Universidade Federal de São Carlos

Relatório: Algoritmos Metódo de Newtom e Método da Secante

Cálculo Numérico
Lucio Mitsuru Seki, 379883
Viviane Bonadia dos Santos, 379972
Professora: Silvia Carvalho

1. Introdução

O presente relatório foi escrito com o objetivo de apresentar a implementação de dois algoritmos que obtêm a raíz de uma função: o Método de Newton e o Método da Secante. Além da apresentação do programa, é feito aqui uma breve discussão dos métodos implementados. Ambos os métodos que serão apresentados e discutidos neste relatório são iterativos, ou seja, a cada iteração realizada os valores obtidos se aproximam cada vez mais dos resultados esperados.

2. Métodos Implementados

2.1. Método de Newton

O método de Newton é uma técnica para encontrar zeros de uma função diferenciável. Este método foi implementado seguindo a Equação 1, onde x_k é o valor obtido na k-ésima iteração. No caso da primeira iteração, o método necessita que um valor seja fornecido, ou seja, ao definir uma função para que o método encontre a raiz é necessário também escolher um ponto inicial (x_0).

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \tag{1}$$

Este método tem convergência quadrática, ou seja, a quantidade de dígitos significativos corretos duplica a cada iteração [Franco, 2007],contudo, a convergência depende fortemente do ponto inicial (x_0). O fato de existir o cálculo da derivada no método de Newton traz uma desvantagem à esta técnica, uma vez que o cálculo de derivada é caro computacionalmente.

2.2. Método da Secante

O método da Secante deriva do método de Newton. Esta técnica elimina uma das desvantagens encontradas no método de Newton que era o cálculo da derivada. A Equação 2 apresenta os cálculos realizados a cada iteração para obter a raiz da função, onde x_k representa o valor obtido na k-ésima iteração e x_{k-1} o valor obtido na iteração anterior à esta. Percebe-se portanto que ao contrário do método apresentado anteriormente, aqui são necessários dois pontos fornecidos inicialmente para que a primeira iteração possa ser feita.

$$x_{k+1} = \frac{x_{k-1}f(x_k) - x_k f(x_{k-1})}{f(x_k) - f(x_{k-1})} \tag{2}$$

Em [Franco, 2007] é possível encontrar uma descrição detalhada dos métodos apresentados.

3. O Programa

As implementações dos dois métodos foram feitas utilizando o programa Matlab.

Para utilizar o programa, é preciso executar o arquivo em Matlab de nome $main$. O usuário deve informar a função que ele deseja encontrar a raiz, dois pontos iniciais, um valor para a tolerância e o número máximo de iterações.

A função de entrada admite apenas x como variável. Os pontos iniciais são utilizados para o cálculo da primeira iteração dos métodos. O método de Newton utiliza apenas um ponto inicial, neste programa portanto é utilizado para o método de Newton o primeiro valor informado pelo usuário, enquanto o método da Secante utiliza os dois pontos fornecidos pelo usuário na primeira iteração.

O valor de tolerância é comparado com o erro obtido a cada iteração. Neste trabalho foi utilizado o erro relativo, como mostra a Equação 3 onde e_{k+1} representa o erro relativo na iteração $k+1$ e x_{k+1} e x_K representam os valores obtidos nas iterações $k+1$ e k respectivamente.

$$e_{k+1} = \frac{|x_{k+1} - x_K|}{|x_{k+1}|} \tag{3}$$

O valor de tolerância é utilizado como critério de parada do algoritmo. Quando $|x_k - x_{k+1}|$ <tolerância, ou seja, quando o algoritmo converge, o algoritmo para. O mesmo valor é utilizado para comparar se $f(x)$ <tolerância, ou seja, se o x está bastante próximo da raíz. Outro critério de parada do algoritmo é o número máximo de iterações.

No final da execução, o programa exibe o relatório de execução de cada um dos métodos, como no exemplo abaixo:

```
Método de Newton para a função f(x) = x^2 - 7
ponto inicial: 2.000000
tolerância: 1.000000e-07
x obtido: 2.645751
número de iterações: 4
tempo de execução: 0.139881


Método da Secante para a função f(x) = x^2 - 7
pontos iniciais: [2.000000, 3.000000]
tolerância: 1.000000e-07
x obtido: 2.645751
número de iterações: 4
tempo de execução: 0.000682
```

O programa é composto ao todo por oito aquivos de implementação matlab.

Para o usuário entrar com uma função no programa é preciso, conforme citado anteriormente executar o arquivo de nome $main.m$.

