Análise Numérica

Raízes e otimização

Renato Martins Assunção

DCC - UFMG

2012

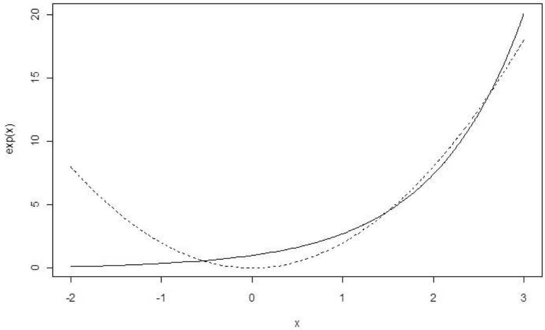
Método de Newton

- Queremos encontrar uma raiz x^* da equação não-linear $f(x) = 0$.
- Vimos o método da bisseção:
  - Acha uma raiz com certeza se começarmos com um intervalo onde a função $f(x)$ alterna seu sinal.
  - Desvantagem: Não podemos generalizar para funções que dependem de mais de uma variável tal como $f(x, y) = 0$.
- *Método de Newton* não tem este problema: pode ser aplicado em funções com qualquer dimensão

$$X_{n+1} = X_n - \frac{f(X_n)}{f'(X_n)}$$

Exemplo

- Ache as raízes de $f(x) = e^x - 2x^2$.
- Se $f(x) = 0$ então $e^x = 2x^2$.
- Uma parábola corta uma exponencial. Esboce o gráfico das duas curvas $\rightarrow$ existem três raízes.



Exemplo

- Ache as raízes de $f(x) = e^x - 2x^2$.
- Começando com $x_0 = 0$ encontramos

$$
\begin{aligned}
x_1 &= x_0 - f(x_0)/f'(x_0) = 0.333333, \\
x_2 &= x_1 - f(x_1)/f'(x_1) = 0.357246, \\
x_3 &= x_2 - f(x_2)/f'(x_2) = 0.357402, \\
x_4 &= x_3 - f(x_3)/f'(x_3) = 0.357402.
\end{aligned}
$$

- Começando com $x_0 = 5$ $\rightarrow\rightarrow\rightarrow$

$$
\begin{aligned}
x_1 &= x_0 - f(x_0)/f'(x_0) = 4.110793, \\
x_2 &= x_1 - f(x_1)/f'(x_1) = 3.329113, \\
x_3 &= x_2 - f(x_2)/f'(x_2) = 2.718701, \\
x_4 &= x_3 - f(x_3)/f'(x_3) = 2.334689, \\
x_5 &= 2.178594, \\
x_6 &= 2.153872, \\
x_7 &= 2.153292, \\
x_8 &= 2.153292.
\end{aligned}
$$

- Fica como exercício começar com $x_0 = -1$ e a achar a 3ª raiz.

Mais um exemplo

- Use o método de Newton para minimizar $f(x) = 0.5 - x\exp(-x^2)$.
- Primeira e segunda derivadas de f são dadas por

$$f'(x) = (2x^2 - 1)\exp(-x^2)$$

e

$$f''(x) = 2x(3 - 2x^2)\exp(-x^2)$$

- Iteração de Newton para zero de f' é dada por

$$x_{k+1} = x_k - (2x_k^2 - 1)/(2x_k(3 - 2x_k^2))$$

- Usando o chute inicial $x_0 = 1$, obtemos

x_k	$f(x_k)$
1.000	0.132
0.500	0.111
0.700	0.071
0.707	0.071

Um outro exemplo

- Use o método de Newton para encontrar um valor aproximado para $\sqrt{8}$.
- Vamos pensar numa função $f(x)$ tal que $f(x) = 0$ tenha solução $x = \sqrt{8}$.
- Tome $f(x) = x^2 - 8$.
- Equação de iteração:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} = x_n - \frac{x_n^2 - 8}{2x_n}$$

- Vamos começar com algo que deve ser próximo da solução: $x_0 = 3 = \sqrt{9}$.
- Assim, $x_1 = 2.83$ e $x_2 = 2.8294$, e etc...

