Solución Examen de Métodos Numéricos Dic. 2008

1) a) En Taylor: $y(x_{n+1}) = y(x_n) + y'(x_n)h + y''(x_n)h^2/2 + O(h^3)$
$y'(x_n) = f(x_n, y(x_n))$; $y''(x_n) = f_x(x_n, y(x_n)) + f_y(x_n, y(x_n)).y'(x_n) =$
$f_x(x_n, y(x_n)) + f_y(x_n, y(x_n)).f(x_n, y(x_n))$ (I)
En el método propuesto desarrollando por Taylor (función de 2 variables :
$\Delta x = h/2$, $\Delta y = (h/2)f(x_n, y_n))$:
$y_{n+1} = y_n + h.(f(x_n, y_n) + f_x(x_n, y_n).h/2 + f_y(x_n, y_n).((h/2).f(x_n, y_n)) + O(h^2)) =$
$y_n + h.f(x_n, y_n) + (f_x(x_n, y_n) + f_y(x_n, y_n). f(x_n, y_n)).h^2/2 + O(h^3)$ (II)
Para el estudio del error local se asume que $y(x_n) = y_n$ y por lo tanto :
$y(x_{n+1}) - y_{n+1} =$ Error local $= O(h^3)$, haciendo la resta (I) – (II)
b) Aplicamos el método al problema test : $y' = qy$, $y(0) = 1$ y resulta:
$y_{n+1} = y_n + h.q.(y_n + (h/2).q.y_n) = (1 + hq + (hq)^2 / 2).y_n$
Queda : $y_n = (1 + hq + (hq)^2 / 2)^n y_0 = (1 + hq + (hq)^2 / 2)^n$
Si $z = hq$, la región de estabilidad son los complejos z que cumplen:
$| 1 + z + z^2/2 | < 1$
c) $y' = 2y$, $y(0) = 1$ es un caso particular del problema test con $q = 2$
La solución exacta es una exponencial: $y(x) = e^{2x}$
Si $h = 0.1$ tendremos : $y_{10} = (1 + 2.(0.1) + (2.(0.1))^2 / 2)^{10} = 1.22^{10} = 7.3046314$
Error global en x = 1 : $e^2 - y_{10} = 7.3890561 - 7.3046314 = 8.442*10^{-2}$
Si ahora $h = 0.05$ tendremos : $y_{20} = (1 + 2.(0.05) + (2.(0.05))^2 / 2)^{20} = 1.105^{20} =$
7.36623484
Error global en x = 1 : $e^2 - y_{20} = 7.3890561 - 7.36623484 = 2.282*10^{-2}$
Este error es aproximadamente la cuarta parte del error para h = 0.1.
Este resultado es coherente ya que al pasar de h = 0.1 a h = 0.05 hemos dividido h por 2 y como el método es de orden 2 (Error global = $O(h^2)$) es de esperar que el error global se divida aproximadamente por 4
d) Esta técnica es usada cuando se quiere calcular el valor límite de una determinada función $\overline{f}(x,h)$ cuando el parámetro h tiende a cero.
Consideremos que estamos tratando de calcular $f(x) = \lim_{h\to 0} \overline{f}(x,h)$ y que el error de truncamiento tiene el siguiente comportamiento:
$$\overline{f}(x,h) - f(x) = C_p h^p + O(h^r) \qquad r > p$$
La idea de la extrapolación de Richardson consiste en obtener una mejor aproximación de *f*(*x*) a partir del cálculo de $\overline{f}(x,h)$ para 2 valores diferentes de h
$e^2 = y_{10} + ch^2 + \ldots$; $e^2 = y_{20} + ch^2/4 + \ldots$
Entonces: $4e^2 - e^2 = 4y_{20} - y_{10} + O(h^3)$; $e^2 = (4y_{20} - y_{10})/3 + O(h^3)$
$(4y_{20} - y_{10})/3 = 7.3867693$ El error cometido vale ahora $2.29*10^{-3}$
La extrapolación de Richardson redujo por un factor de 10 al error del mejor valor de la parte c)

