# Programação I
# Aula 7 — Resolução numérica de equações

Pedro Vasconcelos
DCC/FCUP



## Nesta aula

# Resolução numérica de equações

Problema: encontrar uma solução duma equação como

$$x^3 - 5x + 2 = 0$$

Equivalente: encontrar um zero da função

$$f(x) = x^3 - 5x + 2$$

isto é, um valor $x^*$ tal que $f(x^*) = 0$.



# Resolução numérica de equações (cont.)

- Para alguns casos conseguimos exprimir a solução de forma análitica (exemplo: a formula resolvente do 2º grau)
- Alternativa: obter uma sequência de aproximações numéricas

$$x_1, x_2, \ldots, x_n \ \ldots$$

  que convergem para a raiz
- Quanto mais iterações fizermos, melhor será a aproximação
- Vamos ver o método de bisseções e o método de Newton

# Método de bisseções

Partimos de:

uma função $f$;

um intervalo $[a, b]$;

a tolerância $\epsilon > 0$ (usada como *critério de paragem*);

tais que:

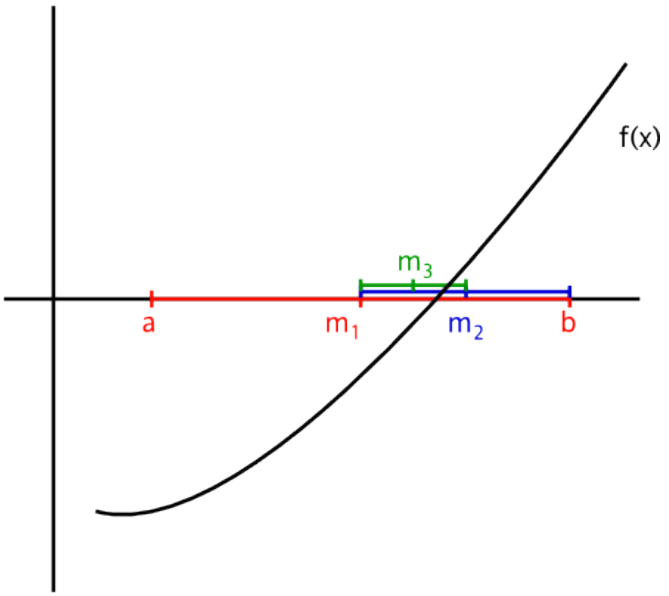
1. $f$ é contínua em $[a, b]$;
2. $f(a) \times f(b) < 0$ (isto é, $f$ muda de sinal em $[a, b]$);
3. $f$ tem um zero único em $[a, b]$.



# Algoritmo

Repetimos enquanto $b - a > \epsilon$:

1. calculamos o ponto médio
   $m \leftarrow (a + b)/2$
2. se $f(a) \times f(m) < 0$:
   a raiz está em $[a, m]$
   $b \leftarrow m$
3. se $f(a) \times f(m) \geq 0$:
   a raiz está em $[m, b]$
   $a \leftarrow m$



# Implementação em Python

```
def bissect(f, a, b, eps):
    "Método de bisseções para f(x)=0 em [a,b]."
    while b-a>eps:          # critério de paragem
        m = (a+b)/2         # calcular o ponto médio
        # atualizar um dos extremos
        if f(a)*f(m) < 0:
            b = m
        else:
            a = m
    # fim do ciclo
    return m                # o ponto médio final
```



# Utilização

Encontrar a raiz:

- do polinómio $P(x) = x^3 - 5x + 2$
- no intervalo $[0, 1]$
- com erro $< 10^{-4}$.

```
>>> def P(x): return x**3-5*x+2
>>> bissect(P, 0, 1, 1e-4)
0.41424560546875
```

# Alternativa

Em vez de definir uma função $P(x)$ podemos usar uma expressão lambda:

```
>>> bissect(lambda x : x**3-5*x+2, 0, 1, 1e-4)
0.41424560546875
```

