Métodos de busca linear

Marina Andretta

ICMC-USP

11 de agosto de 2014

Baseado no livro Numerical Optimization, de J. Nocedal e S. J. Wright.

Busca linear

Veremos agora métodos de busca linear para resolução de problemas de minimização irrestrita, ou seja,

$$\boxed{\text{Minimizar} \quad f(x)} \tag{1}$$

onde

- $x \in \boldsymbol{R}^n$;
- $f \in \boldsymbol{R}^n \to \boldsymbol{R}$ uma função suave.

Busca linear

Em cada iteração de um método de busca linear, calcula-se uma direção p_k e então decide-se o quanto se deve mover ao longo desta direção.

A iteração é dada por

$$x_{k+1} = x_k + \alpha_k p_k.$$

onde α_k é um escalar positivo chamado de tamanho de passo.

O sucesso de um método de busca linear depende tanto da escolha da direção p_k como da escolha do tamanho de passo α_k.

Busca linear

A maior parte dos algoritmos de busca linear exigem que p_k seja uma direção de descida. Isso significa que a direção deve satisfazer $p_k^T \nabla f_k < 0$, o que garante que o valor de f decresce ao longo desta direção.

Mais ainda, as direções de busca geralmente tem a forma

$$p_k = -B_k^{-1} \nabla f_k, \tag{2}$$

onde B_k é uma matriz simétrica e não-singular.

Busca linear

- No método de máxima descida, B_k é simplesmente a matriz identidade.
- No método de Newton, B_k é a Hessiana exata de $\nabla^2 f(x_k)$.
- Em métodos de quase-Newton, B_k é uma aproximação da Hessiana que é atualizada a cada iteração por uma fórmula de posto baixo.

Busca linear

Quando p_k é definida por (2) e B_k é definida positiva, temos que

$$p_k^T \nabla f_k = -\nabla f_k^T B_k^{-1} \nabla f_k < 0$$

e, portanto, p_k é uma direção de descida.

Tamanho de passo

Ao calcular o tamanho do passo, enfrentamos um dilema: por um lado, gostaríamos de escolher α_k de forma a obter uma diminuição substancial de f; por outro lado, não queremos gastar muito tempo fazendo esta escolha.

A escolha ideal seria o minimizador global da função de uma variável

$$\text{Minimizar}_{\alpha>0} \quad \phi(\alpha) = f(x_k + \alpha p_k), \tag{3}$$

mas, em geral, identificar esta solução é muito custoso.

Tamanho de passo

Mesmo encontrar um minimizador local para ϕ com uma precisão pequena pode requerer muitas avaliações da função objetivo f e, possivelmente, de seu gradiente ∇f.

Estratégias mais práticas fazem uma busca linear inexata para identificar um tamanho de passo que proporciona redução suficiente da função f a um custo mínimo.

Algoritmos de busca linear tipicamente testam uma sequência de candidatos a valores de α até que certas condições sejam satisfeitas.

Tamanho de passo

A busca linear é feita em duas fases:

1. Na primeira fase, encontra-se um intervalo que contém tamanhos de passo desejados.
2. Na segunda fase, bissecção ou interpolação são feitas para encontrar o valor de α contido no intervalo calculado na primeira fase.

Tamanho de passo

Antes de focar nos algoritmos para cálculo de α, vejamos quais critérios devem ser satisfeitos por α.

Uma condição simples que poderia ser imposta a α_k é que ela deve gerar decréscimo simples da função f, ou seja, que $f(x_k + \alpha_k p_k) < f(x_k)$. No entanto, isto não é suficiente.

