

## Otimização não linear

Isabel Espírito Santo

Departamento de Produção e Sistemas

Escola de Engenharia

Universidade do Minho

iapinho@dps.uminho.pt

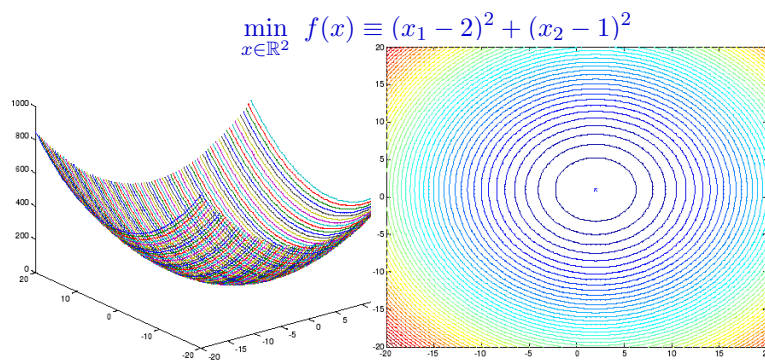
<http://www.norg.uminho.pt/iapinho/>

## Formulação de um problema sem restrições

$$\min_{x \in \mathbb{R}^n} f(x) \quad (1)$$

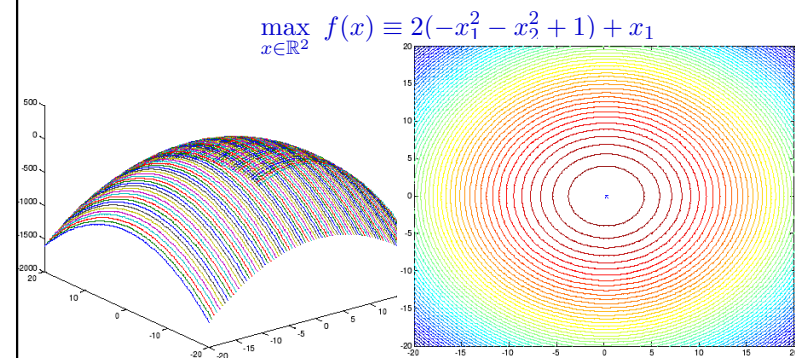
- Se  $n = 1 \Rightarrow$   $\left[ \begin{array}{l} \text{problema unidimensional} \\ x \text{ é escalar} \end{array} \right.$
- Se  $n > 1 \Rightarrow$   $\left[ \begin{array}{l} \text{problema multidimensional} \\ x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \text{ é vetor de dimensão } n \end{array} \right.$

## Problema multidimensional ( $n > 1$ ) sem restrições



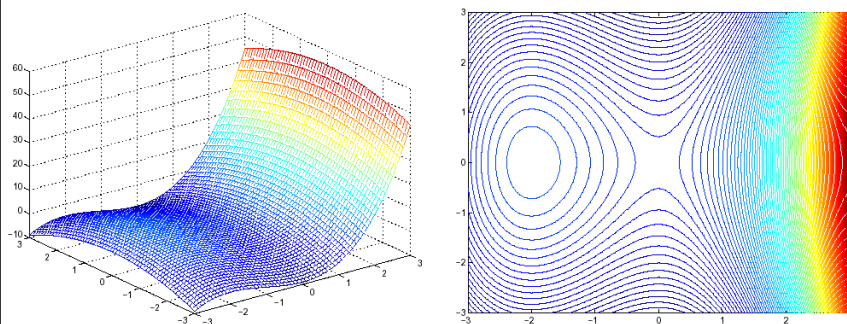
$f(x)$  - função objetivo

## Problema multidimensional ( $n > 1$ ) sem restrições



## Problema multidimensional ( $n > 1$ ) sem restrições

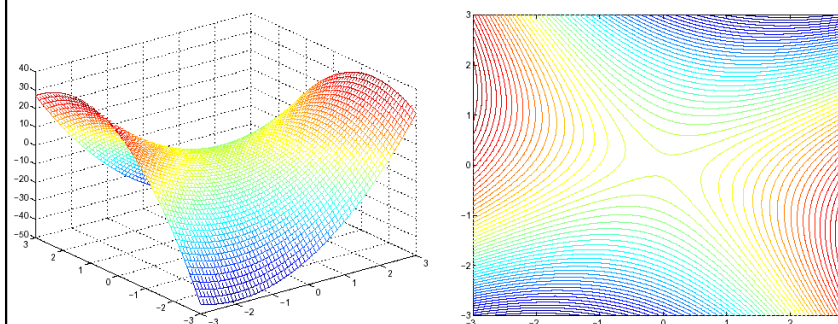
$$\min_{x \in \mathbb{R}^2} f(x) \equiv 3x_1^2 - x_2^2 + x_1^3$$



