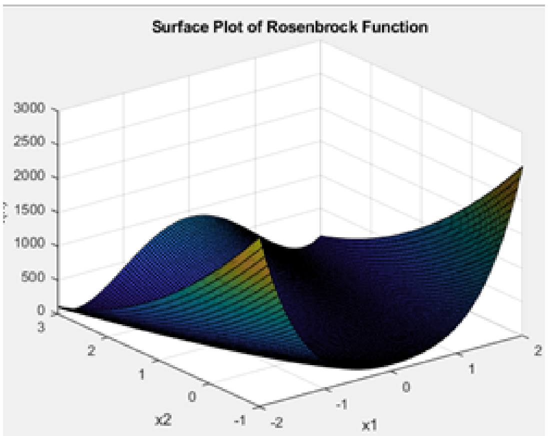
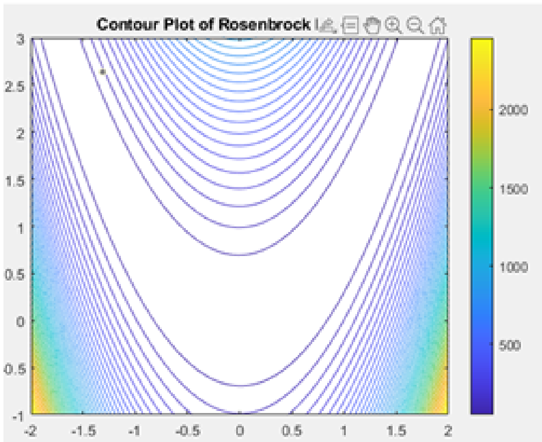
Nonlinear Optimization

Performance Analysis of the Quasi-Newton Method with BFGS Update and Backtracking Line Search





David Matos Furtado

2024

1 Introdução

A otimização é uma área fundamental em diversas disciplinas científicas e engenharias, desempenhando um papel crucial na resolução de problemas complexos de tomada de decisão. Um dos desafios enfrentados pelos pesquisadores e profissionais é a busca por métodos eficazes e eficientes para encontrar soluções ótimas ou aproximadamente ótimas para uma ampla gama de problemas de otimização.

Neste contexto, o método Quasi-Newton com atualização BFGS tem se destacado como uma abordagem amplamente utilizada para a resolução de problemas de otimização sem restrições. Este método pertence à classe dos métodos de gradiente, que iterativamente atualizam uma estimativa da matriz Hessiana para guiar a busca pela solução ótima. Além disso, a incorporação de técnicas de pesquisa em linha, como backtracking e a condição de Armijo, aumenta a eficiência e a robustez do método, tornando-o uma escolha atraente para uma variedade de aplicações práticas.

O presente trabalho tem como objetivo investigar a eficácia e o desempenho do método Quasi-Newton com atualização BFGS e pesquisa em linha na resolução de problemas de otimização. Especificamente, examinaremos como o método se comporta em face a problemas de otimização conhecidos e compararemos seu desempenho com outros métodos amplamente utilizados.

2 Implementação

2.1 Método Quasi-Newton com atualização BFGS

O método Quasi-Newton é um método que busca resolver problemas de otimização não lineares sem a necessidade de calcular matriz Hessiana que em muitos casos é muito dispendioso computacionalmente ou é difícil de ser realizado, tornando-se assim um método mais rápido e eficiente a ser utilizado. Para substituir o cálculo da matriz este método opta por calcular uma aproximação da mesma com base nas diferenças entre os gradientes e dos pontos sucessivos.

A matriz Hessiana aproximada, denotada por B_k, é atualizada em cada iteração do algoritmo BFGS com o objetivo de fornecer uma estimativa precisa da Hessiana da função. A atualização de B_k é realizada utilizando informações sobre as mudanças nos gradientes da função entre iterações sucessivas e em termos matemáticos, a atualização de B_k segue a seguinte fórmula:

$$B_{k+1} = B_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} \tag{1}$$

Essa atualização garante que B_{k+1} seja uma aproximação positiva definida da Hessiana da função.

2.2 Critérios de Paragem

Os critérios de paragem são utilizados para determinar quando encerrar as iterações de um algoritmo de otimização e são fundamentais para determinar a paragem do algoritmo ao chegar a uma solução suficientemente boa, sem desperdiçar recursos computacionais em iterações desnecessárias.

O algoritmo implementado usa o número máximo de iterações e a convergência do algoritmo como critérios. A convergência é geralmente determinada quando a norma do gradiente da função se torna menor do que um valor pré-especificado, ε, e o limite máximo de iterações, $K_{\max}$, é definido para evitar a execução indefinida do algoritmo.

2.3 Pesquisa em Linha com Backtracking

A pesquisa em linha com backtracking é uma estratégia usada para determinar o tamanho do *step* ao longo da direção de descida em algoritmos de otimização. A ideia básica é começar com um tamanho de passo inicialmente grande e, em seguida, reduzi-lo gradualmente até que uma condição adequada seja satisfeita. Isso é feito para garantir que o tamanho do passo escolhido resulte em uma redução suficiente na função, sem ser muito pequeno ou muito grande.

