1 Elementos de Otimização

Neste documento estudaremos brevemente um problema geral de otimização em várias variáveis com destaque para os problemas de otimização sem restrições. Vamos supor que todas as funções envolvidas são contínuas e, pelo menos uma vez diferenciáveis.

1.1 Problema geral de otimização

Considere $f(x)$ uma função contínua de várias variáveis a valores reais. Matematicamente, escrevemos que $f : D \subset \mathbb{R}^n \to \mathbb{R}$. O problema geral de otimização envolvendo f é definido por

$$\begin{array}{rl} \text{minimizar} & f(x) \\ \text{restrita a} & x \in \Omega, \end{array}$$

onde $\Omega \subset D$ é um subconjunto do domínio de definição de f. A função f é chamada de função objetivo do problema. Se $\Omega = \mathbb{R}^n$, diremos que o problema de otimização é irrestrito, caso contrário, o problema será chamado de restrito.

Resolver um problema de otimização significa encontrar minimizadores. Existem dois tipos de minimizadores: os globais e os locais. $x^* \in \Omega$ é um minimizador global do problema se $f(x^*) \leq f(x)$ para todo $x \in \Omega$. No entanto, x^* será considerado minimizador local se $f(x^*) \leq f(x)$ para todo x em uma vizinhança de x^*. É evidente que todo minimizador global é local mas a recíproca não é verdadeira. A Figura 1 ilustra o gráfico de uma função contínua de uma variável que possui um minimizador local e um global denotados por x_{loc} e x_{glob}, respectivamente.

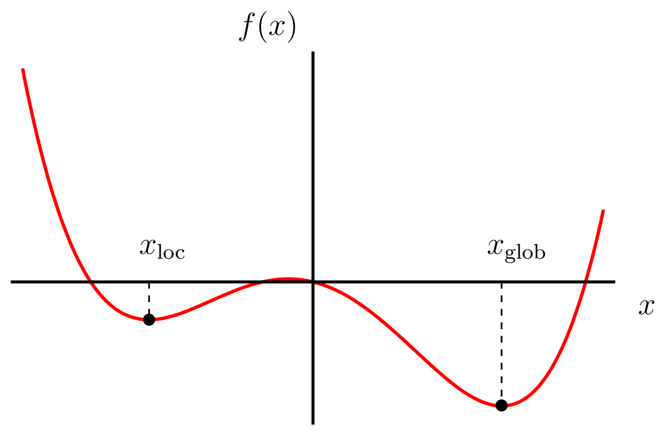

Figura 1: Uma função $f(x)$ com um minimizador local e um global.

Embora tenhamos definido o problema geral de otimização como sendo um problema de minimização, é possível convertê-lo em um problema de maximização pois, como f é contínua em D, minimizar $f(x)$ em Ω é equivalente a maximizar $-f(x)$ em Ω.

1.2 Otimização irrestrita

Nosso objeto de estudo neste documento é o problema de otimização sem restrições

$$\begin{array}{rl} \text{minimizar} & f(x) \\ \text{restrita a} & x \in \mathbb{R}^n, \end{array}$$

onde f é uma função diferenciável. Sabemos dos cursos de Cálculo que os candidatos à minimizadores desse problema satisfazem a condição $\nabla f(x) = 0$, onde $\nabla f(x)$ é o vetor gradiente de f avaliado em x. Como nas aplicações reais os problemas de otimização contêm muitas variáveis, faz-se necessário o desenvolvimento de algoritmos que buscam soluções aproximadas para estes problemas. No que segue, denotaremos o produto escalar (interno) entre dois vetores u e v por $u^T v$.