Ponto sela em  $(0,0)$

## Problema multidimensional ( $n > 1$ ) sem restrições

$$\min_{x \in \mathbb{R}^2} f(x) \equiv 3x_1^2 - 4x_1x_2 - 4x_2^2$$



Ponto sela em  $(0,0)$

## Notação

Vetor gradiente da função  $f(x)$  -  $x \in \mathbb{R}^n$  -

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix} \text{ vector de } \mathbb{R}^n$$

Matriz Hessiana da função  $f(x)$

$$\nabla^2 f(x) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_2 \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_1} \\ \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_1 \partial x_n} & \frac{\partial^2 f}{\partial x_2 \partial x_n} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix} \text{ matriz simétrica de } n \times n$$

## Condições de otimalidade

Assume-se  $f(x)$  continuamente diferenciável até à 2ª ordem.

Condição necessária (e suficiente) de 1ª ordem:

Se  $x^*$  é uma solução do problema (1) então  $\nabla f(x^*) = 0$ ;

(Se  $\nabla f(x^*) = 0$  então  $x^*$  é candidato a minimizante);

**Nota:** A condição  $\nabla f(x) = 0$  define os pontos estacionários de  $f$ :

- { minimizante - exemplo 5
- { maximizante - exemplo 6
- { ponto sela - exemplos 7 e 8

## Condições de otimalidade

Condição necessária de 2ª ordem:

Se  $x^*$  é uma solução do problema (1) que satisfaz a condição de 1ª ordem, então  $\nabla^2 f(x^*)$  é semi-definida positiva.

**Condição suficiente de 2ª ordem:**

Se  $x^*$  é um ponto que verifica a condição de 1ª ordem e se  $\nabla^2 f(x^*)$  é definida positiva, então  $x^*$  é um **minimizante local forte** de (1).

## Condições de otimalidade

Assumindo  $\nabla f(x^*) = 0$ :

- as condições necessária e suficiente de 2ª ordem para um **maximizante** são respetivamente
  - $\nabla^2 f(x^*)$  é semi-definida negativa
  - $\nabla^2 f(x^*)$  é definida negativa
- se  $\nabla^2 f(x^*)$  é indefinida, então  $x^*$  é ponto sela (ou de descanso).

## Conclusão

Seja  $x^*$  um ponto para o qual  $\nabla f(x^*) = 0$  e  $\nabla^2 f(x^*) \neq$  matriz nula:

- Se  $\nabla^2 f(x^*)$  é definida positiva então  $x^*$  é minimizante
- Se  $\nabla^2 f(x^*)$  é definida negativa então  $x^*$  é maximizante
- Se  $\nabla^2 f(x^*)$  é semi-definida positiva então  $x^*$  é minimizante ou ponto sela
- Se  $\nabla^2 f(x^*)$  é semi-definida negativa então  $x^*$  é maximizante ou ponto sela
- Se  $\nabla^2 f(x^*)$  é indefinida então  $x^*$  é ponto sela.

## Métodos numéricos de resolução

problema sem restrições

$$\min_{x \in \mathbb{R}^n} f(x) \quad (n > 1)$$

- Métodos de procura direta;
- Métodos do gradiente.

**Métodos de procura direta:**

- só usam informação da função objetivo  $f$ ;
- são apropriados para **problemas não diferenciáveis** (embora possam ser usados em problemas diferenciáveis);
- método de Nelder-Mead (destina-se a problemas de otimização multidimensionais).

