, \; c = 5 \text{ e } x_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Solution

Calculamos os vectores conjugados de A *(exemplo anterior)* $h_0 = (1, 1)$ *e* $h_1 = (1, -1)$. *Agora*

$$\begin{aligned} r_0 &= b - Ax_0 = \begin{bmatrix} 3 \\ 8 \end{bmatrix} - \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 8 \end{bmatrix} \neq 0 \\ a_0 &= \frac{r_0^T h_0}{h_0^T A h_0} = \frac{11}{2}, \; x_1 = x_0 + a_0 h_0 = \begin{bmatrix} \frac{11}{2} & \frac{11}{2} \end{bmatrix}^T \end{aligned}$$

Método dos Gradientes Conjugados

$$
\begin{aligned}
r_1 &= b - Ax_1 = \begin{bmatrix} 3 \\ 8 \end{bmatrix} - \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} \frac{11}{2} \\ \frac{11}{2} \end{bmatrix} = \begin{bmatrix} -\frac{5}{2} \\ \frac{5}{2} \end{bmatrix} \\
a_1 &= \frac{r_1^T h_1}{h_1^T A h_1} = \frac{5}{6} \\
x_2 &= x_1 + a_1 h_1 = \begin{bmatrix} \frac{11}{2} \\ \frac{11}{2} \end{bmatrix} - \begin{bmatrix} \frac{5}{6} \\ \frac{5}{6} \end{bmatrix} = \begin{bmatrix} \frac{14}{3} \\ \frac{14}{3} \end{bmatrix} \\
r_2 &= b - Ax_2 = \begin{bmatrix} 3 \\ 8 \end{bmatrix} - \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} \frac{14}{3} \\ \frac{14}{3} \end{bmatrix} = \begin{bmatrix} -\frac{5}{3} \\ \frac{10}{3} \end{bmatrix}.
\end{aligned}
$$

Método dos Gradientes Conjugados

Example

Aplicando o método do gradiente conjugado (Fletcher-Reeves), minimiza a seguinte função quadrática

$$\min_{x_1, x_2 \in R} \quad f(x_1, x_2) = x_1^2 - x_1 x_2 + 3x_2^2,$$

onde o ponto inicial é $x_0 = (1, 2)$.

Solution

Calculamos o vector gradiente

$$\nabla f(x) = f'(x) = \begin{bmatrix} 2x_1 - x_2 \\ -x_1 + 6x_2 \end{bmatrix}$$

logo $f'(x_0) = \begin{bmatrix} 0 \\ 11 \end{bmatrix}$ *e* $r_0 = h_0 = \begin{bmatrix} 0 \\ 11 \end{bmatrix}$

Método dos Gradientes Conjugados

Agora

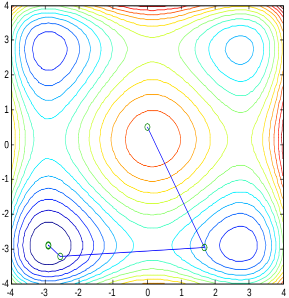
$$
\begin{aligned}
x_1 &= x_0 + a_0 h_0 = \begin{bmatrix} 1 & \frac{1}{6} \end{bmatrix}^T \text{ e } f'(x_1) = \begin{bmatrix} \frac{11}{6} \\ 0 \end{bmatrix} \\
b_1 &= \frac{f'(x_1)^T f'(x_1)}{f'(x_0)^T f'(x_0)} = \frac{1}{36} \\
h_1 &= f'(x_1) + b_1 h_0 = \begin{bmatrix} \frac{11}{6} \\ -\frac{11}{36} \end{bmatrix} \\
x_2 &= x_1 + a_1 h_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.
\end{aligned}
$$

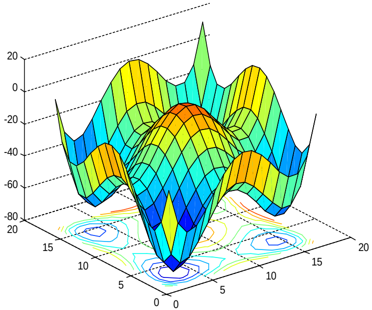
Método dos Gradientes Conjugados

Example

Determinar um mínimo local da função

$$f(x,y) = 0.5(x^4 - 16x^2 + 5x) + 0.5(y^4 - 16y^2 + 5y)$$





```
>> x1=mincg('f801','f801pd','ftau2cg', x0, 0.000005)
> number of iter= 6, > Solution   2.7468   -2.9035
>> x=-4:0.1:4;y=-4:0.1:4; [X,Y]=meshgrid(x,y);
>> z=0.5.*(X.^4-16.*X.^2+5.*X)+0.5.*(Y.^4-16.*Y.^2+5.*Y);
>> x1=[0 1.691 -2.560 -2.909 -2.903 -2.903]; y1=[0.5 -2.948 -3.210
-2.893 -2.903 -2.903];
>> figure; axis([-4 4 -4 4 -80 20]); contour(-4:0.1:4, -4:0.1:4, z,15)
>> hold on; plot(x1,y1,x1,y1,'o')
```