Na prática, é definido o passo inicial, α_0, e um parâmetro de redução, ρ. Para que depois se proceda a verificar a condição de Armijo que assegura que o próximo valor da função depois do *step* é menor ou igual ao valor da função mais uma parte segundo a direção da derivada. Matematicamente, esta condição pode ser expressa da seguinte maneira:

$$f(x_k + \alpha p_k) \leq f(x_k) + c_1 \alpha \nabla f(x_k)^T p_k \tag{2}$$

onde p_k é a direção de descida, c_1 um parâmetro pequeno positivo (geralmente 10^{-4}), e α é o tamanho do passo de linha.

3 Eficiência e Eficácia

Como critérios de eficiência e eficácia serão levados em conta o tempo de execução dos algoritmos e o número de iterações necessárias para chegar à solução. Como problema de otimização foi escolhida uma variante da função Rosenbrock que é amplamente utilizada para testar este tipo de algoritmos. A função é definida por:

$$f(x, y) = (a - x)^2 + b(y - x^2)^2 \tag{3}$$

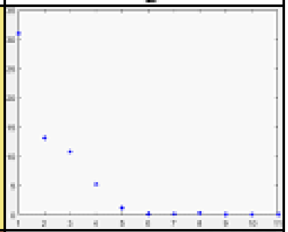
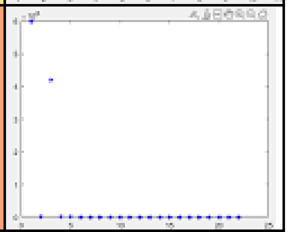
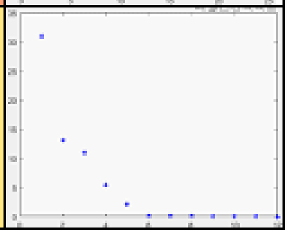
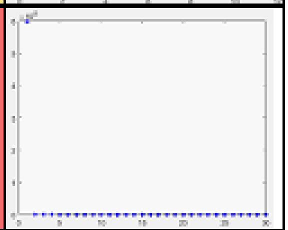
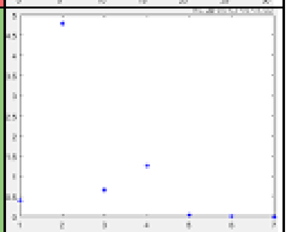
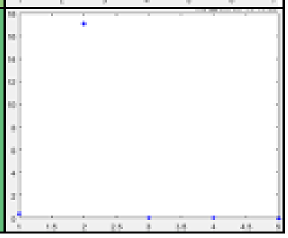
Método	Tempo de Execução (s)	Iterações	Imagem
Método Quasi-Newton com Atualização BFGS e Pesquisa em Linha com Backtracking	0.1831	11	
Método Quasi-Newton com Atualização BFGS	0.2262	22	
Método Quasi-Newton com rank 1 atualização e Pesquisa em Linha com Backtracking	0.1554	12	
Método Quasi-Newton com rank 1	0.1724	30	
Newton Method com Pesquisa em Linha com Backtracking	0.1111	7	
Newton Method	0.1236	5	

Figura 1: Resultado e comparação do desempenho dos vários métodos.

4 Conclusão

Após realizar uma série de testes com diferentes algoritmos, constatei que os resultados práticos geralmente se alinham com as expectativas teóricas. Notavelmente, observou-se que, em geral, o método de Newton se destacou como o mais rápido e exigiu o menor número de iterações para alcançar a solução. Vale constatar que isto serve para este caso específico e deve-se, em parte, à escolha favorável do ponto inicial x_0 e pelo cálculo da matriz Hessiana ser possível, estas condições contribuíram significativamente para a eficiência do método. Em situações onde esses elementos não são tão favoráveis, a eficácia do método de Newton pode ser comprometida, tornando a determinação da solução muito mais desafiadora.

Além disso, foi observado que os métodos, quando combinados com as técnicas de pesquisa em linha e backtracking, demonstraram resultados superiores tanto em termos de tempo de execução quanto em número de iterações, para todos os métodos testados. Os valores padrões escolhidos para os parâmetros de pesquisa em linha, c_1, α e ρ, mostraram-se eficazes para os propósitos deste estudo. No entanto, é importante destacar que ajustes adicionais nesses parâmetros podem levar a resultados ainda melhores, possibilitando um refinamento adicional do desempenho dos métodos de otimização utilizados.

Estes resultados ressaltam a importância de considerar não apenas a teoria por trás dos métodos de otimização, mas também a sua implementação prática e a influência dos parâmetros escolhidos. Ao ajustar e otimizar esses parâmetros, é possível maximizar a eficiência e a eficácia dos algoritmos de otimização, garantindo resultados mais rápidos e precisos em uma variedade de problemas de otimização.