Um algoritmo geral de resolução para o problema de otimização irrestrita associado a uma função objetivo diferenciável $f(x)$ é baseado em uma direção de descida. Dizemos que um vetor $d \in \mathbb{R}^n$ é uma direção de descida a partir de $x \in \mathbb{R}^n$ se $\nabla f(x)^T d < 0$, ou seja, o ângulo que o vetor gradiente faz com o vetor d é obtuso ($> 90°$). Assim uma iteração de um algoritmo desse tipo consiste em, partindo de x, tomar d de descida e andar ao longo de d até que o valor da função f diminua. A Figura 2 ilustra algumas curvas de nível de uma função de duas variáveis $f(x)$. No ponto x o vetor gradiente $\nabla f(x)$ é perpendicular à curva de nível que passa por x. O vetor d que sai de x é uma direção de descida pois faz ângulo obtuso com $\nabla f(x)$. Nesta figura o minimizador global de $f(x)$ está representado por x^*.

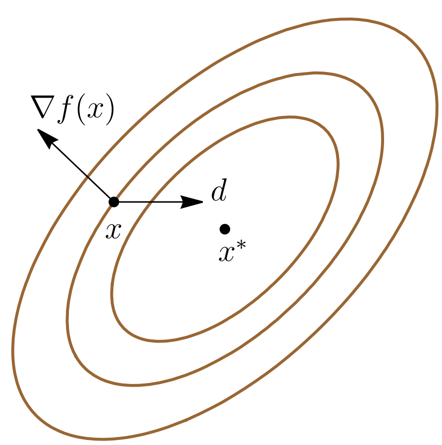

Figura 2: Exemplo de direção de descida a partir de x.

Os principais passos de um algoritmo geral baseado em uma direção de descida são:

0: Dados f diferenciável e $x_0 \in \mathbb{R}^n$, faça $k = 0$.

1: Enquanto $\nabla f(x_k) \neq 0$:

2: calcule a direção de descida d_k,

3: determine $\alpha_k > 0$ tal que $f(x_k + \alpha_k d_k) < f(x_k)$,

4: faça $x_{k+1} = x_k + \alpha_k d_k$, $k = k + 1$ e volte ao passo 1.

5: Fim

Os passos 2, 3 e 4 do algoritmo devem ser repetidos até que se obtenha x_k tal que $\nabla f(x_k) = 0$. O cálculo de α_k no passo 3 é chamado de Busca Linear. Em alguns casos, é possível realizar a Busca Linear Exata, ou seja, encontrar α_k que minimiza $f(x)$ ao longo de d_k partindo de x_k. A forma como a direção d_k é calculada em cada iteração define o tipo de algoritmo utilizado. Veremos a seguir duas escolhas particulares para d_k que dão origem ao Método do Gradiente e ao Método de Newton.

1.3 Método do Gradiente

O método mais comum para otimização irrestrita e também muito utilizado em problemas de inteligência artificial é o Método do Gradiente, também conhecido como Método da Máxima Descida. No contexto do algoritmo geral que discutimos anteriormente, basta tomar em cada iteração $d_k = -\nabla f(x_k)$. Sabemos dos cursos de Cálculo que ao longo dessa direção d_k a função $f(x)$ diminui mais rapidamente, partindo de x_k.

Se a função a ser minimizada for uma função quadrática com matriz hessiana definida positiva, podemos escrever f como

$$f(x) = \frac{1}{2}x^T Ax - b^T x + c.$$

Neste caso temos que $\nabla f(x) = Ax - b$ e $\nabla^2 f(x) = A$ é uma matriz simétrica é definida positiva, ou seja, seus autovalores são todos números reais positivos. Para esta função f é possível mostrar que o valor de α_k na busca linear exata é dado por

$$\alpha_k = \frac{d_k^T d_k}{d_k^T A d_k}.$$

A Figura 1.3 ilustra a evolução do Método do Gradiente na tentativa de encontrar o minimizador global x^* de uma função quadrática $f(x)$ de duas variáveis com matriz hessiana definida positiva. Nesta figura também podemos observar algumas curvas de nível de $f(x)$. Partindo de x_0, se unirmos os iterandos x_k por segmentos de reta, observaremos um caminho no formato de zigue-zague que se aproxima de x^*. É possível mostrar, no caso da busca linear exata, que

$$\nabla f(x_k)^T \nabla f(x_{k+1}) = 0,$$

isto é, os vetores gradientes avaliados em dois iterandos consecutivos são ortogonais. Isso explica o comportameto ilustrado na figura.