## Métodos do gradiente

### Métodos do gradiente:

- usam informação da função e das derivadas (gradiente ou/e Hessiana);
- só podem ser usados na resolução de **problemas diferenciáveis**;
- convergem mais rapidamente do que os métodos de procura directa;
- geram uma sucessão de aproximações  $\{x^{(k)}\}$  à solução:

$$x^{(k+1)} = x^{(k)} + \alpha^k d^{(k)}$$

em que  $d^{(k)}$  (vector) é a **direção de procura** (ou passo) e  $\alpha^k$  (escalar) é o **comprimento do passo**. A equação iterativa para o cálculo da direção de procura é diferente para cada método.

## Método de Newton

Derivando em ordem a  $d$  e igualando a zero (define a condição de 1ª ordem para o mínimo da quadrática,  $\nabla f(d) = 0$ ), obtém-se

$$\begin{aligned}\nabla f(x^{(k)}) + \nabla^2 f(x^{(k)}) d &= 0 \\ \Updownarrow \\ \nabla^2 f(x^{(k)}) d &= -\nabla f(x^{(k)})\end{aligned}\quad (2)$$

A solução do **sistema linear** (2),  $d$ , é a direção de procura:

- se a dimensão do problema ( $n$ ) for pequena ou média, usa-se o método directo e estável - EGPP;
- se  $n$  for grande, usa-se um método iterativo - gradientes conjugados.

## Método de Newton

A nova aproximação  $x^{(k)} + d$  não é necessariamente o minimizante de  $f(x)$  e o processo deve ser repetido.

As equações iterativas do **Método de Newton** (na forma básica) são

$$\begin{cases} \nabla^2 f(x^{(k)}) d_N^{(k)} = -\nabla f(x^{(k)}), & \text{(sistema Newton)} \\ x^{(k+1)} = x^{(k)} + d_N^{(k)} & \text{para } k = 1, 2, \dots \end{cases}$$

em que  $d_N^{(k)}$  é a **direção Newton**.

## Propriedades do método de Newton

- o método de Newton tem convergência
  - **local** (a convergência para a solução só é garantida se a aproximação inicial  $x^{(1)}$  estiver na vizinhança da solução);
  - **quadrática**

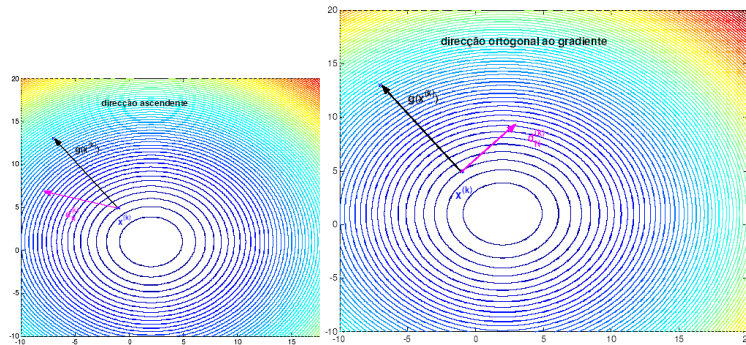
$$\|x^{(k+1)} - x^*\| \leq \gamma \|x^{(k)} - x^*\|^2, \quad \gamma > 0;$$

- o método de Newton possui a propriedade da **terminação quadrática**, i.e., se  $f(x)$  ( $x \in \mathbb{R}^n$ ) for uma função quadrática e convexa o método de Newton necessita no máximo de  $n$  iterações para encontrar a solução.

### Limitações do método de Newton

- a direção  $d_N^{(k)}$  (solução do **sistema Newton**) pode **não** ser **descendente** para  $f$  em  $x^{(k)}$ , ou seja,

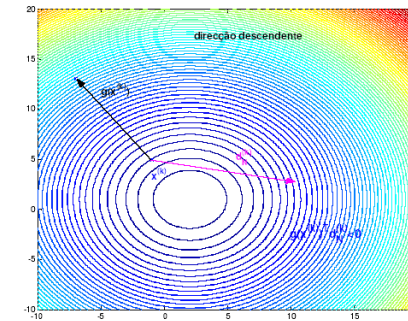
$$\nabla f(x^{(k)})^T d_N^{(k)} > 0 \quad \text{ou} \quad \nabla f(x^{(k)})^T d_N^{(k)} = 0$$