Método de Newton

- O método de Newton requer a computação do **vector gradiente**

$$g = \nabla f(x) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(x) \\ \vdots \\ \frac{\partial f}{\partial x_n}(x) \end{bmatrix}, \text{ onde } f : D \subset \mathbb{R}^n \to \mathbb{R} \text{ e } x = (x_1, ..., x_n)$$

e da **matriz Hessiana**

$$H(f(x)) = \nabla^2 f(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2}(x) & \frac{\partial^2 f}{\partial x_1 \partial x_2}(x) & \dots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(x) \\ \frac{\partial^2 f}{\partial x_2 \partial x_1}(x) & \frac{\partial^2 f}{\partial x_2^2}(x) & \dots & \frac{\partial^2 f}{\partial x_2 \partial x_n}(x) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1}(x) & \frac{\partial^2 f}{\partial x_n \partial x_2}(x) & \dots & \frac{\partial^2 f}{\partial x_n^2}(x) \end{bmatrix}.$$

Método de Newton

Teorema: *Suponhamos que* $f : D \subset \mathbb{R}^n \rightarrow \mathbb{R}$ *é duas vezes diferenciável e* $x^* \in \mathbb{R}^n$ *satisfaz*

1. $\nabla f\left(x^*\right) = 0$
2. $\nabla^2 f\left(x^*\right)$ *é definida positiva (em particular, não-singular)*
3. $\nabla^2 f$ *é Lipschitz contínua numa vizinhança do ponto* x^*.

Então x^* *é um mínimo local de* f *e para qualquer* x_0 *suficientemente próximo de* x^**, o método de Newton define uma sequência que converge (local) quadraticamente para* x^*.

Nota: O método ainda converge se a matriz Hessiana é semidefinida positiva e singular.

Método de Newton

- O método de Newton determina um modelo quadrático para a função objectivo em torno da iterada corrente x_k definida por

$$q_k(h_k) = f(x_k) + \nabla^T f(x_k)\, h_k + \frac{1}{2} h_k^T \nabla^2 f(x_k)\, h_k.$$

Na versão básica do método de Newton, a próxima iterada é obtida a partir do minimizante de q_k. Quando a matriz Hessiana é definida positiva, o modelo quadrático tem um único minimizante que pode ser obtido resolvendo o seguinte sistema simétrico $(n \times n)$

$$\nabla^2 f(x_k)\, h_k = -\nabla f(x_k) \;\Leftrightarrow\; h_k = -\left(\nabla^2 f(x_k)\right)^{-1} \nabla f(x_k)$$

Método de Newton

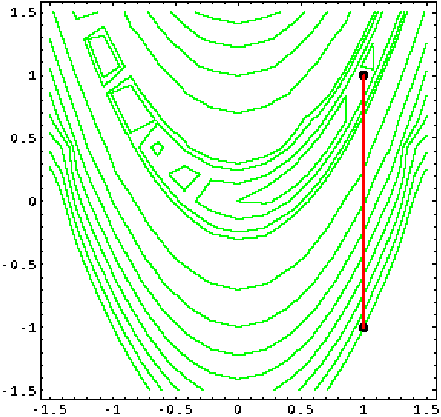
- Agora, a nova iterada vai ser

$$x_{k+1} = x_k + h_k = x_k - \left(\nabla^2 f\left(x_k\right)\right)^{-1} \nabla f\left(x_k\right)$$

Como já referido a convergência (local) é garantida se o ponto de partida é suficientemente próximo de um minimizante local x^*. O critério de paragem é dado por

$$\frac{\alpha^2}{2} \leq \varepsilon \text{ onde } \alpha\left(x\right) = \left(\nabla^T f\left(x\right) \left(\nabla^2 f\left(x\right)\right)^{-1} \nabla f\left(x\right)\right)^{1/2} = \|x - x^*\|$$

Método de Newton



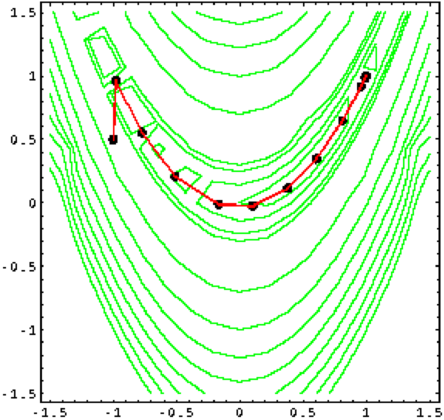


Figure: Método de Newton. Caso (a), convergência imediata para p mínimo (1 iterada) e caso (b) convergência ao fim de 10 iteradas.