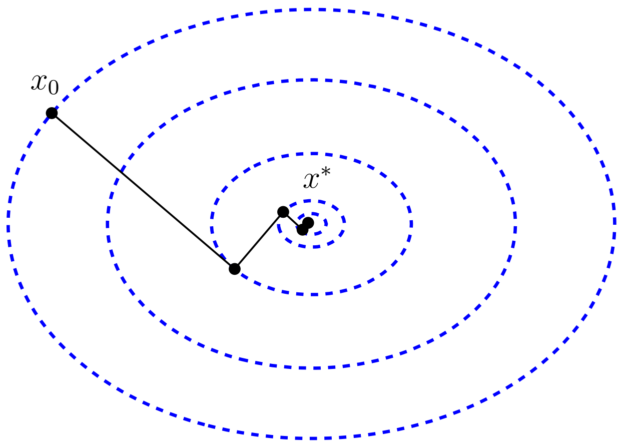

Figura 3: Método do Gradiente aplicado em uma função quadrática.

No caso em que a função objetivo f não é quadrática, não existe fórmula fechada para o cálculo de α_k e o sucesso do Método do Gradiente dependerá da estratégia a ser adotada para a busca linear e de propriedades específicas de $f(x)$.

- **Exemplo 1:** Considere a função $f(x) = 3x_1^2 - 4x_1x_2 + 4x_2^2 + 2x_1 - 3x_2$.

  a) Calcule o vetor gradiente e a matriz hessiana de $f(x)$.

  ◇ **Solução:** Como a função $f(x)$ tem duas variáveis, o vetor gradiente terá tamanho 2×1 e a matriz hessiana terá tamanho 2×2. Assim,

$$\nabla f(x) = \begin{pmatrix} 6x_1 - 4x_2 + 2 \\ -4x_1 + 8x_2 - 3 \end{pmatrix} \quad \text{e} \quad \nabla^2 f(x) = \begin{pmatrix} 6 & -4 \\ -4 & 8 \end{pmatrix}.$$

  b) Determine todos os pontos estacionários de $f(x)$.

  ◇ **Solução:** Chamamos de pontos estacionários aqueles que anulam o vetor gradiente. Logo, para calculá-los basta resolver a equação $\nabla f(x) = 0$. Igualando cada componente de $\nabla f(x)$ a zero, obtemos o seguinte sistema de equações

$$\begin{cases} 6x_1 - 4x_2 = -2, \\ -4x_1 + 8x_2 = 3, \end{cases}$$

cuja solução é $x_1 = -0.125$ e $x_2 = 0.3125$. Como a função $f(x)$ é quadrática e $\nabla^2 f(x)$ é uma matriz simétrica e definida positiva, o único ponto estacionário encontrado é o minimizador global de $f(x)$. Portanto, em $x^* = (-0.125, 0.3125)^T$ a função $f(x)$ atinge o menor valor possível: $f(x^*) = -0.59375$.

c) Partindo de $x_0 = (-1, 0.5)^T$, faça uma iteração do método do gradiente com busca linear exata.

◇ **Solução:** Temos que $d_0 = -\nabla f(x_0) = (6, -5)^T$. O passo α_0 da busca linear exata é dado por

$$\alpha_0 = \frac{d_0^T d_0}{d_0^T A d_0} = 0.0929.$$

Portanto,

$$x_1 = x_0 + \alpha_0 d_0 = \begin{pmatrix} -1 \\ 0.5 \end{pmatrix} + 0.0929 \begin{pmatrix} 6 \\ -5 \end{pmatrix} = \begin{pmatrix} -0.4426 \\ 0.0355 \end{pmatrix}.$$

Para efeito de comparação, observe que $f(x_0) = 2.5$ e $f(x_1) = -0.3361$, ou seja, em uma iteração o valor de $f(x)$ foi diminuído.