A expressão

```
lambda x : x**3-5*x+2
```

representa a função

$$x \longmapsto x^3 - 5x + 2$$



# Melhorar a implementação

- Em cada iteração calculamos $f(a)$ e $f(m)$
- Podemos fazer melhor:
  1. guardamos $f(a)$, $f(m)$ em duas variáveis *fa* e *fm*
  2. actualizamos de forma a manter os invariantes seguintes:

$$\begin{cases} fa = f(a) \\ fm = f(m) \end{cases}$$

- Com isto reduzimos o custo computacional do algoritmo

# Implementação de bisseções (2)

```
def bissect(f, a, b, eps):
    "Método de bissecções para f(x)=0 em [a,b]."
    fa = f(a)          # valor de f no extremo
    while b-a>eps:
        m = (a+b)/2    # calcula o ponto médio
        fm = f(m)      # valor de f no ponto médio
        if fa*fm < 0:
            b = m
        else:
            a = m
            fa = fm    # manter invariante
  return m
```



# Análise da implementação melhorada

- Em cada iteração calculamos *f* apenas uma vez em vez de duas
- Pode ser significativo se *f* for uma expressão complexa

# Método de Newton

Partimos de:

uma função $f$;

a derivada $f'$;

uma aproximação inicial à raiz $x_0$;

uma tolerância $\epsilon > 0$.



# Método de Newton (cont.)

Supondo que

1. $x_0$ está "suficientemente perto" da raiz $x^*$; e
2. a função $f$ é "bem comportada"

então a sucessão de aproximações definida por

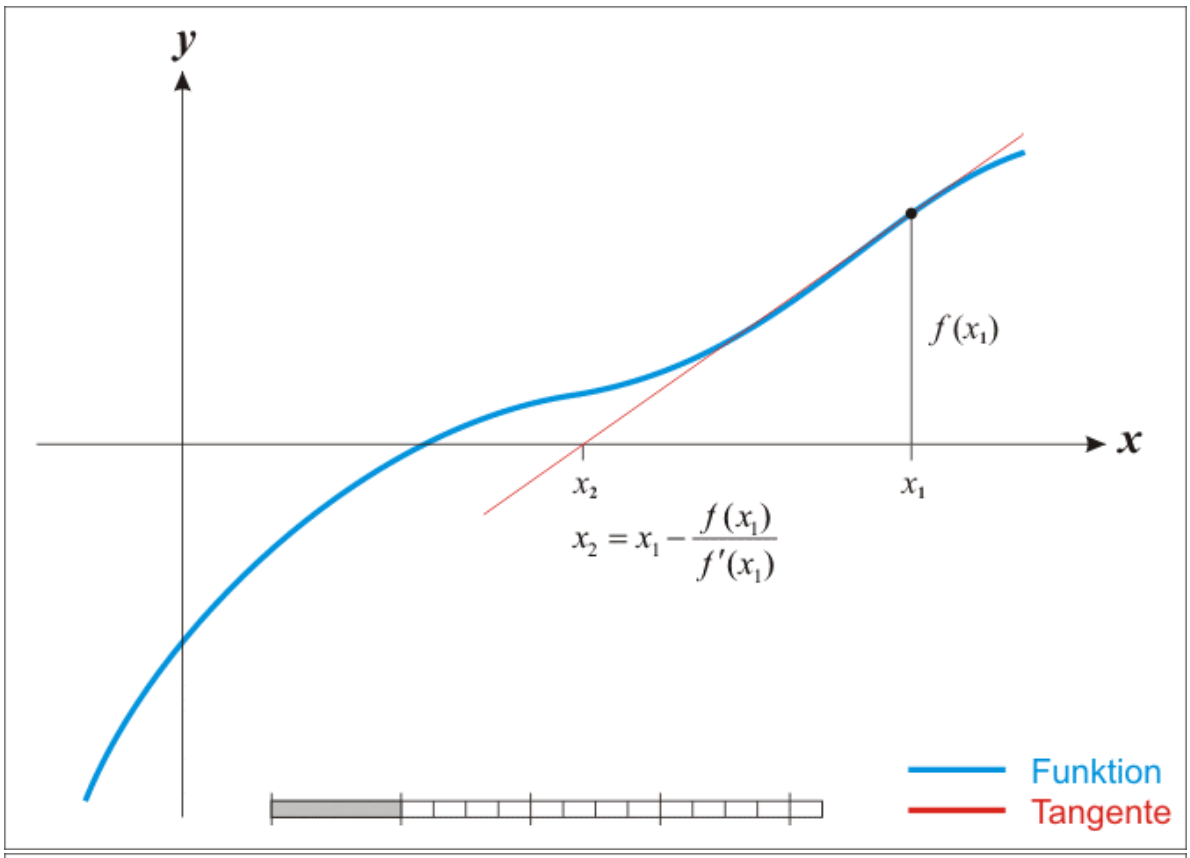
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