Dificuldade com Newton

- Uma desvantagem do método de Newton é a necessidade de calcular a derivada no denominador da equação de iteração:

$$X_{n+1} = X_n - f(X_n)/f'(X_n)$$

- Isto pode parecer pouca coisa mas, de fato, não é.
- O que acontece é que, no caso **UNIDIMENSIONAL**, que estamos estudando até aqui, calcular a derivada realmente é muito simples.
- Entretanto, ela passa a ser muito mais complicada num dos casos mais importantes do uso do método de Newton que é o seguinte:
  - Estamos buscando o **MÁXIMO** de uma função de várias variáveis.
  - Neste caso, calcular a derivada no denominador vai significar, na verdade, encontrar a inversa da matriz de derivadas segundas, um cálculo que pode ser muito custoso no processo iterativo.

Método das secantes

- A ideia do método das secantes é substituir a derivada no denominador da equação de iteração

$$X_{n+1} = X_n - f(X_n)/f'(X_n)$$

  por uma aproximação simples, que não requeira nada além da avaliação da própria função.
- Se x_{n-1} e x_n não estão muito distantes entre si, podemos aproximar

$$f'(x_n) = \lim_{\Delta \to 0} \frac{f(x_n + \Delta) - f(x_n)}{\Delta} \approx \frac{f(x_{n-1}) - f(x_n)}{x_{n-1} - x_n} = s_n$$

- Assim, a equação de iteração torna-se

$$X_{n+1} = X_n - f(X_n)/s_n$$

Método das secantes

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

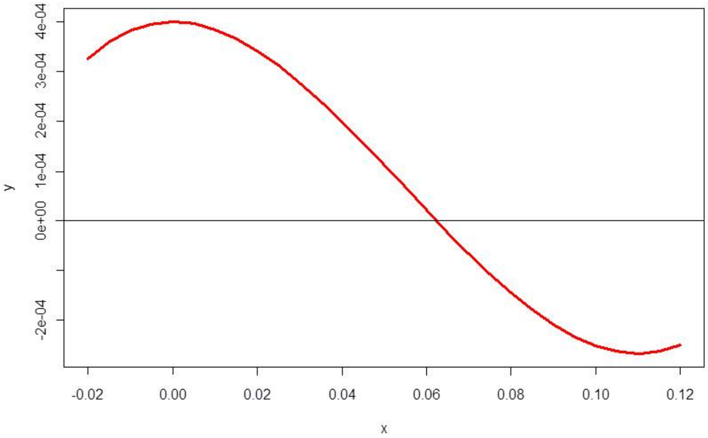
- O nome de método das secantes é devido ao uso da inclinação da reta secante que passa por $(x_n, f(x_n))$ e $(x_{n-1}, f(x_{n-1}))$ ao invés da derivada $f'(x_n)$
- Note que o método das secantes necessita de:
  - de DOIS valores iniciais x_0 e x_1 para começar o processo encontrando x_2.
  - OU ENTÃO de um único valor inicial x_0 e um chute para a derivada $f'(x_0)$.

Geometria

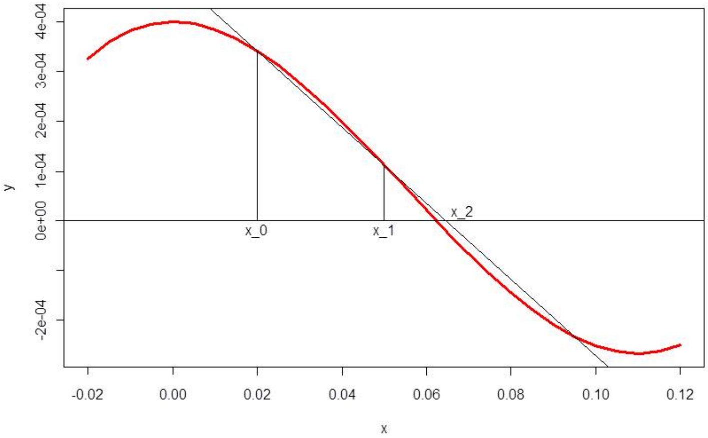
- Relembre Newton:
  - Aproxime a função $f(x)$ pela reta tangente em x_n.
  - Encontre a raiz desta aproximação linear e isto é x_{n+1}.
- Secante:
  - Aproxime a função pela reta que passa por $(x_{n-1}, f(x_{n-1}))$ e por $(x_n, f(x_n))$.
  - Encontre a raiz desta aproximação linear e isto é x_{n+1}.
- Equação da reta que passa por $(x_{n-1}, f(x_{n-1}))$ e por $(x_n, f(x_n))$ é dada por:

