IF824 - Otimização
Algoritmo do gradiente

Gurvan Huiban

28 de abril de 2014

Os trabalhos são realizados na ferramenta Matlab[1] ou na ferramenta Sage[2].

1 Descrição do método

Seja o problema de otimização irrestrito seguinte:

$$\begin{array}{ll} \min & f(X) \\ & |X \in \Omega \end{array} \tag{1}$$

com

$$\Omega = \mathbb{R}^n \text{ e } f : \mathbb{R}^n \to \mathbb{R} \text{ diferenciável}$$

2 Direções de busca

Dado um ponto $X_0 \in \mathbb{R}^n$, um algoritmo de descida vai convergir para um mínimo local da bacia de atração contendo X_0, caso houver. Um algoritmo padrão baseado na estratégia de direções de busca é dado pelo algoritmo 1.

Data: Função f, ponto inicial X_0
Result: Aproximação numérica de um mínimo local de f
$k \leftarrow 0$;
repeat
  Calcular d_k;
  Resolver o problema de otimização unidimensional: $\alpha_k = \arg\min_{\alpha \geqslant 0} f\left(X_k + \alpha_k d_k\right)$;
  $X_{k+1} \leftarrow X_k + \alpha_k d_k$;
  $k \leftarrow k + 1$;
until *Condição de parada*;

Algorithm 1: Algoritmo de descida com direções de busca padrão

2.1 Algoritmo do gradiente

O algoritmo do gradiente é um dos mais simples algoritmos baseados em direções de busca. Neste algoritmo, temos:

- $d_k \leftarrow -\nabla f(X_k)$

onde $\nabla f(X_k)$ representa o vetor gradiente de f em X_k.

[1] ©1994-2013 The MathWorks, Inc.
[2] http://www.sagemath.org/

2.2 Seção áurea

A otimização unidimensional pode ser feita com o método da seção áurea.

Dada uma função $g : \mathbb{R}^+ \to \mathbb{R}$ contínua e dado um intervalo $[c_1, c_2]$ contendo um mínimo local de g, o algoritmo da seção áurea (algoritmo 2) calcula um mínimo local de g sobre $[c_1, c_2]$ com precisão ϵ.

Data: Função g, intervalo $[c_1, c_2]$, precisão ϵ
Result: Mínimo local de g sobre $[c_1, c_2]$
$x_1 \leftarrow c_2 - \alpha\,(c_2 - c_1)$;
$x_2 \leftarrow c_1 + \alpha\,(c_2 - c_1)$;
while $\frac{(c_2 - c_1)}{2} > \epsilon$ **do**
  if $g(x_1) \leqslant g(x_2)$ **then**
    $c_2 \leftarrow x_2$;
    $x_2 \leftarrow x_1$;
    $x_1 \leftarrow c_2 - \alpha\,(c_2 - c_1)$;
  else
    $c_1 \leftarrow x_1$;
    $x_1 \leftarrow x_2$;
    $x_1 \leftarrow c_1 + \alpha\,(c_2 - c_1)$;
  end
end

Algorithm 2: Otimização unidimensional com seção áurea

2.3 Busca de intervalo

Para usar o método da seção áurea, precisamos de um intervalo contendo um mínimo local. A busca deste intervalo pode ser feita com o método seguinte. Dada uma função $g : \mathbb{R}^+ \to \mathbb{R}$ contínua e decrescente em $x = 0$, o algoritmo 3 busca um intervalo $[c_1, c_2]$ contendo um mínimo local de g.

Data: Função g, passo inicial δ
Result: Intervalo $[c_1, c_2]$
$c_1 \leftarrow 0$;
if $g(c_1 + \delta) \leqslant g(c_1)$ **then**
  $x_m \leftarrow c_1 + \delta$;
  $\delta \leftarrow 2\delta$;
  while $g(x_m) > g(x_m + \delta)$ **do**
    $c_1 \leftarrow x_m$;
    $x_m \leftarrow x_m + \delta$;
    $\delta \leftarrow 2\delta$
  end
  $c_2 \leftarrow x_m + \delta$;
else
  $c_2 \leftarrow c_1 + \delta$;
  $\delta \leftarrow \frac{\delta}{2}$;
  while $g(c_1 + \delta) > g(c_1)$ **do**
    $c_2 \leftarrow c_1 + \delta$;
    $\delta \leftarrow \frac{\delta}{2}$;
  end
  $x_m \leftarrow c_1 + \delta$;
end

Algorithm 3: Busca de intervalo incluindo um mínimo local

2.4 Critério de parada

Existem vários critérios de parada. Nós vamos considerar que quando a maior variação da função f nas três últimas iterações ficar menor que um ϵ de precisão, podemos parar o algoritmo do gradiente. Nós queremos também parar caso o número de iterações passe dos 100.

3 Implementação

O objetivo do trabalho é a implementação do algoritmo do gradiente com Matlab ou Sage (sua preferência).

- Implemente o critério de parada, que recebe os três valores da função f, o ϵ de precisão, e a quantidade de iterações já calculadas.
- Implemente o algoritmo da seção áurea, que recebe a função f, o ponto atual X, a direção de busca d, o ϵ de precisão e o intervalo $[c_1, c_2]$ onde procurar um mínimo local para a função $g(\lambda) = f(X + \lambda d)$, e que retorna o mínimo local.
- Implemente o algoritmo de busca de intervalo, que recebe a função f, o ponto atual X, a direção de busca d e um passo inicial δ, e que retorna um intervalo $[c_1, c_2]$ tal que $g(\lambda) = f(X + \lambda d)$ contenha um mínimo local.
- Implemente o algoritmo do gradiente, que recebe uma função f, um ponto inicial X_0 e um ϵ de precisão, e que retorna a solução encontrada.

Para o cálculo do gradiente, use as funções da ferramenta escolhida.

3.1 Problemas sugeridos

Seguem abaixo alguns problemas que podem ser usados para testar a implementação do algoritmo. Se possível, representar graficamente a função e a solução encontrada pelo algoritmo.

- $f : (x_1, x_2) \mapsto x_1^2 + x_2^2 - 8x_1 + 5x_2 + 22.25$
- $f : (x_1, x_2) \mapsto 10x_1^2 + x_2^2 - 8x_1 + 5x_2 + 22.25$
- $f : (x_1, x_2) \mapsto 9x_1^2 - 3x_1x_2 + 1.25x_2^2 - 24x_1 + 9x_2 + 22.25$
- $f : (x_1, x_2) \mapsto 0.5x_1^2 + 0.25x_2^2 + 20\sin(0.1x_1x_2) + 0.25$