Tamanho de passo

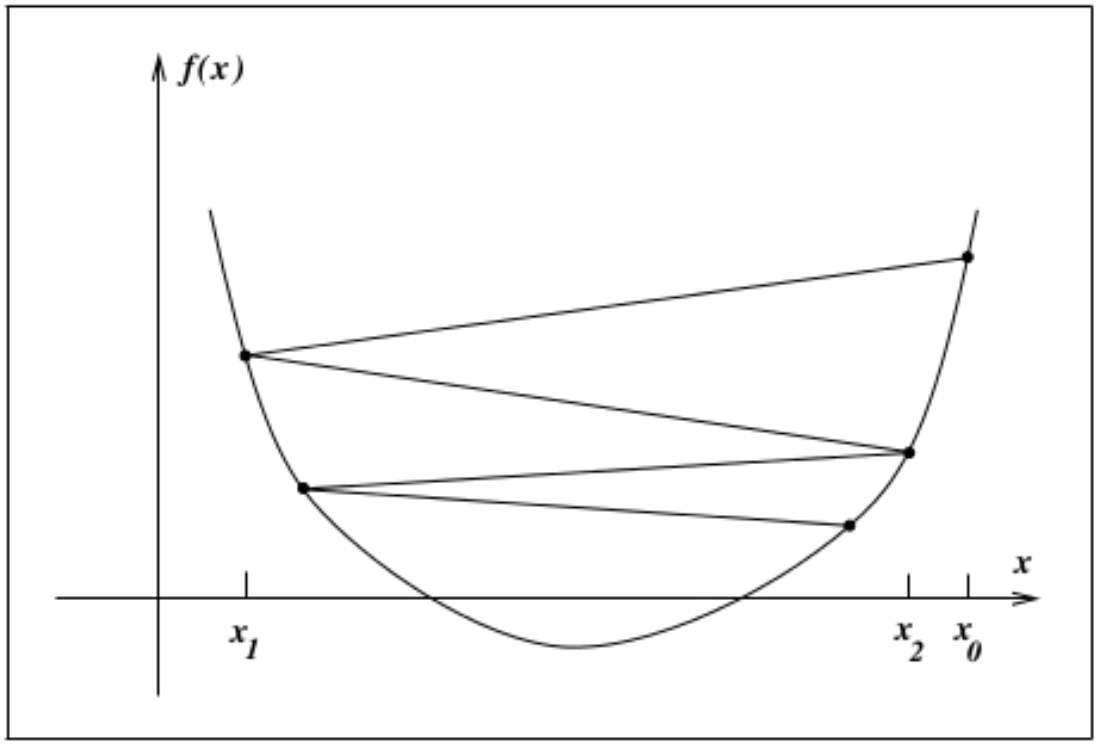

Figura : Exemplo com decréscimo simples (Figura 3.2 de Numerical Optimization, de J. Nocedal e S. J. Wright)

Condições de Wolfe

Uma condição popular para busca linear inexata estipula que α_k deve, antes de mais nada, fornecer decréscimo suficiente da função f, como medido pela desigualdade

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f_k^T p_k, \tag{4}$$

para uma constante $c_1 \in (0, 1)$. Ou seja, a redução de f deve ser proporcional tanto ao tamanho de passo α como à derivada direcional $\nabla f_k^T p_k$.

Condições de Wolfe

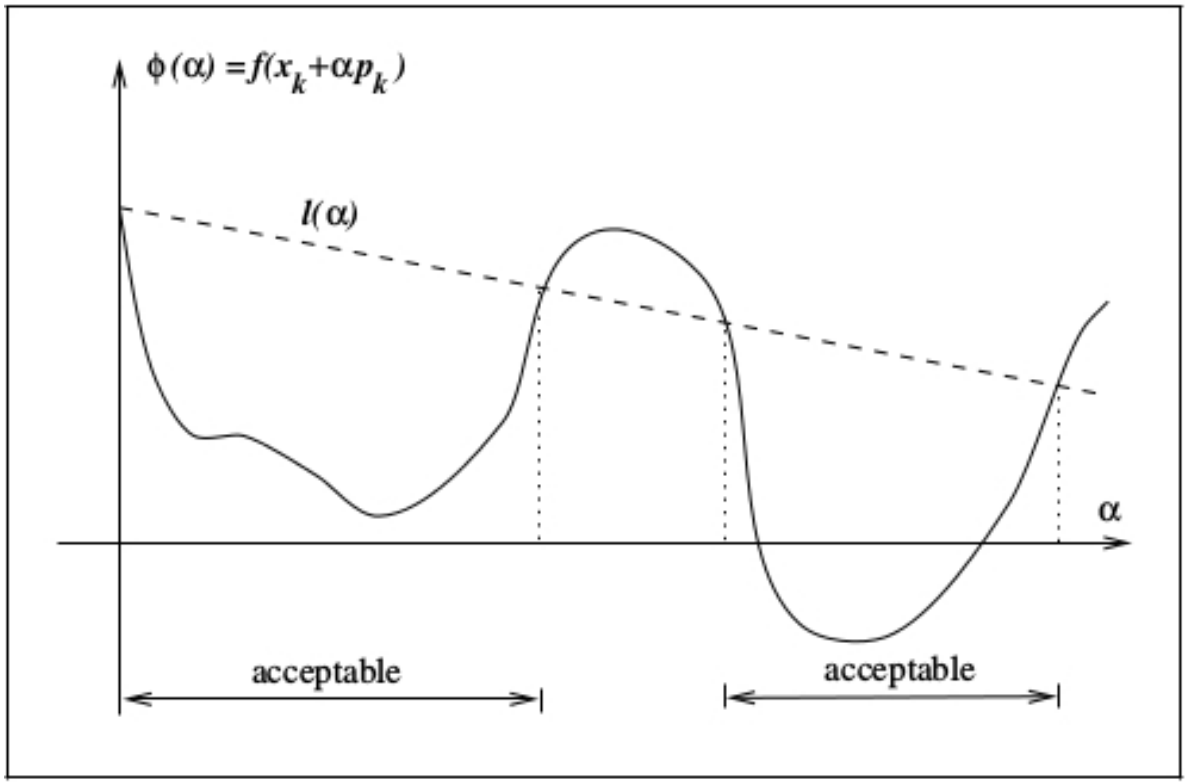

Figura : Exemplo com condições de Wolfe (Figura 3.3 de Numerical Optimization, de J. Nocedal e S. J. Wright)

