Computação Numérica

Ano letivo 2011/12

Orientações de resposta ao e-fólio A

1. Recordemos a forma do polinómio de Taylor para $f(x)$ à ordem 3:

$$p_3(x) = f(x_0) + (x - x_0)f'(x_0) + \frac{1}{2!}(x - x_0)^2 f''(x_0) + \frac{1}{3!}(x - x_0)^3 f'''(x_0) + R_3$$

$$|R_3| \leq \left|\frac{1}{4!}(x - x_0)^4 f^{(4)}(\eta)\right| \quad ; \quad \eta \in [x, x_0]$$

com $f^{(4)}$ a quarta derivada de f. Para $x_0 = 0$ temos $f(0) = \text{sen}(0) - 0^2 = 0$ e as derivadas de $f(x) = \text{sen}(x) - x^2$ são, no geral e para x_0,

$$f'(x) = \cos(x) - 2x \quad ; \quad f'(x_0) = f'(0) = 1$$

$$f''(x) = -\text{sen}(x) - 2 \quad ; \quad f''(x_0) = f''(0) = -2$$

$$f'''(x) = -\cos(x) \quad ; \quad f'''(x_0) = f'''(0) = -1$$

$$f^{(4)}(x) = \text{sen}(x)$$

Substituindo tudo isto no polinómio de Taylor temos

$$p_3(x) = 0 + (x - 0) \cdot 1 + \frac{1}{2}(x - 0)^2 \cdot (-2) + \frac{1}{6}(x - 0)^3 \cdot (-1) + R_3 \Leftrightarrow P_3(x)$$

$$= x - x^2 - \frac{1}{6}x^3 + R_3 \quad ; \quad |R_3| \leq \left|\frac{1}{24}x^4 \,\text{sen}(\eta)\right| \quad ; \quad \eta \in [0, x]$$

2. Para $x = 1$ temos então

$$f(1) \approx p_3(1) = 1 - 1^2 - \frac{1}{6} \cdot 1^3 \approx -0{,}1667 \quad ; \quad \epsilon \leq \left|\frac{1}{24}1^4 \,\text{sen}(\eta)\right| \leq \frac{1}{24}1^4 \cdot 1 \leq 0{,}05$$

A explicação para o valor máximo do erro de truncatura é a seguinte: o termo $|\text{sen}(\eta)|$ é certamente igual ou inferior a 1.

Há que notar aqui o seguinte: entre 0 e 1 a função $\text{sen}(x)$ é crescente. Logo, o maior valor de $\text{sen}(\eta)$ entre 0 e 1 é $\text{sen}(1)$. Porque não usar então este valor em ϵ, em vez de 1? Não devemos fazer isso porque $\text{sen}(1)$ é parte do n.º que estamos a tentar achar. Não podemos usar sen (1) para calcular $\text{sen}(1) - 1$! Isso seria raciocínio circular.

Voltando à explicação do erro de truncatura, temos então que o quociente 1/24 é 0,04166... Limitado a 1 algarismo (um erro não precisa de mais do que 1 algarismo significativo; quando muito usa-se 2), nos dá os 0,05 indicados. O valor de $f(1)$ estará assim entre $-0{,}1667 \pm 0{,}05$. Eliminando as 3º e 4ª casas decimais da estimativa, que não têm significado dado serem inferiores ao erro, e arredondando temos

$$f(1) = -0{,}17 \pm 0{,}05$$

i.e. $f(1) \in [-0{,}12\,;\,-0{,}22]$. O intervalo de oscilação é grande, refletindo incerteza elevada na estimativa de $f(1)$ encontrada, e também algum conservadorismo no cálculo de R_3. Para melhorar essa estimativa haveria que aumentar a ordem do polinómio de Taylor.