Método de Newton

- O algoritmo de Newton se funciona bem, funciona muito bem. Com tudo isto não é o método ideal, pois requer a computação da matriz hessiana em cada iterada, também como a resolução do sistema linear que envolve a hessiana. Falha facilmente se a matriz hessiana não é definida positiva ou regular.
- Nestes casos convém substituir a matriz Hessiana H por uma nova matriz $\hat{H} = H + E$. O método de Levenberg-Marquardt pode ser utilizado com sucesso, fazendo a substituição: $\hat{H} = H + \gamma I$, com I matriz identidade. O papel de γ é manter a matriz H definida positiva.
- **Matlab**: m-file **newtons.m**

$$\boxed{\text{[x,fx,xx] = newtons(f,x0,TolX,MaxIter,varargin)}}$$

Métodos Quase-Newton ou métodos métricos variáveis

- Podem ser utilizados quando a matriz Hessiana é difícil de calcular ou demora muito tempo para ser avaliada.
- Este método constrói gradualmente uma matriz Hessiana aproximada utilizando informações a partir do gradiente avaliado para quase todas as iteradas anteriores visitadas pelo algoritmo.
- Consta de dois passos fundamentais
  - Determinação de uma direcção de investigação
  - Escolha do método de investigação da linha

Métodos Quase-Newton ou métodos métricos variáveis

- Dada a iterada corrente x_k e a matriz Hessiana aproximada B_k em x_k, a solução do sistema linear

$$B_k h_k = -\nabla f(x_k) \Leftrightarrow h_k = -B_k^{-1} \nabla f(x_k)$$

gera uma direcção h_k.

- A próxima iterada encontra-se fazendo uma investigação linear sobre h_k e considerando

$$x_{k+1} = x_k + \alpha_k h_k,$$

onde o comprimentos de passo α_k satisfaz as condições de Wolfe, isto é,

$$\begin{aligned} f(x_k + \alpha_k h_k) &\leq f(x_k) + \sigma_1 \alpha_k \nabla^T f(x_k) h_k \\ \nabla^T f(x_k + \alpha_k h_k) h_k &\leq \sigma_2 \nabla^T f(x_k) h_k \end{aligned}$$

onde $0 < \sigma_1 < \sigma_2 < 1$.

Métodos Quase-Newton ou métodos métricos variáveis

- Para obter a aproximação B_{k+1} a partir de B_k é preciso considerar o Teorema Fundamental de Cálculo Integral. Se definimos

$$\begin{aligned} s_k &= x_{k+1} - x_k \\ y_k &= \nabla f(x_{k+1}) - \nabla f(x_k) \end{aligned}$$

então

$$\underbrace{\left\{ \int_0^1 \nabla^2 f(x_k + t s_k)\, dt \right\}}_{\text{média da Hessiana sobre o segmento } [x_k, x_k + s_k]} \qquad s_k = y_k.$$

- A partir dessa observação podemos obter a aproximação B_{k+1} imitando o comportamento da matriz $\nabla^2 f$ e considerando a condição de quase-Newton

$$B_{k+1} s_k = y_k.$$

Métodos Quase-Newton ou métodos métricos variáveis

- Ao fim de cada passo a aproximação da matriz Hessiana vem actualizada. A mais comum família de actualizações é a de **Broyden de rank 2**, isto é
$$B_{k+1} = B_k - \frac{B_k s_k \left(B_k s_k\right)^T}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k} + \phi_k [s_k^T B_k s_k] v_k v_k^T,$$
onde
$$\phi_k \in [0,1] \text{ e } v_k = \left[\frac{y_k}{y_k^T s_k} - \frac{B_k s_k}{s_k^T B_k s_k}\right].$$
Para $\phi_k = 0$ tem-se a actualização de **Broyden -Fletcher -Goldfarb-Shanno** (BFGS) e para $\phi_k = 1$ tem-se a actualização de **Davidon-Fletcher-Powell** (DFP).
- Aplicando qualquer uma dessas actualizações a matriz Hessiana aproximada permanece definida positiva e portanto a direcção de procura é sempre uma direcção descendente.
- São algoritmos que convergem rapidamente e evitam a computação

Métodos Quase-Newton ou métodos métricos variáveis

- **Matlab**

fminunc é o m-file que determine extremos livres de uma função real de várias variáveis reais.

$>>$ $\boxed{\textbf{x=fminunc('fname',x0)}}$

Inicia em x_0 e tenta encontrar o mínimo local x da função 'fname' (definida como m-file também)

- **Exemplo**

```
>>x = fminunc('ffdi',[2 3])   % sem gradiente
> f = -0.2750
...
> f = -2.0000
Optimization terminated: relative infinity-norm of gradient less than options.TolFun.
x = 3.1416       4.7124
```

Métodos Quase-Newton ou métodos métricos variáveis

- Para minimizar esta função já com o gradiente dado modificamos a função como a seguir (indicando no comando **fminunc** que o gradiente está disponível utilizando OPTIONS)

```
>>x = fminunc('fname', x0, options);
>> options=optimset('GradObj','on'); x0=[2 3]
>> [x,fval]=fminunc('dffdi', x0, options)
Optimization terminated: first-order optimality less than
OPTIONS.TolFun,
and no negative/zero curvature detected in trust region model.
x = 3.1416     10.9956
fval = -2
```