Condições de Wolfe

A condição (4) é conhecida como condição de Armijo.

Na prática, c_1 é definido como um número pequeno, como $c_1 = 10^{-4}$.

Condições de Wolfe

A condição de decréscimo suficiente não é o bastante para garantir que o algoritmo faça progresso suficiente, pois ele é satisfeito para todos valores suficientemente pequenos de α.

Para eliminar tamanhos de passo inaceitavelmente pequenos, introduzimos uma segunda condição, chamada de condição de curvatura:

$$\nabla f(x_k + \alpha_k p_k)^T p_k \geq c_2 \nabla f_k^T p_k, \tag{5}$$

para uma constante $c_2 \in (c_1, 1)$ e c_1 a constante da condição (4).

Condições de Wolfe

Note que o lado esquerdo da inequação é a derivada $\phi'(\alpha_k)$. Então, a condição de curvatura garante que a inclinação de $\phi(\alpha_k)$ é maior do que c_2 multiplicado pelo gradiente $\phi'(0)$.

Isto faz sentido porque, se a inclinação $\phi'(\alpha)$ é muito negativa, temos uma indicação de que a f pode ser reduzida consideravelmente movendo-se bastante na direção p_k. Por outro lado, se a inclinação é pouco negativa ou até mesmo positiva, isto é um sinal de que não podemos esperar uma grande redução de f na direção p_k e, por isso, faz sentido terminar a busca linear.