$$y = f(x_n) + s_n(x - x_n) \text{ (verifique isto)}$$

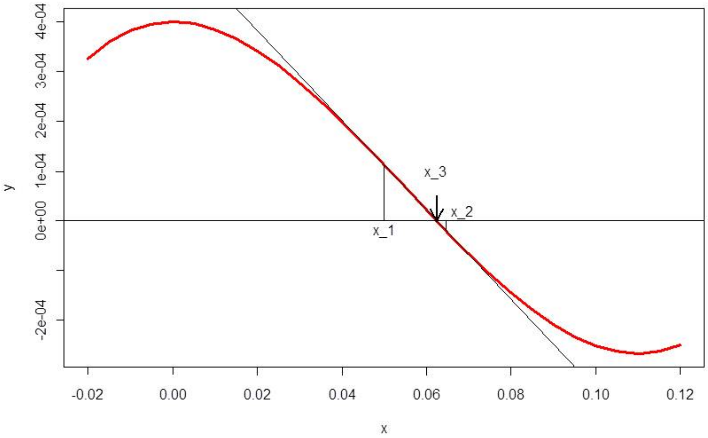
- Encontre a raiz: $0 = f(x_n) + s_n(x - x_n) \rightarrow x = x_n - f(x_n)/s_n$
- Esta é exatamente a equação de iteração do método da secante.



$$f(x) = x^3 - 0.165x^2 + 3.993 * 10 - 4$$



$x_0 = 0.02$ e $x_1 = 0.05 \rightarrow x_2 = 0.05 - f(0.05)/s_1 = 0.06461$
$100\% * |x_2 - x_1|/|x_2| = 22.61\%$

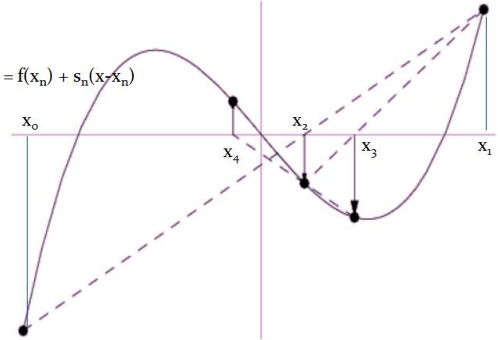


$x_1 = 0.05$ e $x_2 = 0.06461 \rightarrow x_3 = 0.06461 - f(0.06461)/s_2 = 0.06241$
$100\% * |x_3 - x_2|/|x_3| = 3.52\%$

Outra fórmula para a iteração

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

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



Intuição da nova fórmula

- Equação de iteração do método da secante

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

ou

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

onde os pesos w_n e w_{n-1} são dados por
$w_n = \dfrac{f(x_n)}{f(x_n) - f(x_{n-1})}$ e $w_{n-1} = \dfrac{f(x_{n-1})}{f(x_n) - f(x_{n-1})}$.

- INTUIÇÃO: x_{n+1} estará mais próximo do extremo cuja imagem for menor (em valor absoluto) e portanto esperamos que esteja mais próximo da raiz.

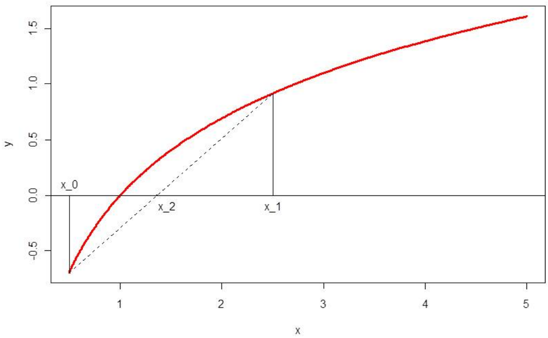
Convergência: rapidez

- Newton: $e_{n+1} \approx C * e_n^2$.
- Prova-se que o método da secante tem uma velocidade de convergência menor: $e_{n+1} \approx C * e_n^p$
- onde $p = (1 + 51/2)/2 \approx 1.618$ (a razão áurea).
- Apesar da ordem de convergência ser menor do que a do método de Newton, o método da secante não requer o cálculo e avaliação da derivada.