3. Os erros absoluto e relativo são, designando a nossa estimativa como $\bar{f}(1)$,

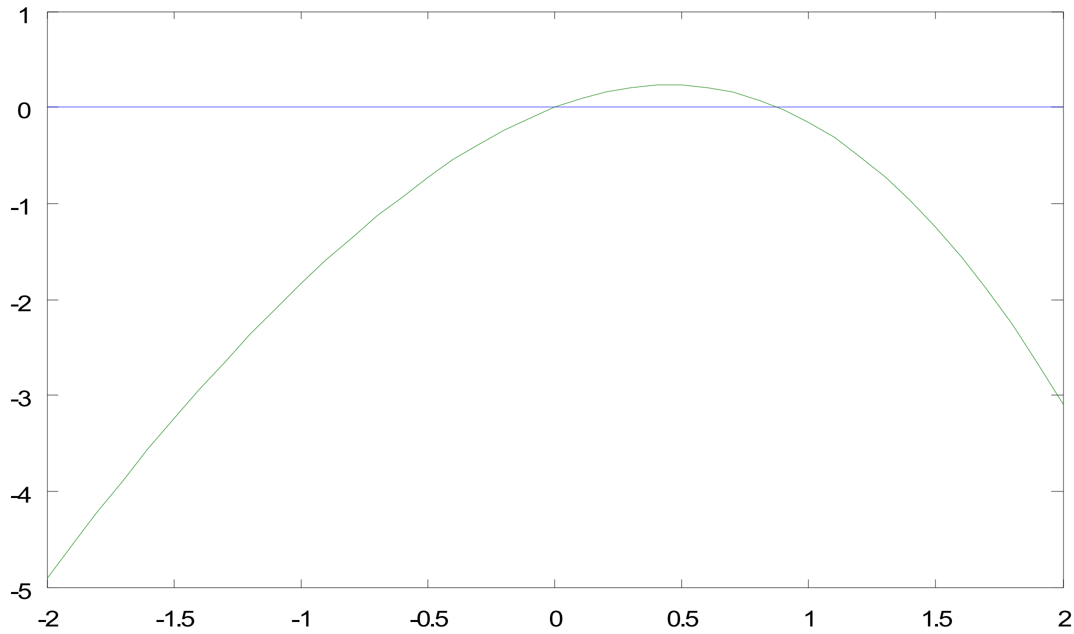
$$\epsilon = \left|f(1) - \bar{f}(1)\right| = |-0{,}1585\,... - (-0{,}17)| = 0{,}0115$$

$$r = \left|\frac{f(1) - \bar{f}(1)}{f(1)}\right| = \left|\frac{-0{,}1585 \ldots - (-0{,}17)}{-0{,}17}\right| = 0{,}068 \quad (6{,}8\%)$$

O erro absoluto é, como esperado, inferior aos 0,05 do valor máximo do erro de truncatura. O erro relativo está no limiar do que é aceitável numa situação prática.

4. Reescrevendo $f(x) = 0 \Leftrightarrow \text{sen}(x) - x^2 + x = x$ temos $g(x) = \text{sen}(x) - x^2 + x$. Uma das raízes de f é obviamente $x = 0$. Outras raízes só poderão estar na vizinhança desta porque o termo $-x^2$ rapidamente arrasta a função para valores muito negativos, longe de $x = 0$. Basta-nos portanto tirar o gráfico de f para valores próximos de zero. Inserindo no Octave os comandos abaixo obtemos o gráfico:

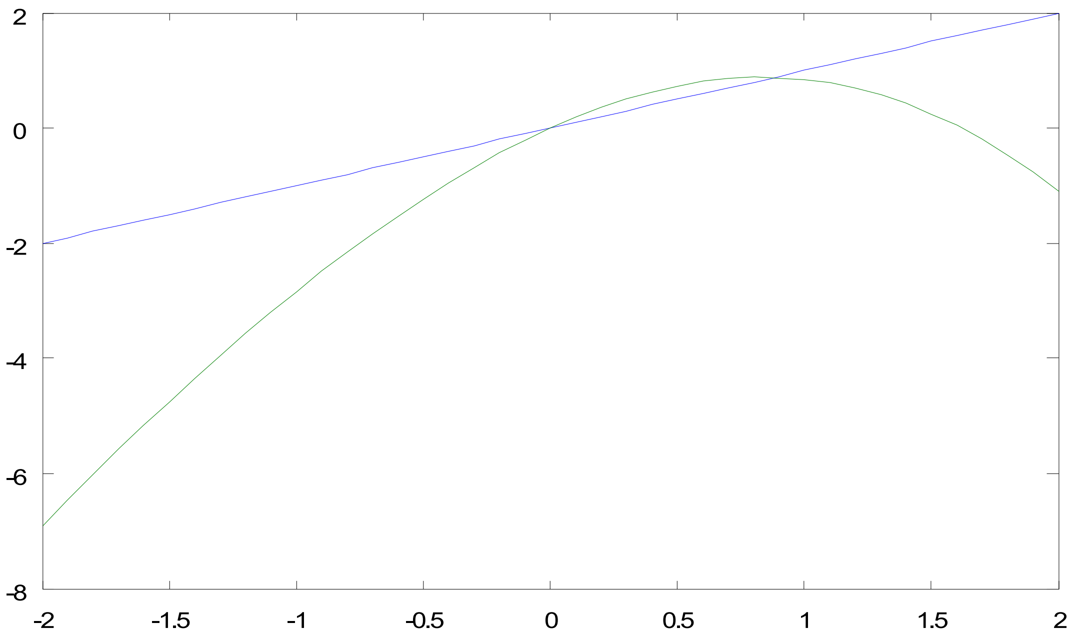
```
> x=(-2:0.1:2) ; y1=x*0 ; y2=sin(x)-x.^2
> plot(x,[y1;y2])
```



