Cálculo Numérico Computacional	Exercícios **lista 04**
Raizes aproximadas	Varredura, método da tangente
Prof. Tarcisio Praciano-Pereira	**Dep. de Matemática**
tarcisio@member.ams.org	
aluno:	
Univ. Estadual Vale do Acaraú	Sobral, 11 de novembro de 2007
`Documento escrito com LaTeX`	`sis. op. Debian/Gnu/Linux`

Por favor, prenda esta folha de rosto na sua solução desta lista deixando-a em branco, caso você a responda em papel. Ela será usada para a correção.

Se você quiser entregar esta lista em papel, entregue-a na secretária do seu curso até a data estipulada observando o horário de abertura da secretaria do Curso de Matemática. Alternativamente entregue na página do curso, por favor observe a nomenclatura dos arquivos eletrônicos. Preferencialmente entregue em pdf.

data da entrega desta lista: 19 de Novembro de 2007 até 24:00 h, via eletrônica ou até 22 horas, na secretária do Curso de Matemática pelo método medieval.

1 Orientação

Objetivo: Aplicar "`varredura`" na solução de problemas que modelem um fenômeno ao longo de um intervalo. Calcular aproximadamente a raíz de f usando a equação da reta tangente. Você pode fazer (deve) os cálculos usando um programa ou uma calculadora, entretanto os gráficos, nesta lista de exercícios, se forem feitos à mão, devem ser feito em papel quadriculado (ou milimetrado). Todo os gráficos podem ser feitos com `gnuplot` porém tenha o cuidado de garantir as proporções como uma prova de que você está entendendo.

Vamos estudar uma metodologia que tem 200 anos, o objetivo em si não é a determinação de raízes, mas dominar uma técnica que tem aplicações diversas. Vou usar o capítulo 2 de ananu00_2p.pdf que se encontra aqui:

`http://www.calculo-numerico.sobralmatematica.org`,

procure o link "textos".

As questões estão redigidas na forma de um tutorial, se dividem em itens de tal modo que devem auxiliá-l@ a avançar na solução à medida que você as for resolvendo.

palavras chave: gnuplot, raíz aproximada, raízes de funções polinomiais, malha, varredura

2 Exercicios

1. gráfico de uma função Considere

$$f(x) = (2x + 1)\sin(\frac{(x + 4)\sqrt{2}}{x + 13})$$

   (a) Faça o gráfico de f com `gnuplot` no intervalo $[-10, 10]$.

   (b) Faça os gráficos das retas tangentes ao gráfico de f no ponto $x = a$; $a \in \{-8, -7, -6, -5\}$. Faça os gráficos simultâneos destas retas, como o gráfico de f.

   sugestão: se inspire em `exer04_02.gnuplot` que você encontra no link "soluçoes" da página.

2. A raíz da reta tangente

   (a) Calcule a raíz (o zero) da reta tangente ao gráfico de

$$f(x) = (2x + 1)\sin(\frac{(x + 4)\sqrt{2}}{x + 13})$$

   no ponto

$$(a, f(a))\ ;\ a = -5.$$

   Chame está raíz de a_1 e calcule

$$b_1 = f(a_1).$$

(b) Faça os gráficos simultâneos de f e da reta tangente.

(c) Faça um zoom no gráfico, use `set xrange` para verificar que a raíz da reta tangente representa uma aproximação da raíz de f.

Calcule o valor $f(a_1)$, lembre-se que `gnuplot` sabe fazer isto, use `print`.

(d) Calcule a raíz (o zero) da reta tangente ao gráfico de $f(x)$ no ponto $(a_1, f(a_1)$. Chame está raíz de a_2 e calcule

$$b_2 = f(a_2).$$

(e) Faça os gráficos simultâneos de f e da reta tangente.

(f) Faça um zoom no gráfico, use `set xrange` para verificar que a raíz da reta tangente representa uma aproximação da raíz de f.

Calcule o valor $f(a_2)$, lembre-se que `gnuplot` sabe fazer isto, use `print`.

(g) Prossiga com o método e calcule $a_3, f(a_3)$ fazendo os gráficos.