Falsa Posição ou Regula Falsi

- O método da secante pode não funcionar bem se a função for muito diferente de uma função linear no intervalo inicial.
- Uma possível correção é o método da regula falsi.
- Ele é só uma variação do método das secantes.
- A cada passo, mantemos dois valores x_n e x_{n-1} tais que os sinais de $f(x)$ sejam diferentes.
- Isto garante que existirá uma raiz entre os dois pontos.
- Para entender a motivação para esta variação, veja o exemplo a seguir.

Primeira iteração - secante

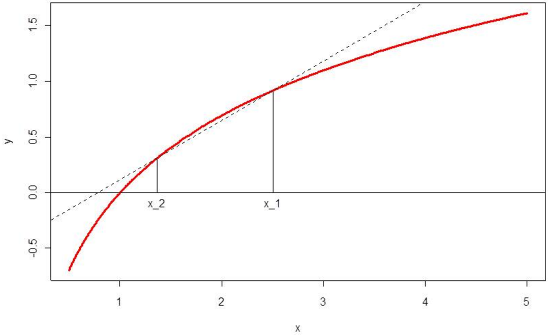


- $f(x) = \log(x)$

$$\begin{aligned} x_0 &= 0.5 \\ x_1 &= 2.5 \\ x_2 &= 1.36 \end{aligned}$$

- Qual é o próximo valor x_3?

Segunda iteração - secante

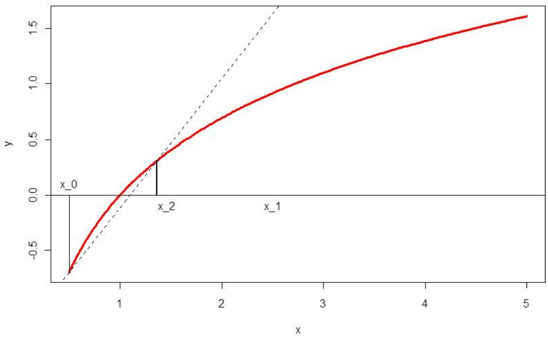


- $f(x) = \log(x)$

$$
\begin{aligned}
x_0 &= 0.5 \\
x_1 &= 2.5 \\
x_2 &= 1.36 \\
x_3 &= 0.783
\end{aligned}
$$

- Veja que os extremos $f(x_1)$ e $f(x_2)$ possuem o MESMO sinal.

O que aconteceria se ...

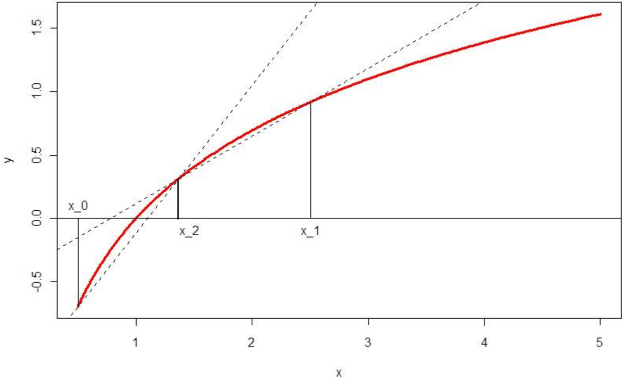


- $f(x) = \log(x)$

$$\begin{aligned} x_0 &= 0.5 \\ x_1 &= 2.5 \\ x_2 &= 1.36 \\ x_3 &= 0.783 \end{aligned}$$

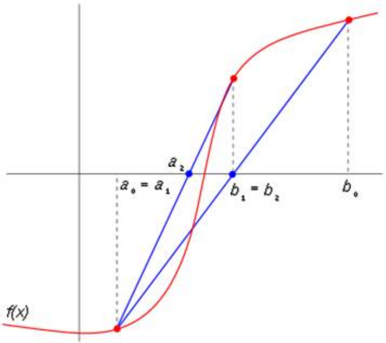
- Para calcular x_3, vamos usar x_0 e x_2: $\rightarrow f(x_0) \cdot f(x_2) < 0$

Comparando ...