converge para a raiz $x^*$.

Paramos a aproximação quando se verifica um critério de paragem, por exemplo:

$$|x_n - x_{n-1}| < \epsilon$$

# Interpretação geométrica






# Implementação

```
def newton(f,df,x0,eps):
    '''Método de Newton com função f, derivada df,
       ponto inicial x0 e tolerância eps.'''
    x = x0
    fx = f(x0)          # valor da função
    dfx = df(x0)        # valor da derivada
    delta = fx/dfx      # correção da aproximação
    while abs(delta) > eps:
        x = x - delta
        fx = f(x)
        dfx = df(x)
        delta = fx/dfx
    return x
```

# Exemplo

Encontrar uma raiz de $x^3 - 5x + 2 = 0$.

Função:

$$F(x) = x^3 - 5x + 2$$

Derivada:

$$F'(x) = 3x^2 - 5$$



# Exemplo (cont.)

Em Python:

```
>>> def F(x): return x**3-5*x+2
>>> def dF(x): return 3*x**2-5
```

Encontrar a raiz partindo de $x_0 = 0$ com erro $< 10^{-8}$:

```
>>> newton(F, dF, 0, 1e-8)
0.4142135615572082
```

# Falha de convergência

O método de Newton pode não convergir se certas condições não se verificarem.

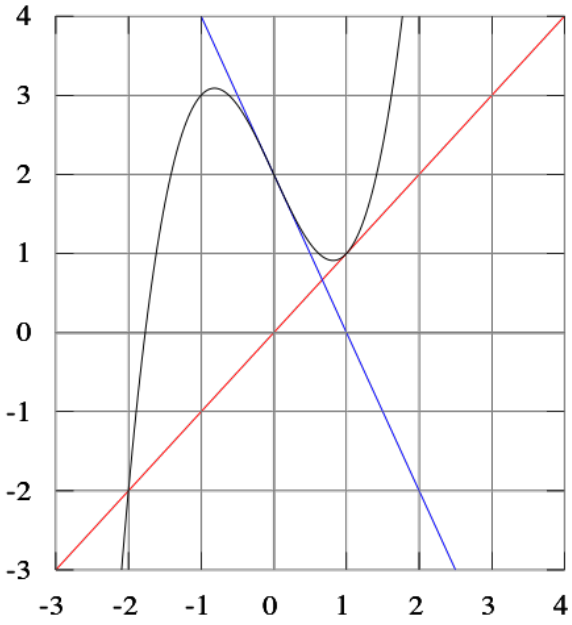
`http://en.wikipedia.org/wiki/Newton's_method`

Exemplo 1: se $f(x) = 1 - x^2$ e tomarmos $x_0 = 0$ então $f'(x_0) = 0$; logo $x_1$ é indefinido (divisão por zero).

Exemplo 2: se $f(x) = x^3 - 2x + 2$ e $x_0 = 0$ então $x_1 = 1, x_2 = 0, x_3 = 1, x_4 = 0, \ldots$ e o método diverge.



# Falha de convergência (cont.)




$f(x) = x^3 - 2x + 2$