Solução 1 `exer04_02.gnuplot` *que você encontra no link "soluções" da página.*

3. A fórmula da raíz da reta tangente Escreva a equação da reta tangente ao gráfico de uma função f no ponto $(a, f(a))$ e escreva a expressão da raíz desta função do primeiro grau.

Solução 2 `exer04_03.png` *que você encontra no link "soluções" da página.*

4. Procura de raízes de uma função

(a) Faça uma varredura inteira do intervalo $[-10, 10]$ para encontrar pontos em que

$$f(x) = (2x+1)\sin(\frac{(x+4)\sqrt{2}}{x+13})$$

troca de sinal. Use $passo = 1$ e trabalhe no intervalo $[-10, 10]$. Experimente com `gnuplot`, faça o gráfico, para ter uma visão antecipada do que vai acontecer. A condição para testar *trocas de sinal* é

```
 if (f(x)*f(x+delta) <= 0)
           printf(''houve troca de sinal no intervalo\n
[\%f, \%f] \n'', x, x+delta);
```

ou em `Pascal`

```
 if (f(x)*f(x+delta) <= 0) then
writeln(''houve troca de sinal no intervalo ['',x,'','',x+delta,'']'');
```

sugestão estude `exer04_04.gnuplot` que você encontra nas soluções da página.

5. Com a informação obtida nos itens anteriores, faça um esboço gráfico de $y = f(x)$ no intervalo $[-10, 10]$. Para isto use um programa que lhe permita desenho à mão livre, sob `LinuX` você encontra `xfig`. Observe que você pode obter o gráfico de f com `gnuplot` para testar se você usou corretamente as informações, faça isto.

6. **Objetivo:** desenvolver a intuição gráfica do efeito de senoides combinadas com polinômios. Repita as questões anteriores com as equações abaixo.

   Você não deve trabalhar excessivamente. Se concluir que *sabe o que está fazendo*, não precisa fazer todos os itens. Você tem aqui uma oportunidade para fazer revisão do Cálculo aplicado na disciplina atual.

$f(x) = (2x-1)\cos(\frac{x+4\sqrt{2}}{x+13})$	$f(x) = (2x-1)\cos(\frac{x+4}{1+x^2})$
$f(x) = \frac{1}{1+x^2}$	$f(x) = \frac{-2x}{1+x^2}$
$f(x) = \sin(x)\frac{-2x}{1+x^2}$	$f(x) = \frac{3\cos(2x)}{1+x^2}$
$f(x) = \frac{4\sin(2x+3)}{1+x^2}$	$f(x) = \frac{x\cos(x/4.0)}{1+x^2}$

7. raíz aproximada - método da tangente

   (a) Faça uma varredura inteira para determinar todos os sub-intervalos de $[-4, 10]$ em que
   $$f(x) = x * sin(\frac{x+6}{x+13})$$
   troca de sinal.

   (b) Em cada subintervalo $[a_i, b_i]$ de $[-4, 10]$ em que houver troca de sinal, de
   $$f(x) = x * sin(\frac{x+6}{x+13})$$
   trace a reta tangente no ponto $(a_i, f(a_i))$, junto com o gráfico da função neste intervalo.

   Você pode fazer isto combinando `gnuplot` e xfig. Com `gnuplot` você pode produzir um gráfico no formato `.fig` que poderá ser modificado com `xfig`. Certamente você pode usar outras combinações de programas gráficos com `gnuplot`, veja os formatos possíveis de gravação do `gnuplot` com `help plot` dentro do `gnuplot` e procure um editor gráfico adequado. Por favor, não me faça perguntas sobre como usar isto dentro do windows (eu não tenho a menor idéia a respeito).

   (c) Em cada subintervalo $[a_i, b_i]$ onde houver troca de sinal, calcule o *zero* a_{i1} da função do primeiro grau tangente e verifique que é uma aproximação do zero de f (calcule $f(a_{i1})$).