### Limitações do método de Newton

- A direção  $d_N^{(k)}$ , ainda que seja **descendente** ( $\nabla f(x^{(k)})^T d_N^{(k)} < 0$ ), pode ser muito grande e se  $x^{(k+1)} = x^{(k)} + d_N^{(k)}$  não se verifica

$$f(x^{(k)} + d_N^{(k)}) < f(x^{(k)})$$



### Limitações do método de Newton

- Além disso, a matriz  $\nabla^2 f(x^{(k)})$  (matriz dos coeficientes do **sistema Newton**) pode ser singular, o que significa que o sistema Newton não tem solução ou tem uma infinidade de soluções



$$\nexists d_N^{(k)} \text{ (única)}$$

### Desvantagens do método de Newton

- Cálculo das segundas derivadas:
  - se a expressão de  $f$  é complicada, estas tornam-se difíceis de calcular;
  - exigem um grande esforço de cálculo quando  $n$  é grande.
- A convergência é local.

#### Para ultrapassar a convergência local

deve implementar-se uma técnica de globalização para garantir que o método converge para a solução, a partir de qualquer aproximação inicial.

## Técnicas de globalização

Os métodos do gradiente (quando convergem) convergem para um ponto estacionário ( $\nabla f(x) = 0$ ).

**Porquê** implementar uma técnica de globalização?

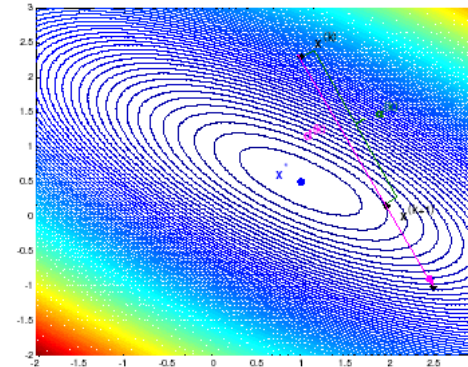
- para garantir que o método converge, qualquer que seja a aproximação inicial  $x^{(1)}$  (i.e.,  $x^{(1)}$  pode estar fora da região de convergência do método);
- para garantir que o método converge para um ponto estacionário que é **minimizante**.

1. Procura unidimensional (line search)  $\left\{ \begin{array}{l} \text{exata} \\ \text{aproximada} \end{array} \right.$
2. Região de confiança (trust region)
3. Filtro

## Procura unidimensional exata

Dados  $x^{(k)}$  e  $d^{(k)}$ , calcular  $\alpha^{(k)}$ :

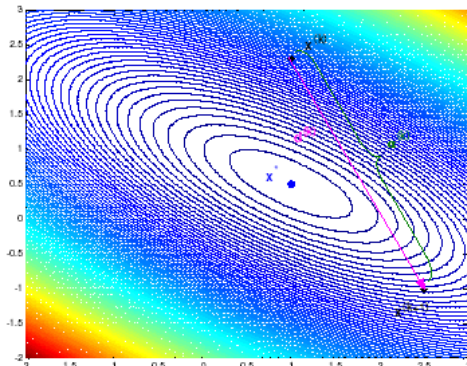
$$\left\{ \begin{array}{l} \text{comprimento do passo ótimo: } \alpha^{(k)} = \arg \min_{\alpha} f(x^{(k)} + \alpha d^{(k)}) \\ \text{nova aproximação à solução: } x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{(k)} \end{array} \right.$$



## Procura unidimensional aproximada

Dados  $x^{(k)}$  e  $d^{(k)}$ , calcular  $\alpha^{(k)}$  (comprimento do passo) que origina uma redução significativa no valor de  $f$  na nova aproximação (isto é, satisfaz a **condição de Armijo**):

$$f(x^{(k)} + \alpha^{(k)} d^{(k)}) \leq f(x^{(k)}) + \mu \alpha^{(k)} \nabla f(x^{(k)})^T d^{(k)}$$



## Crítério de Armijo

O objetivo é exigir uma redução significativa (usando a condição de Armijo) no valor da função objetivo, i.e.,

$$f(x^{(k)} + \alpha^{(k)} d^{(k)}) \leq f(x^{(k)}) + \mu \alpha^{(k)} \nabla f(x^{(k)})^T d^{(k)}$$

com  $0 < \mu < \frac{1}{2}$ .

