

Metaheurísticas

Hill Climbing

Felipe Augusto Lima Reis

felipe.reis@ifmg.edu.br



**INSTITUTO
FEDERAL**
Minas Gerais

Sumário



1 Otimiz. Gradiente

2 Hill Climbing

OTIMIZAÇÃO BASEADO EM GRADIENTE

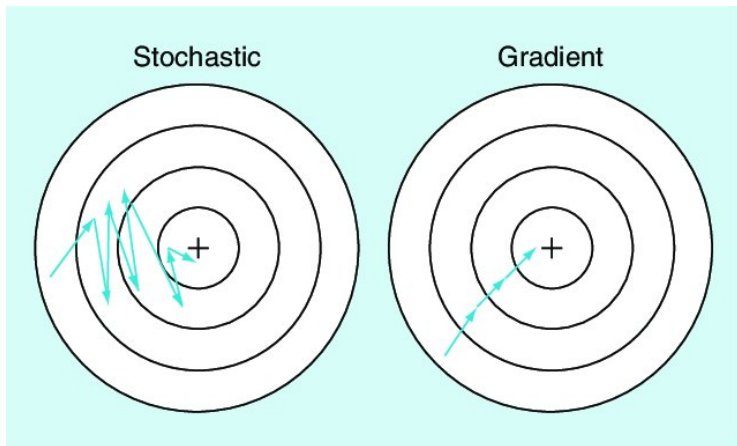
Método do Gradiente

- O método do gradiente é um método numérico usado para maximização ou minimização de uma função objetivo [Luke, 2013]
 - Método do máximo declive (*gradient descent*) busca minimizar o erro esperado;
 - Método do *gradient ascent* busca encontrar o valor máximo de uma função;
- O método busca seguir o caminho mais íngreme na superfície que representa a função objetivo [Coppin, 2004] [Luke, 2013].

Método do Gradiente Estocástico (SGD)

- Variação do método do gradiente que realiza uma aproximação estocástica da otimização;
- Substitui o gradiente real por uma estimativa do mesmo, calculada a partir de um subconjunto de dados selecionado aleatoriamente;
- Reduz o número de iterações em problemas de otimização, alcançando iterações mais rápidas, em troca de uma menor taxa de convergência.

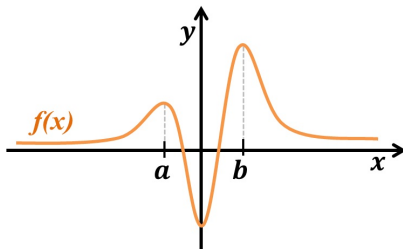
Gradiente x Gradiente Estocástico



Fonte: [Carpenter et al., 2018]

Pontos de Máximo e Mínimo

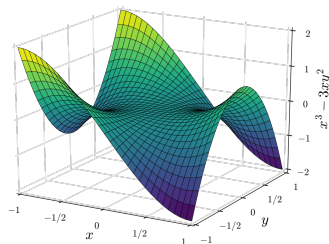
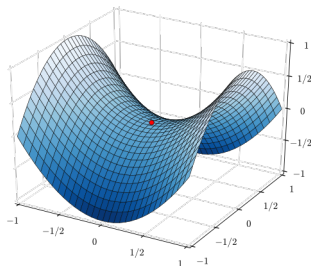
- Pontos de máximos e mínimos são os pontos de picos e de depressões da função;
- Podemos ter máximos/mínimos locais e absolutos (globais);
- Esses pontos, em funções otimização são conhecidos como **ótimos locais** e **ótimos globais**.



Fonte: [Dicas de Cálculo, 2020]

Pontos de Sela

- Um ponto de sela (*saddle point*) é o ponto sobre uma superfície no qual a declividade é nula;
- É o ponto sobre na qual a elevação é máxima em uma direção e mínima em outra;
- Não corresponde ao ponto de mínimo tampouco de máximo.



Fonte: [Wikipedia contributors, 2020]

HILL CLIMBING

Hill Climbing¹

- Hill Climbing é um tipo de busca local informada, relacionada ao método do gradiente (*ascendente*) [Coppin, 2004] [Luke, 2013];
- Tipo de método de otimização local, uma vez que tenta otimizar um conjunto de valores e frequentemente encontra o máximo local, ao invés do máximo global [Coppin, 2004];
- Não armazena o caminho percorrido até a solução corrente e sim a solução propriamente dita como estado [Luzia and Rodrigues, 2009].

¹ Tradução literal: escalada de montanha

Hill Climbing



- Apesar de relacionado ao método do gradiente, o Hill Climbing não precisa de conhecer o gradiente;
- As soluções candidatas são testadas na região do candidato atual;
- Caso uma solução candidata seja melhor que a solução atual, ela é adotada;
- O algoritmo “escala a montanha” até atingir o ótimo local [Luzia and Rodrigues, 2009] [Luke, 2013].

Hill Climbing



- O algoritmo inicia com uma solução randômica, potencialmente ruim;
- Iterativamente, efetua pequenas modificações na solução atual, em busca de melhorar o resultado da função objetivo;
- O algoritmo termina quando não encontra nenhuma melhoria possível em uma iteração;
- Idealmente a solução obtida é ótima, porém não há garantia de otimalidade [Luzia and Rodrigues, 2009].