Note-se o ponto em `x.^2`. sem ele o Octave tentaria fazer uma multiplicação matricial impossível e daria uma mensagem de erro. Se preferirmos gráficos de $g(x)$ e $y = x$ poderemos fazer, ao invés,

```
> x=(-2:0.1:2) ; y1=x ; y2=sin(x)-x.^2+x
> plot(x,[y1;y2])
```

Nota: podíamos também ter tratado o problema da seguinte forma: $f(x) = 0 \Leftrightarrow \text{sen}(x) = x^2 \Leftrightarrow x = \pm\sqrt{\text{sen}(x)}$. O sinal menos não dará resultados mas o positivo sim. Neste caso $g(x) = \sqrt{\text{sen}(x)}$ e temos uma alternativa para a função g. Como se pode ver, a escolha da função g nem sempre é única. Por vezes uma escolha em particular leva a $|g'(x)| > 1$(método não convergente) e outra a $|g'(x)| < 1$ (método converge). No nosso caso ambas convergem, mas podia não ser assim. Outra hipótese ainda para $g(x)$ seria escolher $x = \frac{\sin(x)}{x} \rightarrow g(x) = \frac{\sin(x)}{x}$.



Como se vê facilmente dos gráficos, a função f tem uma raiz algures entre $x = 0{,}5$ e $x = 1$. Como a raiz parece estar mais perto de 1 do que de 0,5 o ponto 1 será o candidato natural para iniciar as iterações. Isto se o método do ponto fixo for aplicável a este problema, que é o que vamos ver agora.

5. A função auxiliar g tem derivada $g'(x) = \cos(x) - 2x + 1$. Do 2º gráfico vemos que o declive de g junto ao local em que $g(x) = x$ (local da raiz de f) parece ser inferior a 1. Recorrendo à calculadora vemos que em $x = 1$ temos $g'(1) \approx -0{,}46$. Ora, do gráfico vemos também que o declive aumenta em módulo desde o local da raiz até $x = 1$, pelo que concluímos que em toda a região onde queremos aplicar o algoritmo do ponto fixo deveremos ter $|g'(x)| < 1$ e o método deverá assim convergir.

   Este raciocínio não é, de forma alguma, o único a provar que o método converge. Aliás, não é sequer matematicamente rigoroso dado que se baseia em observações gráficas e cálculos elementares. O estudante pode argumentar de qualquer outra forma lógica para justificar a aplicabilidade do ponto fixo.

6. A função g é definida facilmente por

```
function y=f(x) y=sin(x)-x^2+x ; endfunction
```

Uma rotina para o ponto fixo, talvez a mais simples possível, seria

```
function [x,ea]=pontofixo(x0,n);                (1)
     x=zeros(1,1); ea=x; x=x0; ea=f(x0);        (2)
     err=0.5*10^(-n);                           (3)
while ea>err                                    (4)
     ea=abs((f(x)-x)/f(x));                     (5)
     x=f(x)                                     (6)
endwhile                                        (7)
endfunction;                                    (7)
```

Comentários, por linha:

(1) Define a função, de nome 'pontofixo' que devolve dois valores: a estimativa da raiz `x` e o erro da aproximação `ea`. Recebe como input o ponto inicial das iterações `x0` e o n.º de algarismos significativos desejados, `n`.
(2) Define `x` e `ea` como matrizes 1x1 e carrega-os com os seus valores iniciais.
(3) Transforma o n.º de algarismos significativos no valor máximo de erro de aproximação desejado.
(4) Ciclo while que implementa o ponto fixo e que pára assim que o critério de paragem é satisfeito.
(5) Cálculo do erro de aproximação, usando `f(x)` como aproximação atual e `x` como aproximação anterior.
(6) Executa a iteração. Esta linha tem de vir depois da anterior, senão ter-se-ia `ea` = 0 e o ciclo terminaria à primeira iteração.
(7) Fecho de ciclo while e função.

Correndo o comando Octave `[x,ea]=pontofixo(1,5)` obteremos

```
x =  0.84147
x =  0.87902
x =  0.87646
x =  0.87676
x =  0.87672
x =  0.87673
x =  0.87673
ea = 4.4028e-006
```

Se quiséssemos guardar os valores sucessivos das aproximações e do erro de aproximação poderíamos ter refinado o programa para

```
function [x,ea]=PF(x0,n);
     x=zeros(5,1); ea=x; x(1)=x0; ea(1)=f(x(1)); k=1;
     err=0.5*10^(-n);
while ea(k)>err
     x(k+1)=f(x(k));
     ea(k+1)=abs((x(k+1)-x(k))/(x(k+1)));
     k=k+1;
endwhile
endfunction;
```

As variáveis `x` e `ea` começam por ser vetores-coluna de 5 linhas, mas crescem em tamanho se o critério não for atingido em 5 iterações. O resto do programa é simples de entender.