Condições de Wolfe

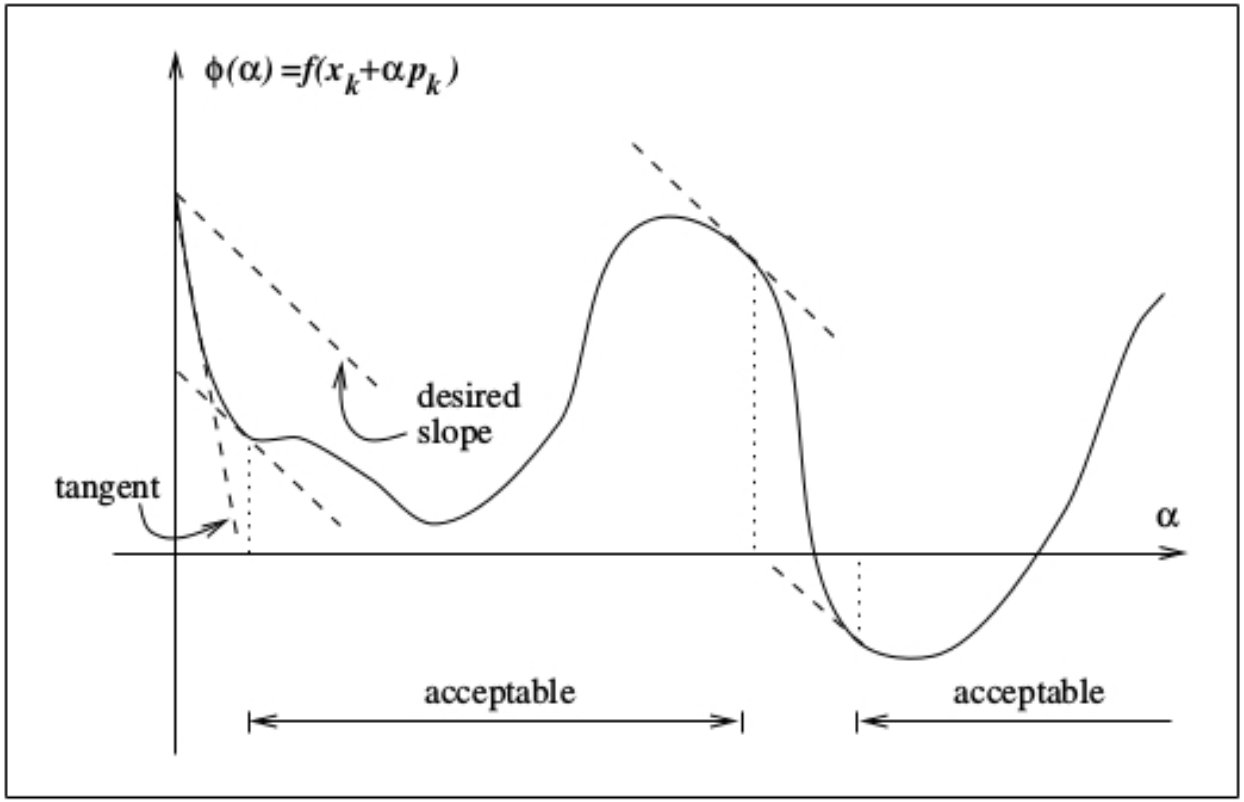

Figura : Exemplo com condições de Wolfe (Figura 3.4 de Numerical Optimization, de J. Nocedal e S. J. Wright)

Condições de Wolfe

Valores típicos de c_2 são 0.9, quando a direção de busca p_k é calculada pelo método de Newton ou quase-Newton e 0.1, quando p_k é calculada por um método de gradientes conjugados não-linear.

Condições de Wolfe

As condições de decréscimo suficiente e de curvatura são conhecidas, em conjunto, como condições de Wolfe:

$$\begin{array}{rcl} f(x_k + \alpha p_k) & \leq & f(x_k) + c_1 \alpha \nabla f_k^T p_k, \\ \nabla f(x_k + \alpha_k p_k)^T p_k & \geq & c_2 \nabla f_k^T p_k, \end{array} \tag{6}$$

para $0 < c_1 < c_2 < 1$.

Condições de Wolfe

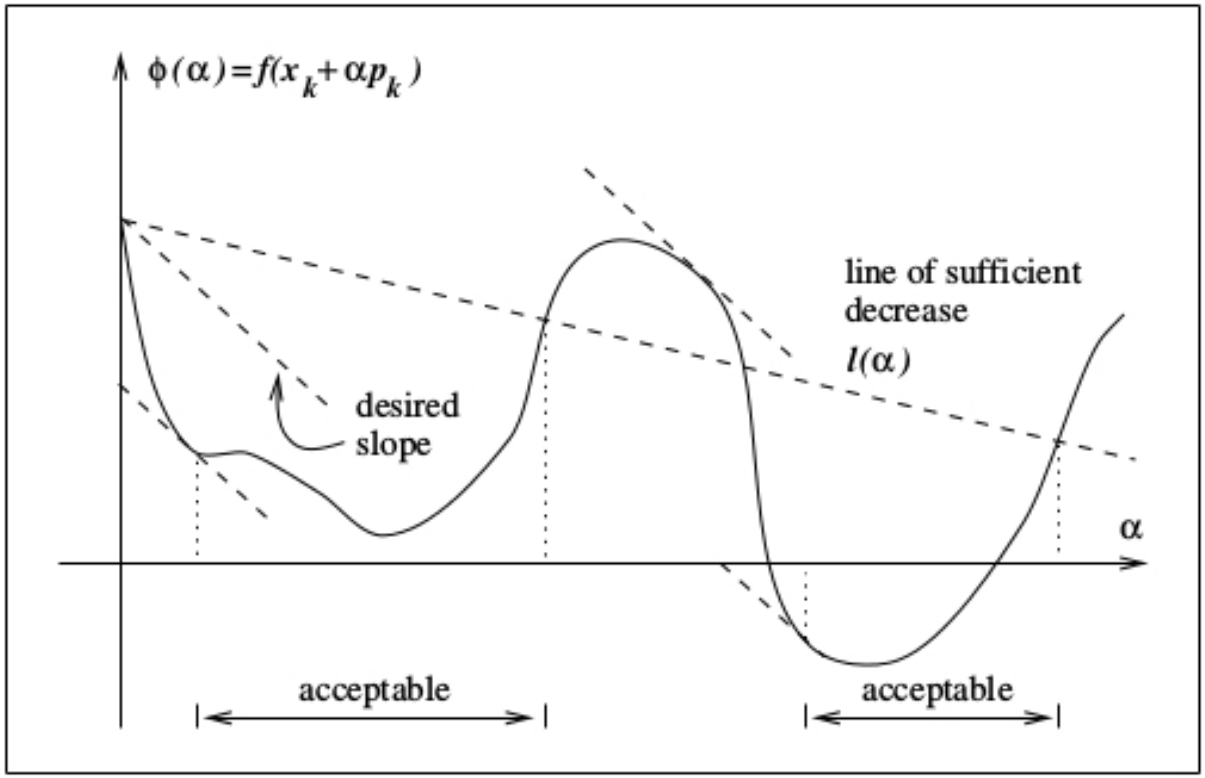

Figura : Exemplo com condições de Wolfe (Figura 3.5 de Numerical Optimization, de J. Nocedal e S. J. Wright)

Condições de Wolfe

Um tamanho de passo pode satisfazer as condições de Wolfe sem estar perto do minimizador de ϕ. No entanto, podemos modificar a condição de curvatura para forçar que α_k esteja em pelo menos uma larga vizinhança de um minimizador local ou um ponto estacionário de ϕ.