Hill Climbing

- O pseudo-algoritmo do Hill Climbing pode ser visto abaixo;
- É possível observar que o algoritmo é extremamente simples e pode ser implementado rapidamente;
- O método “*NovaSolução*” (linha 3), pode ser implementado como um simples transformação estocástica (aleatória) a partir do valor anterior [Luzia and Rodrigues, 2009].

```
1: S ← solução inicial
2: repita
3:   R ← NovaSolução(S)
4:   se (Qualidade(R) > Qualidade(S)) então
5:     S ← R
6: até que S seja a solução ideal ou o tempo tenha se esgotado
7: devolva S
```

Hill Climbing, versão básica

Fonte: [Luzia and Rodrigues, 2009]

Steepest Ascent Hill Climbing

- Nesta melhoria são avaliados os sucessores da solução atual, para escolha daquela que oferecer o melhor resultado [Luzia and Rodrigues, 2009] [Luke, 2013];
- O algoritmo pode tomar n direções aleatórias e escolher aquela cujo crescimento (ou decrescimento) é maior.

```

01:  $n \leftarrow$  número de extensões a serem geradas
02:  $S \leftarrow$  solução candidata inicial qualquer
03: repita
04:    $R \leftarrow$  NovaSolução(  $S$  )
05:   repita  $n - 1$  vezes
06:      $W \leftarrow$  NovaSolução(  $S$  )
07:     se Qualidade( $W$ ) > Qualidade( $R$ ) então
08:        $R \leftarrow W$ 
09:   se Qualidade( $R$ ) > Qualidade( $S$ ) então
10:      $S \leftarrow R$ 
11: até que  $S$  seja a solução ideal ou o tempo tenha se esgotado
12: devolva  $S$ 
    
```

Steepest Ascent Hill Climbing

Fonte: [Luzia and Rodrigues, 2009]

Steepest Ascent Hill Climbing with Replacement

- Nesta versão, não é avaliado se a solução seguinte é melhor que a atual;
- A solução atual é simplesmente substituída pela seguinte, sem comparação;
- Essa variação tem como objetivo aumentar a área avaliada pelo algoritmo;
- Como ponto negativo, pode não retornar a melhor solução, uma vez que ela pode ser perdida ao longo do tempo [Luke, 2013].

Hill Climbing with Random Restart

- Esta melhoria busca fazer com que o método cubra uma maior área no espaço de busca;
- O algoritmo executa o Hill Climbing por um certo tempo e então efetua uma busca aleatória, de modo a fugir de máximos locais [Luzia and Rodrigues, 2009] [Luke, 2013].

```

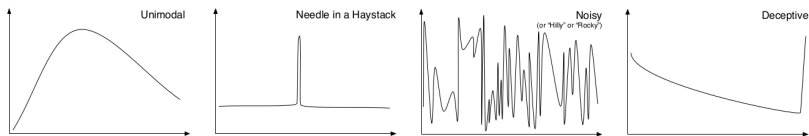
01: S ← solução candidata inicial qualquer
02: Melhor ← S
03: repita
04:   período ← NovoTempo()
05:   repita
06:     R ← NovaSolução( S )
07:     se Qualidade(R) > Qualidade(S) então
08:       S ← R
09:   até que S seja ideal, ou que período ou o tempo total tenham se esgotado
10:   se Qualidade(S) > Qualidade(Melhor) então
11:     Melhor ← S
12:   S ← solução candidata aleatória
13: até que S seja a solução ideal ou o tempo tenha se esgotado
14: devolva Melhor
    
```

Hill Climbing with Random Restart

Fonte: [Luzia and Rodrigues, 2009]

Hill Climbing

- O Hill Climbing e suas variações podem possuir dificuldade de convergência para máximos globais em algumas funções
 - Dependendo da aplicação, uma análise do cenário e da função a ser solucionada pode auxiliar na escolha do algoritmo;
 - Melhorias do Hill Climbing, em geral, tem desempenho superior ao algoritmo original.



Fonte: [Luke, 2013]

Vantagens e Desvantagens

- Vantagens

- Fácil de ser implementado;
- Pode ser utilizado como base para construção de métodos mais sofisticados [Luzia and Rodrigues, 2009];
- Devido à sua simplicidade, pode ser utilizado em problemas que possuem limitação computacional ou de tempo de processamento;

- Desvantagens

- Possibilidade de ficar preso em máximos locais;
- Baixa efetividade em “regiões planas”;
- Efetividade do método é dependente da função em que está sendo aplicado [Luzia and Rodrigues, 2009].

Referências I



Carpenter, K. A., Cohen, D. S., Jarrell, J. T., and Huang, X. (2018).
Deep learning and virtual drug screening.
Future Medicinal Chemistry, 10(21):2557–2567.
PMID: 30288997.



Coppin, B. (2004).
Artificial intelligence illuminated.
Jones and Bartlett illuminated series. Jones and Bartlett Publishers, 1 edition.



Dicas de Cálculo (2020).
Máximos e mínimos de uma função: teste da primeira derivada.
[Online]; acessado em 22 de Setembro de 2020. Disponível em: <https://www.dicasdecaculo.com.br/conteudos/derivadas/aplicacoes-de-derivadas/maximo-minimo-funcao/>.



Luke, S. (2013).
Essentials of Metaheuristics (Second Edition).
lulu.com, 2 edition.
[Online]; Disponível em: <https://cs.gmu.edu/~sean/book/metaheuristics/>.



Luzia, L. F. and Rodrigues, M. C. (2009).
Estudo sobre as metaheurísticas.
[Online]; acessado em 22 de Setembro de 2020. Disponível em:
<https://www.ime.usp.br/~gold/cursos/2009/mac5758/LeandroMauricioHeuristica.pdf>.

Referências II



Wikipedia contributors (2020).

Saddle point.

[Online]; acessado em 09 de Setembro de 2020. Disponível em:
https://en.wikipedia.org/wiki/Saddle_point.