Neste caso, o novo valor ficou mais próximo da raiz mas isto não ocorre sempre e nem é esta a razão para usar falsa posição.

O que aconteceria se ...



- Esta é a VERDADEIRA razão para o método de falsa posição .
- Se usarmos os pontos b_1 e b_0 não faremos um bom negócio.
- Podemos divergir para $-\infty$.
- Usar a_0 e b_1 para obter a próxima aproximação é claramente melhor.
- Veja os sinais de f.

Falsa posição

- A ideia é gerar as aproximações garantindo que elas sempre estejam num intervalo cujos extremos tenham valores de f com sinais trocados.
- Calcule o novo ponto da forma usual (como no método da secante)

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

- Se $f(x_{n+1}) * f(x_{n-1}) < 0 \rightarrow x_n = x_{n-1}$ e prossiga.

Ordem de convergência

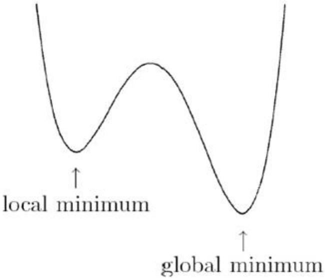
- Se f é uma função contínua em um intervalo $[a; b]$ e se $f(a) * f(b) < 0$ então o método da Falsa Posição converge.
- Igual ao método das secantes

$$p = (1 + \sqrt{5})/2 \approx 1.618$$

- Podem ocorrer problemas de *overflow*, além de convergência demorada devido a escolha do intervalo.

Otimização de funções

- Um dos principais usos dos algoritmos para encontrar raiz é para encontrar máximo ou mínimo de funções não lineares $\rightarrow$ otimização de funções.
- Encontrar o ponto x^* tal que $f(x^*) = \min_x \{f(x)\}$
- Este ponto x^* é um ponto de mínimo global.
- A função pode ter também mínimos locais.



Otimização global

- Em geral, encontrar um mínimo global não é uma tarefa fácil.
- Simplesmente verificar que certo ponto x^* é um mínimo global é também muito difícil.
- A maioria dos métodos de otimização buscam encontrar um mínimo local, sem nenhuma garantia de que foi encontrado um mínimo global.
- Se quisermos obter um mínimo global:
  - Tentamos vários valores iniciais bem diferentes.
  - Cada um deles convergirá para um mínimo ~~global~~ **local**.
  - Espera-se que o mínimo global esteja entre eles.

Mínimo ou máximo?

- Se tivermos um método para achar pontos de mínimo $\rightarrow$ também temos um método para achar pontos de máximo.
- A razão:
$$\max_x \{f(x)\} = -\min_x \{-f(x)\}$$
- Assim, encontramos x^*, o ponto de mínimo de $-f(x)$.
- Este será tambem o ponto de máximo de $f(x)$.

Otimização e pontos críticos

- Um dos principais usos dos algoritmos para encontrar raiz é para encontrar máximo ou mínimo de funções não lineares $\rightarrow$ otimização de funções.
- Encontrar o ponto x^* tal que $f(x^*) = \min_x \{f(x)\}$.
- Se $f(x)$ é função derivável até a segunda ordem podemos procurar a RAIZ da equação $f'(x) = 0$.
- Isto é, chame $g(x) = f'(x)$ e ache a raiz de $g(x) = 0$.
- Este é um ponto critico de $f(x)$ e pode ser:
  - Um ponto de mínimo se $f''(x^*) > 0$
  - Um ponto de máximo se $f''(x^*) < 0$
  - Um ponto de inflexão se $f''(x^*) = 0$