**Nota:** se a direção  $d^{(k)}$  usada for **descendente** para  $f$ , ou seja,  $\nabla f(x^{(k)})^T d^{(k)} < 0$ , existe um valor de  $\alpha^{(k)} \in (0, 1]$  que verifica esta condição.



### Algoritmo do critério de Armijo para calcular $\alpha^{(k)}$

Dados  $x^{(k)}, d^{(k)}, \nabla f(x^{(k)}), f(x^{(k)})$  e  $\mu$

1.  $\alpha \leftarrow 1$
2.  $\bar{x} \leftarrow x^{(k)} + \alpha d^{(k)}$
3. se  $(f(\bar{x}) \leq f(x^{(k)}) + \mu \alpha \nabla f(x^{(k)})^T d^{(k)})$  então  
fazer  $\alpha^{(k)} = \alpha$   
senão  
 $\alpha \leftarrow \alpha/2$  e voltar a 2.

### Algoritmo genérico de um método do gradiente

ler aproximação inicial  $x^{(1)} \in \mathbb{R}^n$

$k \leftarrow 0$

repetir

$\left\{ \begin{array}{l} k \leftarrow k + 1 \\ \text{calcular } d^{(k)} \text{ (direção de procura ou passo)} \\ \quad \text{(por exemplo, a direção do método de segurança de Newton} \\ \quad \text{ou a direção do método quasi-Newton)} \\ \text{calcular } \alpha^{(k)} \text{ (comprimento do passo)} \\ \quad \text{(usando, por exemplo, o critério de Armijo)} \\ \text{definir } x^{(k+1)} \leftarrow x^{(k)} + \alpha^{(k)} d^{(k)} \end{array} \right.$

até (CP = verdadeiro)

Solução:  $\begin{cases} x^* \approx x^{(k+1)} \\ f^* \approx f(x^{(k+1)}) \end{cases}$

### Critério de paragem (CP)

Parar o processo iterativo se

$$\underbrace{\|\nabla f(x^{(k+1)})\|_2}_{\text{medida de estacionaridade}} \leq \varepsilon$$

$\varepsilon$  constante positiva próxima de zero.

### Método de segurança de Newton

É possível ultrapassar as limitações do método de Newton, implementando, em cada iteração, as seguintes **sugestões** - garantem que a direção calculada é **descendente** para a função  $f \Rightarrow$  algoritmo de segurança de Newton.

Em qualquer iteração  $k$ :

- Quando  $\nabla^2 f(x^{(k)})$  é singular  
 $\Rightarrow$  usar  $d_{SN}^{(k)} = -\nabla f(x^{(k)})$   
(em que  $-\nabla f(x^{(k)})$  é a direção de descida máxima e é descendente para  $f$ )

## Método de segurança de Newton

- Quando  $d_N^{(k)}$  é ortogonal ao gradiente  
 $\Leftrightarrow \left| \nabla f(x^{(k)})^T d_N^{(k)} \right| \leq \eta$   
 (com  $\eta > 0 (\approx 0)$ )  
 $\Rightarrow$  usar  $d_{SN}^{(k)} = -\nabla f(x^{(k)})$
- Quando  $d_N^{(k)}$  é ascendente  
 $\Leftrightarrow \nabla f(x^{(k)})^T d_N^{(k)} > \eta$   
 $\Rightarrow$  usar  $d_{SN}^{(k)} = -d_N^{(k)}$
- Quando  $d_N^{(k)}$  é descendente  
 $\Rightarrow$  usar  $d_{SN}^{(k)} = d_N^{(k)}$

NOTA:  $d_{SN}^{(k)}$  é descendente, para todo o  $k$ .