As condições fortes de Wolfe exigem que α_k satisfaça

$$\begin{aligned} f(x_k + \alpha p_k) &\leq f(x_k) + c_1 \alpha \nabla f_k^T p_k, \\ |\nabla f(x_k + \alpha_k p_k)^T p_k| &\leq c_2 |\nabla f_k^T p_k|, \end{aligned} \tag{7}$$

para $0 < c_1 < c_2 < 1$.

Condições de Wolfe

A única diferença entre as condições de Wolfe e as condições fortes de Wolfe é que estas não permitem que a derivada $\phi'(\alpha_k)$ seja muito positiva. Assim, são excluídos os pontos que estão longe dos pontos estacionários de ϕ.

É simples mostrar que existem tamanhos de passo que satisfazem as condições de Wolfe para toda função f suave e limitada inferiormente.

Condições de Wolfe

Lema 1: *Suponha que* $f : \boldsymbol{R}^n \to \boldsymbol{R}$ *possua primeira derivada contínua. Seja* p_k *uma direção de descida a partir de* x_k*. Suponha que* f *seja limitada inferiormente ao longo do raio* $\{x_k + \alpha p_k | \alpha > 0\}$*.*
Então, se $0 < c_1 < c_2 < 1$*, existem intervalos de tamanhos de passo que satisfazem as condições de Wolfe (6) e as condições fortes de Wolfe (7).*

Condições de Wolfe

As condições de Wolfe são invariantes quanto ao escalamento: multiplicar a função f por uma constante ou fazer uma mudança afim nas variáveis não as altera.

Estas condições podem ser usadas na maior parte dos métodos de busca linear e são particularmente importantes na implementação de métodos quase-Newton.

Decréscimo suficiente e *backtracking*

Como mencionado anteriormente, apenas a condição de decréscimo suficiente (4) não basta para garantir que o algoritmo obtenha progresso suficiente ao longo de uma dada direção.

No entanto, se o algoritmo de busca linear escolhe os candidatos a tamanho de passo de forma apropriada, usando o chamado *backtracking*, podemos ignorar a condição de curvatura (5) e usar apenas a condição de decréscimo suficiente para terminar o procedimento de busca linear.

Decréscimo suficiente e *backtracking*

Busca linear com backtracking: Dados $\bar{\alpha} > 0$, ρ, $c \in (0, 1)$.

Passo 1: Faça $\alpha \leftarrow \bar{\alpha}$.

Passo 2: Se $f(x_k + \alpha p_k) \leq f(x_k) + c\alpha \nabla f_k^T p_k$ então
faça $\alpha_k \leftarrow \alpha$ e pare.

Senão

faça $\alpha \leftarrow \rho\alpha$ e repita o Passo 2.

Decréscimo suficiente e *backtracking*

O tamanho de passo inicial $\bar{\alpha}$ é 1 quando um método de Newton ou quase-Newton é usado para calcular p_k, mas pode ter diferentes valores quando são usados outros métodos no cálculo de p_k, como máxima descida ou gradientes conjugados.

Um tamanho de passo aceitável será encontrado depois de um número finito de iterações, pois, em algum momento α, se tornará pequeno suficiente para satisfazer a condição de decréscimo suficiente.

Decréscimo suficiente e *backtracking*

Na prática, o fator de contração ρ pode variar em cada iteração da busca linear. Só é preciso garantir que, a cada iteração, $\rho \in [\rho_1, \rho_2]$ para constantes fixadas $0 < \rho_1 < \rho_2 < 1$.

O procedimento de *backtracking* garante que α_k será ou um valor $\bar{\alpha}$ ou um valor menor que garanta decréscimo suficiente, porém não muito pequeno (já que os valores de α vão diminuindo e é aceito o primeiro que satisfaz a condição de decréscimo suficiente).

Algoritmos para escolha de tamanho de passo

Vamos considerar agora algoritmos para encontrar um minimizador da função unidimensional

$$\phi(\alpha) = f(x_k + \alpha p_k),$$

ou para simplesmente encontrar um α_k que satisfaça alguma das condições vistas anteriormente.

Supomos que p_k é uma direção de descida, ou seja, $\phi'(0) < 0$, para que nossa busca se restrinja a valores positivos de α.