1.4 Método de Newton

Considere uma função f duas vezes diferenciável e o problema de encontrar um minimizador para f. Seja x_k no domínio de f tal que $\nabla f(x_k) \neq 0$. Podemos utilizar as derivadas de primeira e segunda ordens de f (contidas no vetor gradiente e na matriz hessiana, respectivamente) para construir uma aproximação quadrática de $f(x)$ na vizinhança de x_k. Desta forma

$$f(x) \approx q(x) = f(x_k) + \nabla f(x_k)^T (x - x_k) + \frac{1}{2}(x - x_k)^T \nabla^2 f(x_k)(x - x_k),$$

para x em uma vizinhança de x_k. Esta aproximação é a expansão de Taylor de segunda ordem de $f(x)$ em torno de x_k.

O método de Newton substitui, em cada iteração, o problema de minimizar $f(x)$ pelo problema de minimizar $q(x)$. Como $q(x)$ é uma função quadrática, o candidato a ponto de mínimo satisfaz $\nabla q(x) = 0$. Fazendo $d = x - x_k$ na expressão de $q(x)$, podemos escrever

$$\nabla q(x) = 0 \Rightarrow \nabla^2 f(x_k) d = -\nabla f(x_k),$$

que é um sistema de equações lineares. Uma vez que d foi encontrado resolvendo o sistema, o novo vetor x_{k+1} será dado por $x_{k+1} = x_k + \alpha_k d_k$, onde α_k é o tamanho de passo obtido por realizar uma busca linear ao longo de d_k para diminuir o valor de f. É importante observar que a matriz de coeficientes do sistema linear pode não admitir inversa e a resolução desse sistema pode não estar bem definida.

O algoritmo do Método de Newton pode ser resumido nos seguintes passos:

0: Dados $f(x)$ duas vezes diferenciável e $x_0 \in \mathbb{R}^n$, faça $k = 0$.

1: Enquanto $\nabla f(x_k) \neq 0$:

2: determine d_k tal que $\nabla^2 f(x_k) d_k = -\nabla f(x_k)$

3: encontre $\alpha_k > 0$ tal que $f(x_k + \alpha_k d_k) < f(x_k)$

4: faça $x_{k+1} = x_k + \alpha_k d_k$, $k = k + 1$ e volte ao passo 1.

5: Fim.

Observações:

a) Se a matriz hessiana $\nabla^2 f(x_k)$ for definida positiva, ela será invertível e a solução exata do sistema linear no passo 2 é dada por

$$d_k = [\nabla^2 f(x_k)]^{-1} \nabla f(x_k).$$

Além disso, é possível mostrar que nessas condições d_k é uma direção de descida. Na prática deve-se evitar inverter a matriz hessiana para o cálculo de d_k, substituindo este cálculo pelo uso de um método numérico para resolução de sistemas lineares.

b) Se $f(x)$ for uma função quadrática com hessiana definida positiva, o algoritmo encontra o minimizador de $f(x)$ em uma iteração, qualquer que seja x_0 escolhido.

c) Uma forma de evitar que na busca linear o passo dado produza pouco decréscimo no valor de $f(x)$ é pedir que α_k satisfaça a desigualdade

$$f(x_k + \alpha_k d_k) < f(x_k) + \lambda \nabla f(x_k)^T \alpha_k d_k,$$

com $\lambda \in (0, 1)$ fixo. Esta desigualdade é chamada na literatura de condição de Armijo.

É sabido que o método de Newton possui convergência local, isto é, se x^* é um minimizador local de $f(x)$ e $\nabla^2 f(x^*)$ é definida positiva, se escolhermos x_0 suficientemente próximo de x^*, a sequência de vetores gerada pelo algoritmo do método convergirá para x^* com taxa de convergência quadrática (considerada rápida em comparação a outros métodos para minimização de funções).