## Algoritmo para o cálculo da direção de segurança de Newton

**Dados**  $x^{(k)}$  e  $\eta$ ,

**Resolver** o sistema linear Newton  $\nabla^2 f(x^{(k)}) d_N^{(k)} = -\nabla f(x^{(k)})$  por EGPP

se (o sistema linear tem solução única -  $\exists d_N^{(k)}$ ) então

se  $\left| \nabla f(x^{(k)})^T d_N^{(k)} \right| \leq \eta$

então  $d_{SN}^{(k)} \leftarrow -\nabla f(x^{(k)})$

senão

se  $\nabla f(x^{(k)})^T d_N^{(k)} > \eta$

então  $d_{SN}^{(k)} \leftarrow -d_N^{(k)}$

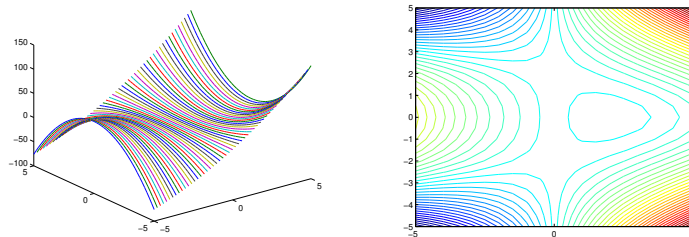
senão  $d_{SN}^{(k)} \leftarrow d_N^{(k)}$

senão

$d_{SN}^{(k)} \leftarrow -\nabla f(x^{(k)})$

## Exemplo

Dada a função  $f(x_1, x_2) = x_1 x_2^2 + (2 - x_1)^2$  calcule o seu mínimo usando o algoritmo de segurança de Newton. A partir de  $(1, 1)$ , considere  $\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = 0.1$ ,  $\eta = 0.0001$  e  $\mu = 0.001$ .



## A equação para o cálculo da direção Newton - relembrar

método de Newton

$$\nabla^2 f(x^{(k)}) d_N^{(k)} = -\nabla f(x^{(k)})$$

Para evitar o cálculo das 2<sup>as</sup> derivadas para a Hessiana,  $\nabla^2 f(x)$ , pode usar-se uma aproximação:

$$B^{(k)} \approx \nabla^2 f(x^{(k)})$$

ou, como o sistema Newton pode ser escrito da seguinte forma - não aconselhável na prática,

$$d_N^{(k)} = - \left( \nabla^2 f(x^{(k)}) \right)^{-1} \nabla f(x^{(k)})$$



## A equação para o cálculo da direção quasi-Newton

é aconselhável usar-se, em cada iteração  $k$ , uma aproximação à **inversa da Hessiana**

$$H^{(k)} \approx \left( \nabla^2 f(x^{(k)}) \right)^{-1}$$

**método quasi-Newton**

$$d_{QN}^{(k)} = -H^{(k)} \nabla f(x^{(k)})$$

$d_{QN}^{(k)}$  - é a direção quasi-Newton

## Método quasi-Newton

Evita-se desta forma:

- o cálculo das segundas derivadas (para formar a matriz Hessiana);
- a resolução de um sistema linear em cada iteração, substituindo-o pelo produto de uma matriz por um vetor.

As equações que descrevem o método quasi-Newton são:

$$\begin{cases} d_{QN}^{(k)} = -H^{(k)} \nabla f(x^{(k)}), \\ x^{(k+1)} = x^{(k)} + \alpha^{(k)} d_{QN}^{(k)} \quad \text{para } k = 1, 2, \dots \end{cases}$$

## Características da matriz H:

- deve aproximar, o melhor possível, a inversa de  $\nabla^2 f(x^{(k)})$ , ou seja, deve verificar a **condição secante**:

$$H^{(k)} y^{(k-1)} = s^{(k-1)}$$

com

$$y^{(k-1)} = \nabla f(x^{(k)}) - \nabla f(x^{(k-1)})$$

(variação verificada no gradiente da iteração  $k-1$  para a iteração  $k$ )

$$s^{(k-1)} = x^{(k)} - x^{(k-1)} = \alpha^{(k-1)} d_{QN}^{(k-1)}$$

(variação verificada em  $x$ )

## Características da matriz H:

- deve preferencialmente ser

$$\begin{cases} \textbf{simétrica} & \text{(pois } \nabla^2 f(x^{(k)})^{-1} \text{ também é simétrica)} \\ \textbf{definida positiva} & \text{(pois a direção } d_{QN}^{(k)} = -H^{(k)} \nabla f(x^{(k)}) \end{cases}$$

é **descendente** para  $f$  em  $x^{(k)}$ .