Algoritmos para escolha de tamanho de passo

Se f é uma função quadrática convexa $f(x) = \frac{1}{2}x^T Qx + b^T x + c$, seu minimizador ao longo de $x_k + \alpha p_k$ pode ser calculado analiticamente e é dado por

$$\alpha_k = -\frac{\nabla f_k^T p_k}{p_k^T Q p_k}.$$

Se a função f é uma função não-linear qualquer, é necessário usar um processo iterativo para o cálculo de α_k. Note que a eficiência deste processo impacta diretamente na eficiência do algoritmo para minimização de f.

Algoritmos para escolha de tamanho de passo

Todos os procedimentos de busca linear precisam de uma estimativa inicial α_0 e geram uma sequência $\{\alpha_i\}$ que termina com um passo que satisfaz as condições especificadas pelo usuário (por exemplo, as condições de Wolfe) ou determinam que tal tamanho de passo não existe.

Procedimentos típicos consistem em duas fases: determinar um intervalo $[a, b]$ que contém tamanhos de passo aceitáveis e buscar o tamanho de passo desejado neste intervalo.

Geralmente, a segunda fase diminui o intervalo que contém o tamanho de passo desejado e interpola algumas informações da função e da derivada obtidas em suas iterações anteriores para tentar descobrir qual o tamanho de passo buscado.

Interpolação

Vamos denotar por α_k e α_{k-1} os tamanhos de passo usados nas iterações k e $k-1$ do algoritmo de otimização. Denotaremos por α_i os valores intermediários de α durante o procedimento de busca linear. α_0 será a estimativa inicial da busca linear.

Vamos descrever um procedimento de busca linear baseado na interpolação de valores conhecidos da função ϕ e de sua derivada. Este procedimento é um melhoramento do procedimento de *backtracking* visto anteriormente.

O objetivo é encontrar um valor de α que satisfaça a condição de decréscimo suficiente (4) sem ser muito "pequeno". Este procedimento gera uma sequência α_i de valores decrescentes, com cada valor α_i não muito menor do que α_{i-1}.

Interpolação

Note que podemos escrever a condições de descréscimo suficiente como

$$\phi(\alpha_k) \leq \phi(0) + c_1 \alpha_k \phi'(0).$$

Como, na prática, c_1 é um número pequeno, o que se pede é um pouco mais do que descréscimo simples de f.

Interpolação

Suponha que uma estimativa inicial α_0 seja dada. Se temos que

$$\phi(\alpha_0) \leq \phi(0) + c_1\alpha_0\phi'(0),$$

este tamanho de passo satisfaz a condição de decréscimo suficiente e paramos o procedimento de busca linear com tamanho de passo $\alpha_k = \alpha_0$.

Interpolação

Caso contrário, sabemos que o intervalo $[0, \alpha_0]$ contém tamanhos de passo aceitáveis. Calculamos a aproximação quadrática $\phi_q(\alpha)$ de ϕ interpolando as três informações disponíveis: $\phi(0)$, $\phi'(0)$ e $\phi(\alpha_0)$.

Impondo que $\phi_q(0) = \phi(0)$, $\phi_q'(0) = \phi'(0)$ e $\phi_q(\alpha_0) = \phi(\alpha_0)$, obtemos

$$\phi_q(\alpha) = \left(\frac{\phi(\alpha_0) - \phi(0) - \alpha_0\phi'(0)}{\alpha_0^2}\right)\alpha^2 + \phi'(0)\alpha + \phi(0).$$

Interpolação

O novo valor α_1 é definido como o minimizador desta quadrática, ou seja,

$$\alpha_1 = -\frac{\phi'(0)\alpha_0^2}{2[\phi(\alpha_0)-\phi(0)-\alpha_0\phi'(0)]}.$$

Se a condição de decréscimo suficiente é satisfeita para α_1, paramos a busca linear com $\alpha_k = \alpha_1$.

Interpolação

Caso contrário, construímos a interpolação cúbica usando as informações $\phi(0)$, $\phi'(0)$, $\phi(\alpha_0)$ e $\phi(\alpha_1)$, obtendo

$$\phi_c(\alpha) = a\alpha^3 + b\alpha^2 + \phi'(0)\alpha + \phi(0),$$

com

$$\begin{bmatrix} a \\ b \end{bmatrix} = \frac{1}{\alpha_0^2\alpha_1^2(\alpha_1-\alpha_0)} \begin{bmatrix} \alpha_0^2 & -\alpha_1^2 \\ -\alpha_0^3 & \alpha_1^3 \end{bmatrix} \begin{bmatrix} \phi(\alpha_1) - \phi(0) - \phi'(0)\alpha_1 \\ \phi(\alpha_0) - \phi(0) - \phi'(0)\alpha_0 \end{bmatrix}.$$

Calculando a derivada de $\phi_c(x)$, temos que o minimizador α_2 de ϕ_c está no intervalo $[0, \alpha_1]$ e é dado por

$$\alpha_2 = \frac{-b+\sqrt{b^2-3a\phi'(0)}}{3a}.$$

Interpolação

Se α_2 satisfaz a condição de decréscimo suficiente, a busca linear pára com $\alpha_k = \alpha_2$. Caso contrário, continua-se fazendo a interpolação cúbica, sempre tomando os dois últimos valores de α_i, até que seja calculado um valor de α_i que satisfaça as condições de decréscimo suficiente.