- **Exemplo 2:** Seja $f(x) = (1 - x_1)^2 + 100(-x_1^2 - x_2)^2 + 1$. Partindo de $x_0 = (-1.2, 1)^T$, faça uma iteração do método de Newton considerando o tamanho de passo $\alpha_0 = 1$.

◇ **Solução:** Primeiramente necessitamos do vetor gradiente e da matriz hessiana de f avaliados em x_0. Fazendo isso, obtemos:

$$\nabla f(x_0) = \begin{pmatrix} -1175.6 \\ 488.0 \end{pmatrix} \quad \text{e} \quad \nabla^2 f(x_0) = \begin{pmatrix} 2130 & -480 \\ -480 & 200 \end{pmatrix}.$$

Em seguida, devemos resolver o sistema linear $\nabla^2 f(x_0)d_0 = -\nabla f(x_0)$ para obter a direção de descida $d_0 = (0.0044, -2.4292)^T$. Como $\alpha_0 = 1$, o resultado da primeira iteração do método é

$$x_1 = x_0 + d_0 = \begin{pmatrix} -1.1955 \\ -1.4292 \end{pmatrix}.$$

Os valores de $f(x)$ em x_0 e x_1 são, respectivamente, $f(x_0) = 601.2$ e $f(x_1) = 5.82022$.

Quando utilizamos $\alpha_k = 1$ em uma iteração do método, dizemos que o passo dado foi "puro". Os *softwares* que implementam o método de Newton costumam empregar uma estratégia para controlar o tamanho de α_k. A Figura abaixo ilustra a evolução do método aplicado à função do Exemplo 2, partindo de $x_0 = (-1.2, 1)^T$. O minimizador global de f é $x^* = (1, -1)^T$ e também está indicado na figura juntamente com algumas curvas de nível de f. O cálculo de x_1 foi feito com um passo puro ($\alpha_0 = 1$) como vimos no Exemplo 2, mas no cálculo dos demais iterandos x_k o tamanho de α_k foi controlado ao longo das iterações. É possível observar que a sequência de vetores está convergindo para x^*.

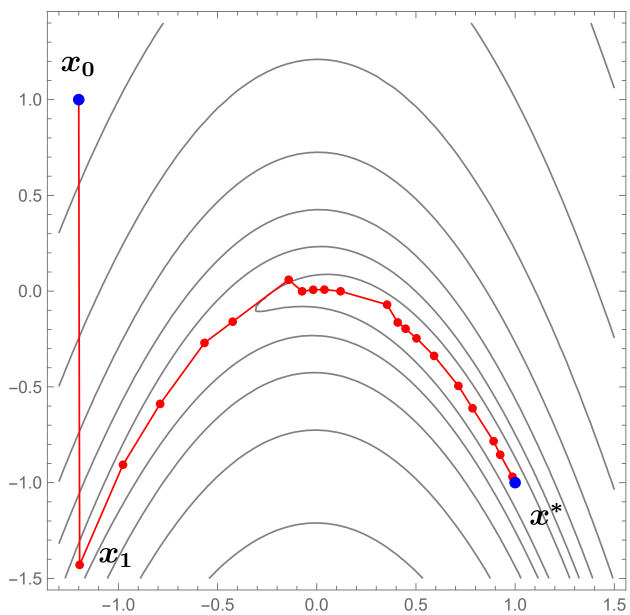

Figura 4: Método de Newton com controle de passo.

Bibliografia

1. Elementos de Programação Não Linear - A. Friedlander - Editora da Unicamp, 1994.

2. Numerical Optimization - J. Nocedal, S. J. Wright - Springer, 2006.

3. Linear and Nonlinear Programming - D. Luenberger, Y. Ye - Springer, 2010.