As matrizes  $H^{(k)}$  são geradas através de **fórmulas de atualização** do tipo

$$H^{(k)} = H^{(k-1)} + E^{(k-1)}, \quad k > 1$$

e devem manter-se **simétricas** e **definidas positivas**.

## Matriz H da 1ª iteração

- Para que simetria + definida positiva se conservem ao longo do processo iterativo, a matriz inicial  $H^{(1)}$ , para  $k = 1$ , deve ser também simétrica e definida positiva. Por exemplo

$$H^{(1)} = I.$$

A  $I$  não é necessariamente uma boa aproximação a  $\nabla^2 f(x^{(1)})^{-1}$ , mas as fórmulas de atualização de  $H$  melhoram as aproximações.

- Existem várias fórmulas de atualização para as matrizes  $H$ :
  - nem todas conservam a simetria + definida positiva;
  - as 2 fórmulas seguintes conservam simetria + definida positiva.

## Fórmulas de atualização

Davidon, Fletcher e Powell - **DFP**

$$H^{(k)} = H^{(k-1)} - \frac{H^{(k-1)} y^{(k-1)} y^{(k-1)T} H^{(k-1)}}{y^{(k-1)T} H^{(k-1)} y^{(k-1)}} + \frac{s^{(k-1)} s^{(k-1)T}}{s^{(k-1)T} y^{(k-1)}}$$

Broyden, Fletcher, Goldfarb e Shanno - **BFGS**

$$H^{(k)} = \left( I - \frac{s^{(k-1)} y^{(k-1)T}}{s^{(k-1)T} y^{(k-1)}} \right) H^{(k-1)} \left( I - \frac{y^{(k-1)} s^{(k-1)T}}{s^{(k-1)T} y^{(k-1)}} \right) + \frac{s^{(k-1)} s^{(k-1)T}}{s^{(k-1)T} y^{(k-1)}}$$

**Nota:**  $y^{(k-1)T} s^{(k-1)} > 0$  é a condição necessária e suficiente para que as matrizes se conservem definidas positivas.

## Propriedades do método quasi-Newton

- o método quasi-Newton tem convergência
  - local** (a convergência para a solução só é garantida se a aproximação inicial  $x^{(1)}$  estiver na vizinhança da solução);
  - superlinear** verifica-se
 
$$\|x^{(k+1)} - x^*\| \leq \gamma_k \|x^{(k)} - x^*\|$$
 com a sucessão  $\{\gamma_k\} \rightarrow 0$  quando  $k \rightarrow \infty$ ;
- o método quasi-Newton satisfaz a propriedade da **terminação quadrática** – isto é, o mínimo de uma função quadrática  $q(x)$ ,  $x \in \mathbb{R}^n$ , obtém-se em  $n$ , ou menos, do que  $n$  iterações.

## Limitação do método quasi-Newton

★ Os erros de arredondamento que se cometem nos cálculos podem fazer com que  $H^{(k)}$  deixe de ser definida positiva e a direção  $d_{QN}^{(k)}$  deixa de ser descendente para  $f$  em  $x^{(k)}$

↓

**Solução:** fazer  $H^{(k)} = I$  (neste caso  $H^{(k)}$  é simétrica e definida positiva)

↓

$$d_{QN}^{(k)} = -\nabla f(x^{(k)})$$

## Algoritmo para o cálculo da direção quasi-Newton

**Dado**  $x^{(k)}$

**Calcular**  $d_{QN}^{(k)} \leftarrow -H^{(k)} \nabla f(x^{(k)})$ , sendo  $H^{(k)}$  dada por:

se  $k = 1$ , então  $H^{(k)} \leftarrow I$

senão  $\left\{ \begin{array}{l} s^{(k-1)} \leftarrow x^{(k)} - x^{(k-1)} \\ y^{(k-1)} \leftarrow \nabla f(x^{(k)}) - \nabla f(x^{(k-1)}) \\ \text{actualizar } H^{(k)} \text{ pela fórmula DFP ou BFGS} \end{array} \right.$

se a direção não é descendente

$$\left( \nabla f(x^{(k)})^T d_{QN}^{(k)} \geq 0 \right)$$

então fazer  $d_{QN}^{(k)} \leftarrow -\nabla f(x^{(k)})$