Os arquivos $newton.m$ e $secante.m$ contém as implementações dos métodos de Newton e da Secante respectivamente.

Os arquivos $test\_funcao\_sem\_zero.m$, $test\_ruggiero\_newton.m$, $test\_ruggiero\_secante.m$ e $test\_tolerancia.m$ possuem casos de teste utilizados. O arquivo $exec\_testes.m$ ao ser executado roda o programa para os casos de testes definidos nos arquivos citados anteriormente.

4. Validação do Programa

Para verificar a corretude das implementações, estas foram testadas com o uso do *framework* de teste unitário xunit [Eddins, 2010], utilizando os exemplos descritos em [Ruggiero and da Rocha Lopes, 1996].

Os testes podem ser reproduzidos executando o *script* exec_testes.

Além dos testes de corretude, os métodos foram testados para se estudar o seu comportamento conforme os parâmetros passados. Alguns exemplos são citados abaixo para ilustrar o impacto de alguns parâmetros no comportamento dos métodos:

Seja a função:

$$f(x) = x^2 - 7 \tag{4}$$

Os testes abaixo foram testados com a tolerância de 10e-8

A Tabela 1 apresenta um resumo dos resultados obtidos para a função 4. Nesta tabela x_0 e x_1 representam os pontos inicais $k(Newton)$ e $k(Secante)$ o numero de iterações necessárias para o método convergir.

Tabela 1. Resumo dos resultados.

x_0	x_1	k(Newton)	k(Secante)
2	3	4	4
-1	2	5	7
3	10	3	6
1	2.645751	5	2

Seja a função:

$$f(x) = sin(x) \tag{5}$$

Com os pontos iniciais [-1.3, 5.2], o método de Newton convergiu em 5 iterações, mas o da Secante não convergiu em 30 iterações. Com os pontos iniciais[-1, 1.5], os métodos convergiram em 4 e 5 iterações respectivamente.

Na maioria dos testes, o método de Newton demorou mais de 100 vezes do que o método da Secante. Na maioria dos testes, o método de Newton convergiu em menos iterações do que o método da Secante.

O método da Secante converge em um menor número de iterações comparado a Newton somente quando o segundo ponto inicial é bastante próximo da raíz.

Na função seno, o método da Secante com os pontos iniciais [-1.3, 5.2] dificilemnte converge, pois a função oscila demais, e a reta da secante dificilmente acompanha a curva da função. Com os pontos iniciais [1, 1.5], no entanto, o método convergiu rapidamente.

5. Conclusão

Os testes mostram que o método de Newtoton demora, na maioria das vezes, mais de 100 vezes que o método da Secante para executar. Isto se deve ao fato do método de Newton utilizar a função derivada, que é computacionalmente cara.

O método de Newton costuma convergir em menos iterações, mas o método da Secante tem a vantagem de poder receber o segundo ponto inicial (que pode melhorar no desempenho do algoritmo). Dependendo do valor escolhido, pode diminuir drasticamente a quantidade de iterações. Ambos os métodos são bastante sensíveis à escolha dos pontos iniciais.

Ambos os métodos também são bastante sensíveis à função em que se aplicam. Na função seno, por exemplo, o método da secante é totalmente ineficiente quando a reta da secante atravessa as curvas do seno, enquanto o médo de Newton tende a acompanhar a curva.

Os testes citados foram repetidos com diferentes números de tolerância (10e-8, 10e-16, 10e-32), mas a quantidade de iterações não aumentou muito. Isto se deve porque os métodos possuem convergência alta.

O objetivo da implementação dos métodos foi comparar na prática a eficiência de um método em relação a do outro, além de possibilitar testes com uma robustez maior. Através do estudo dos métodos, implementação e experimentos realizados foi possível perceber o comportamento de cada um dos métodos nos diferentes testes utilizados.

O método de Newton, possui sobre o método da Secante a vantagem de convergência maior. Ainda assim, esta vantagem não compensa já que demora muito mais que o método da Secante por executar a custosa derivada.

A elaboração deste projeto permitiu um melhor entendimento sobre os métodos iterativos apresentados em aula, e também ajudou a compreender as suas caraaterísticas.

Referências

[Eddins, 2010] Eddins, S. (2010). Matlab xunit test framework. url: http://www.mathworks.com/matlabcentral/fileexchange/22846-matlab-xunit-test-framework.

[Franco, 2007] Franco, N. M. B. (2007). *Cálculo Numérico*.

[Ruggiero and da Rocha Lopes, 1996] Ruggiero, M. A. G. and da Rocha Lopes, V. L. (1996). *Cálculo Numérico: aspectos teóricos e computacionais*.