Se algum dos α_i está muito próximo de seu predecessor α_{i-1} ou muito longe deste, redefinimos α_i como $\alpha_{i-1}/2$. Esta salvaguarda serve para garantirmos que um progresso razoável é obtido e que o α final não seja muito pequeno.

Interpolação

Em muitos casos, é mais interessante usar apenas a interpolação quadrática no cálculo de α_k. O procedimento é o mesmo apresentado aqui, mas sem que seja calculada a interpolação cúbica. Apenas a informação de ϕ no ponto anterior é necessária.

Este procedimento parte do princípio que calcular derivadas é custoso. Se este não for o caso, os valores das derivadas podem ser usados para obter melhor progresso no cálculo de α.

Há outros procedimento para o cálculo de α_k, que definem um intervalo $[a, b]$ no qual sabe-se que existem os tamanhos de passo procurados, buscam neste intervalo um valor de α que satisfaça as condições de Wolfe (por exemplo) e, a cada iteração, atualizam o intervalo $[a, b]$ para a realização de uma nova busca.

Tamanho de passo inicial

Para métodos de Newton e quase-Newton, o tamanho de passo $\alpha_0 = 1$ sempre deve ser usado como estimativa inicial. Isto garante que passos unitários serão aceitos sempre que satisfizerem as condições de parada, o que permite que as propriedades de convergência rápida destes métodos terão efeito.

Para métodos que não produzem passos bem-escalados, como o método de máxima descida e o de gradientes conjugados, é importante usar informações sobre o problema e sobre o algoritmo para definir a estimativa inicial para o tamanho de passo.

Tamanho de passo inicial

Uma estratégia popular é supor que a mudança de primeira ordem da função no iterando x_k será a mesma obtida pelo passo anterior. Ou seja, escolhemos a estimativa inicial α_0 tal que $\alpha_0 \nabla f_k^T p_k = \alpha_{k-1} \nabla f_{k-1}^T p_{k-1}$.

Portanto,

$$\alpha_0 = \alpha_{k-1} \frac{\nabla f_{k-1}^T p_{k-1}}{\nabla f_k^T p_k}.$$

Tamanho de passo inicial

Outra estratégia é calcular a interpolação quadrática dos dados $f(x_{k-1})$, $f(x_k)$ e $\phi'(0) = \nabla f_k^T p_k$ e definir α_0 como o minimizador desta quadrática.

Ou seja,

$$\alpha_0 = \frac{2(f_k - f_{k-1})}{\phi'(0)}.$$

Convergência de métodos de busca linear

Para obter convergência global de métodos de busca linear, além de um tamanho de passo bem escolhido, precisamos que a direção de busca p_k satisfaça algumas propriedades.

A propriedade chave que a direção de busca p_k deve satisfazer está relacionada ao ângulo θ_k entre p_k e a direção de máxima descida $-\nabla f_k$, definido por

$$\cos\theta_k = \frac{-\nabla f_k^T p_k}{\|\nabla f_k\|\|p_k\|}. \tag{8}$$

Convergência de métodos de busca linear

O teorema a seguir foi mostrado por Zoutendijk. Ele mostra, por exemplo, que o método de máxima descida é globalmente convergente. Para outros algoritmos, ele descreve o quanto p_k pode se desviar de ∇f_k e ainda gerar uma iteração globalmente convergente.

Vários critérios de parada para a busca linear podem ser usados para mostrar este resultado, mas, para analisar um caso concreto, serão consideradas apenas as condições de Wolfe (6).