Newton - otimização

- Um dos métodos mais usados para otimização é baseado no método de Newton para encontrar raízes.
  - Comece com um valor inicial x_0
  - Itere:

$$\begin{aligned} x_{n+1} &= x_n - g(x_n)/g'(x_n) \\ &= x_n - f'(x_n)/f''(x_n) \end{aligned}$$

- ASSIM: Equação de iteração:

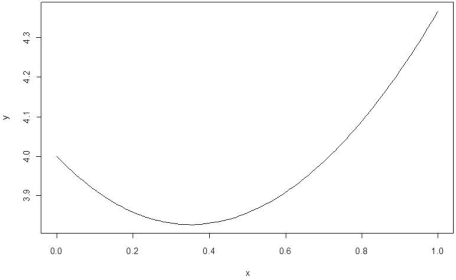
$$x_{n+1} = x_n - f'(x_n)/f''(x_n)$$

- Note que, para obter o máximo ou mínimo precisamos:
  - da derivada primeira
  - da derivada segunda

Exemplo

- Achar o ponto de mínimo de $f(x) = x^2 + \exp(-x) + 3$
- $x_{n+1} = x_n - f'(x_n)/f''(x_n) = x_n - (2x_n - \exp(-x_n))/(2 + \exp(-x_n))$

  - Começando com $x_0 = 0$
  - 0.0000000
  - 0.0.3333333
  - 0.3516893
  - 0.3517337
  - 0.3517337



Newton para sistemas não-lineares

Exemplo

Agora queremos resolver um sistema não-linear tal como

$$\begin{aligned} f(t,\alpha) &= t - 1 + t\cos\alpha = 0 \\ g(t,\alpha) &= 1 - e^{-t} - t\sin\alpha + \frac{t^2}{10} = 0 \end{aligned}$$

Usando notação vetorial, nosso sistema não-linear acima pode ser escrito como

$$\mathbf{f}(\mathbf{x}) = \mathbf{0},$$

em que $\mathbf{x} = [t,\alpha]^T$ e $\mathbf{f} = [f,g]^T$.

- O método mais usado para resolver este sistema é o de Newton, uma versão multidimensonal do que vimos antes onde $f = (f_1,\ldots,f_m)^t$ e cada f_i é função de $x_1,\ldots,x_m$.

Newton para sistemas não-lineares

- A equação de iteração no caso unidimensional é $x_{n+1} = x_n - f(x_n)/f'(x_n)$.
- Ela foi obtida encontrando a raiz da RETA TANGENTE à curva $f(x)$ passando por $(x_n, f(x_n))$.
- No caso multidimensional, a equação acima tem uma versão matricial.
- Seja J a **matriz Jacobiano** do sistema $f = (f_1, \ldots, f_m)^t$ onde

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_m} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_m} \end{bmatrix}$$

Newton para sistemas não-lineares

- Caso unidimensional é $x_{n+1} = x_n - f(x_n)/f'(x_n)$.
- A matriz J faz o papel da derivada $f'(x_n)$

$$\mathrm{J} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_m} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_m} \end{bmatrix}$$

Se simbolicamente escrevemos $\mathbf{f}'$ em vez de J, então a iteração de Newton torna-se

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - \underbrace{\left[\mathbf{f}'(\mathbf{x}^{(n)})\right]}_{\text{matriz}}^{-1} \mathbf{f}(\mathbf{x}^{(n)})$$

que se parece com a fórmula de iteração de Newton para o caso de equação única/variável única.

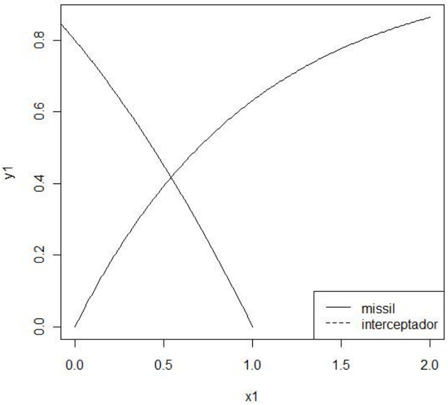
Um exemplo

Considere um míssil M seguindo o caminho parametrizado

$$x_M(t) = t,\; y_M(t) = 1 - e^{-t},$$

e um interceptador de míssil I cujo ângulo de lançamento α queremos determinar para que intercepte o caminho do míssil. O caminho parametrizado para o interceptador é dado por

$$x_I(t) = 1 - t\cos\alpha,\; y_I(t) = t\sin\alpha - \frac{t^2}{10}.$$



$\alpha = \pi/4$
Curvas são $(x(t), y(t))$ para cada objeto.