(d) Itere o processo anterior considerando agora um dos sub-intervalos

$$[a_i, a_{i1}] \ ; \ [a_{i1}, a_{i+1}]$$

escolhendo aquele onde houver troca de sinal de f, até a quarta iteração. Quer dizer que você deve obter

$$\{a_{i1}, a_{i2}, a_{i3}, a_{i4}\}$$

em cada intervalo I_i e também calcular

$$\{f(a_{i1}), f(a_{i2}), f(a_{i3}), f(a_{i4})\}$$

que deve ser uma sucessão convergindo para zero.

8. Questão discursiva. Defeito do método: O método das tangentes nem sempre funciona como o seguinte exemplo mostra.

(a) Considere $P(x)$ o polinômio do terceiro grau no gráfico (1) página 5, sabendo que o gráfico de P é tangente ao gráfico das retas

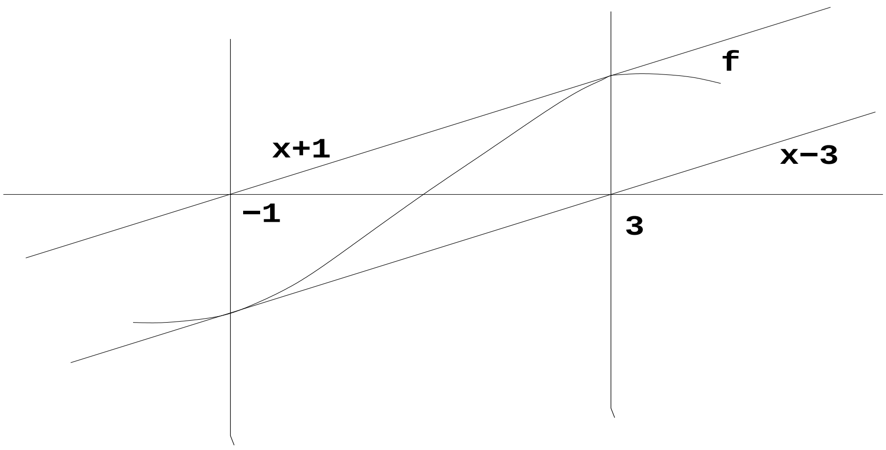

Figura 1: Método da tangente para raízes

$$y = x - 3, y = x + 1$$

obtenha a equação de P.

(b) Justifique porque o método da tangente não fuciona com $y = P(x)$ no intervalo $[-1, 3]$ em que P troca de sinal.

9. Programas

(a) Escreva um programa que encontre os sub-intervalos em que uma função $y = f(x)$ troca de sinal. O programa deve solicitar o usuário os extremos de um intervalo de busca e um passo e varrer o intervalo usando a condição de troca de sinal. Defina uma função no programa para testá-lo e teste o seu programa até que ele esteja funcioando a contento.

(b) Usando derivada aproximada, *quociente de diferenças*, no programa, melhore o programa anterior para calcular também os zeros da derivada.

(c) Inclua no programa uma rotina que, quando for encontrada uma troca de sinal, ela seja chamada para iteradamente aplicar o método da tangente e encontrar uma aproximação das raizes de f usando o teste $|f(x)| < \epsilon$ como teste de parada do algoritmo.

Em `Pascal` pode ser uma *procedure* ou uma `function`. Em `C` é uma função. Elas devem produzir a raíz com a aproximação desejada.

observação: a seu pedido eu posso lhe enviar trechos de programa, mas você tem que me dizer o que deseja de forma justificada. Não me peça programas inteiros!

Referências

[1] Praciano-Pereira, T. *Cálculo numérico Computacional - Introdução à linguagem Pascal*
- Editora da Universidade Estadual Vale do Acaraú - 2000

http://www.4shared.com/file/14206895/ae074651/pascal.html

[2] *Cálculo numérico Computacional* - Edição Eletrônica
Laboratório de Matemática Computacional - 2007

http://www.4shared.com/dir/1751707/4c187abc/sharing.html

[3] Praciano-Pereira, T *Programas para Cálculo Numérico*

http://www.4shared.com/dir/3801087/2fa7cabd/programas.html

[4] *A enciclopédia livre na Internet - Wikipédia*

http://encyclopedia.thefreedictionary.com/

http://en.wikipedia.org/wiki/Fibonacci_number