Convergência de métodos de busca linear

Teorema 1: *Considere uma iteração da forma* $x_{k+1} = x_k + \alpha_k p_k$, *onde* p_k *é uma direção de descida e* α_k *satisfaz as condições de Wolfe (6). Suponha que* f *seja limitada inferiormente em* $\mathbf{R}^n$ *e que* f *tenha primeira derivada contínua em um conjunto aberto* $\mathcal{N}$ *que contém o conjunto de nível* $\mathcal{L} = \{x | f(x) \leq f(x_0)\}$, *com* x_0 *ponto inicial da iteração. Suponha também que o gradiente* ∇f *seja Lipschitz contínuo em* $\mathcal{N}$, *ou seja, existe uma constante* $L > 0$ *tal que* $\|\nabla f(x) - \nabla f(\tilde{x})\| \leq L\|x - \tilde{x}\|$, $\forall x, \tilde{x} \in \mathcal{N}$.
Então,

$$\sum_{k \geq 0} \cos^2 \theta_k \|\nabla f_k\|^2 < \infty. \tag{9}$$

Convergência de métodos de busca linear

Note que as hipóteses do Teorema 1 não são muito restritivas:

- Se a função f não é limitada inferiormente, o problema de otimização não está bem definido.
- As hipóteses de suavidade (gradiente Lipschitz contínuo) são implicadas por várias condições de suavidade usadas em teoremas de convergência local e, comumente, são satisfeitas na prática.

Convergência de métodos de busca linear

A desigualdade (9), chamada de condição de Zoutendijk, implica que

$$\cos^2 \theta_k \|\nabla f_k\|^2 \to 0. \tag{10}$$

Este limite pode ser usado para derivar resultados de convergência global para algoritmos de busca linear.

Convergência de métodos de busca linear

Se nosso método para escolher a direção de busca p_k na iteração $x_{k+1} = x_k + \alpha_k p_k$ garante que o ângulo θ_k definido por

$$\cos\theta_k = \frac{-\nabla f_k^T p_k}{\|\nabla f_k\|\|p_k\|}$$

está longe de 90°, existe uma constante positiva tal que

$$\cos\theta_k \geq \delta > 0, \quad \forall k.$$

Convergência de métodos de busca linear

Portanto, segue imediatamente de (10) que

$$\lim_{k\to\infty} \|\nabla f_k\| = 0. \tag{11}$$

Em outras palavras, podemos garantir que a sequência de normas dos gradientes converge para zero se as direções de busca nunca estão muito perto da ortogonalidade com o gradiente.

Em particular, o método de máxima descida, para o qual o ângulo entre a direção de busca e o gradiente é zero, produz uma sequência de gradientes que converge para zero, dado que ele use uma busca linear que satisfaça as condições de Wolfe.

Convergência de métodos de busca linear

Para métodos de busca linear da forma geral $x_{k+1} = x_k + \alpha_k p_k$, o limite (11) é resultado de convergência global mais forte que se pode obter: não podemos garantir que o método converge para um minimizador, mas somente que ele é atraído por pontos estacionários.

Somente exigindo mais condições de p_k (por exemplo, introduzindo informação de curvatura negativa da Hessiana) podemos deixar esses resultados mais fortes incluindo a convergência a minimizadores locais.

Convergência de métodos de busca linear

Considere agora métodos do tipo Newton com iterações do tipo $x_{k+1} = x_k + \alpha_k p_k$, $p_k = -B_k^{-1}\nabla f_k$. Suponha que as matrizes B_k sejam definidas positivas com um número de condição uniformemente limitado. Ou seja, existe uma constante M tal que

$$\|B_k\|\|B_k^{-1}\| \leq M, \quad \forall k.$$

É fácil mostrar, pela definição (8), que

$$\cos\theta_k \geq 1/M.$$

Convergência de métodos de busca linear

Combinando este limitante com o limite (10), temos que

$$\lim_{k\to\infty} \|\nabla f_k\| = 0.$$

Portanto, temos que os métodos de Newton e quase-Newton são globalmente convergentes se as matrizes B_k são definidas positivas (necessário para que a direção p_k seja de descida) e possuem número de condição limitado.

Convergência de métodos de busca linear

Para alguns algoritmos, como métodos de gradientes conjugados, não é possível mostrar que vale o limite (11), mas sim o resultado mais fraco

$$\liminf_{k\to\infty} \|\nabla f_k\| = 0.$$

Em outras palavras, apenas uma subsequência das normas dos gradientes converge para zero, em vez da sequência toda. Este resultado também pode ser mostrado usando a condição de Zoutendijk (9).

Convergência de métodos de busca linear

Usando os resultados vistos anteriormente, é possível mostrar a convergência global de toda uma classe de algoritmos.

Considere qualquer algoritmo para o qual

1. toda iteração produz decréscimo da função objetivo;
2. toda m-ésima iteração é uma iteração de máxima descida, com tamanho de passo escolhido de modo a satisfazer as condições de Wolfe.

Convergência de métodos de busca linear

Como $\cos\theta_k = 1$ para a direção de máxima descida, o resultado (11) vale.

É claro que nas demais $m-1$ iterações a idéia é fazer algo melhor do que uma iteração de máxima descida. Apesar desta iteração poder não fornecer um decréscimo grande, pelo menos ela garante a convergência global do método.