Sistema de equações não-lineares

$$x_M(t) = t, \; y_M(t) = 1 - e^{-t} \text{ Equação do míssil}$$

$$x_I(t) = 1 - t\cos\alpha, \; y_I(t) = t\sin\alpha - \frac{t^2}{10} \text{ Equação do interceptador}$$

Então temos que resolver o sistema de equações não-lineares

$$\begin{cases} t & = & 1 - t\cos\alpha \\ 1 - e^{-t} & = & t\sin\alpha - \dfrac{t^2}{10} \end{cases}$$

ou

$$\begin{cases} f(t,\alpha) & = & t - 1 + t\cos\alpha = 0 \\ g(t,\alpha) & = & 1 - e^{-t} - t\sin\alpha + \dfrac{t^2}{10} = 0 \end{cases}$$

Exemplo

Resolva o problema de interceptar o míssil

$$\begin{cases} t - 1 + t\cos\alpha = 0 \\ 1 - e^{-t} - t\sin\alpha + \dfrac{t^2}{10} = 0 \end{cases}$$

Aqui

$$\mathbf{f}(t,\alpha) = \begin{bmatrix} f_1(t,\alpha) \\ f_2(t,\alpha) \end{bmatrix} = \begin{bmatrix} t - 1 + t\cos\alpha = 0 \\ 1 - e^{-t} - t\sin\alpha + \dfrac{t^2}{10} = 0 \end{bmatrix}$$

e

$$J = (t,\alpha) = \begin{bmatrix} \dfrac{\partial f_1}{\partial t} & \dfrac{\partial f_1}{\partial \alpha} \\ \dfrac{\partial f_2}{\partial t} & \dfrac{\partial f_2}{\partial \alpha} \end{bmatrix} (t,\alpha) = \begin{bmatrix} 1 + \cos\alpha & -t\sin\alpha \\ e^{-t} - \sin\alpha + t/5 & -t\cos\alpha \end{bmatrix}.$$

Iterações

- A equação de iteração de Newton é:

$$x_{n+1} = x_n + J^{-1} * f'(x_n)$$

- Onde J é a matriz Jacobiano 2×2 e f é o vetor 2×1 com as derivadas parciais, tudo avaliado em x_n.
- Vamos começar com $x_0 = (t_0, \alpha_0) = (0.1, \pi/10)$.
- Ver código em `SciLab`.

Exemplo

Resolva

$$\begin{cases} x^2 + y^2 = 4 \\ xy = 1 \end{cases}$$

que corresponde a encontrar pontos de interseção de um círculo e uma hiperbole no plano. Aqui

$$\mathbf{f}(x,y) = \begin{bmatrix} f_1(x,y) \\ f_2(x,y) \end{bmatrix} = \begin{bmatrix} x^2 + y^2 - 4 \\ xy - 1 \end{bmatrix}$$

e

$$J = (t,\alpha) = \begin{bmatrix} \dfrac{\partial f_1}{\partial x} & \dfrac{\partial f_1}{\partial y} \\ \dfrac{\partial f_2}{\partial x} & \dfrac{\partial f_2}{\partial y} \end{bmatrix} (t,\alpha) = \begin{bmatrix} 2x & 2y \\ y & x \end{bmatrix}.$$

Observação

1. Método de Newton requer do usuário a matriz Jacobiano $m \times m$ (que depende de um sistema não-linear específico a ser resolvido). Isto é bastante complicado.
2. Em cada iteração um sistema linear $m \times m$ (denso) tem que ser resolvido. Isto faz do método de Newton muito caro e lento.
3. Para "bons" valores iniciais, o método de Newton converge quadraticamente para zeros simples, i.e., soluções para qual $J^{-1}(\mathbf{z})$ exista.