2) a) Si A es una matriz m x n el problema de mínimos cuadrados lineal consiste en hallar el valor de un vector x de R^n que minimiza $\| Ax - b \|_2$ donde b es un vector conocido de R^m
Ecuaciones Normales: x es solución del problema de mínimos cuadrados lineal si y sólo si verifica las Ecuaciones Normales: $(A^tA)x = A^tb$
Sea x / $A^t(b - Ax) = 0$. $b - Ay = (b-Ax) + A(x-y)$
$\|b-Ay\|_2^2 = ((b-Ax)^t + (A(x-y))^t) .((b-Ax) + A(x-y)) = \|b-Ax\|_2^2 + \|A(x-y)\|_2^2$
$+ (A(x-y))^t(b-Ax) + (b-Ax)^tA(x-y)$
$(A(x-y))^t(b-Ax) = (x-y)^tA^t(b-Ax) = 0$; $(b-Ax)^tA(x-y) = (x-y)^tA^t(b-Ax) = 0$
Entonces $\|b-Ay\|_2^2 = \|b-Ax\|_2^2 + \|A(x-y)\|_2^2$ y por tanto $\|b-Ay\|_2^2 \geq \|b-Ax\|_2^2$ para cualquier y de R^n. Por tanto x minimiza $\|b-Ay\|_2^2$
Supongamos que $A^t(b - Ax) = z$ distinto del vector nulo.

Si $x - y = -hz$ para $h > 0$ suficientemente pequeño:
$\|b-Ay\|_2^2 = \|b-Ax\|_2^2 + h^2\|Az\|_2^2 - 2h(Az)^t(b-Ax)$; $(Az)^t(b-Ax) = z^tA^t(b-Ax) = \|z\|_2^2$
Entonces: $\|b-Ay\|_2^2 = \|b-Ax\|_2^2 + h^2\|Az\|_2^2 - 2h\|z\|_2^2 < \|b-Ax\|_2^2$ si h es suf. pequeño
y por lo tanto x no minimiza $\|b-Ay\|_2^2$

b) Dada una matriz A m x n , existe una matriz ortogonal Q m x m y una matriz triangular superior R m x n tales que A = QR
$\|b-Ax\|_2^2 = \|b-QRx\|_2^2$ = (Q ortogonal conserva la norma 2) $\|Q^t (b-QRx)\|_2^2 =$
$\|Q^t b-Rx\|_2^2$ $R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$, $Q^t b = \begin{bmatrix} d_1 \\ d_2 \end{bmatrix}$, $\| R - Q^t b \| = \left\| \begin{matrix} R_1 x - d_1 \\ -d_2 \end{matrix} \right\|$ y el mínimo se encuentra cuando x es solución del sistema de n ecuaciones con n incógnitas:
$R_1 x = d_1$

c) Los pasos a seguir son:
i) Hallar la forma económica de la descomposición QR de A por Gram Schmidt
ii) Resolver los s sistemas sobredeterminados.
i) La forma económica de la QR consiste en hallar Q matriz m x n y R matriz triangular superior n x n
El paso iterativo de Gram Schmidt se realiza n veces. En el paso k calculamos básicamente k productos internos de vectores de R^m
Cada producto interno cuesta 2m flops. Tendré en total aprox. 2m(1+2+3+…+n) que es aprox. mn^2 . Además está el cálculo de la nueva columna de Q que se calcula restando a la columa k de A una combinación lineal de las k-1 columnas ya anteriormente obtenidas de Q. Esto genera 2(1+2+3+…+n)m flops más
ii) Para cada sistema hay que hallar $Q^t b_k$. Tiene n posiciones y cada posición cuesta un producto interno de R^m . En total 2mn.
Además está el costo de resolver el sistema triangular superior que es n^2
Esto hay que hacerlo s veces
En total resulta: $2mn^2 + s(2mn + n^2)$ que es $O(mn(n+s))$ ya que $m > n$

d) El método se basa en la iteración de Newton para la resolución del sistema (no lineal) sobredeterminado : $r(x) \approx \vec{0}$. En cada paso se debe resolver un problema de mínimos cuadrados lineal.
En el paso k de la iteración tenemos el punto x_k en el cual efectuamos una aproximación lineal de $r(x)$ de la forma:
$\overline{r_k}(x) = r(x_k) + J(x_k)(x - x_k)$ donde J matriz m x n es la matriz jacobiana de $r(x)$. Si la nueva aproximación es $x_{k+1} = x_k + p_k$ entonces la corrección p_k se calcula como la solución al problema de mínimos cuadrados lineal:
$J(x_k)p_k \approx -r(x_k)$. El iterado siguiente será $x_{k+1} = x_k + p_k$

3) a) Para x_1 : $F(x_1) = a_1\phi(x_1,x_1) + a_2\phi(x_1,x_2) + \dots\dots\dots + a_n\phi(x_1,x_n)$
Para x_k : $F(x_k) = a_1\phi(x_k,x_1) + \dots. + a_r\phi(x_k,x_r) + \dots. + a_n\phi(x_k,x_n)$, k = 1,2,…,n
Resulta un sistema simétrico de matriz n x n cuyas incógnitas son los a_j :

$$\begin{bmatrix} \phi(x_1,x_1) & . & . & . & \phi(x_1,x_n) \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & \phi(x_k,x_r) & . & . \\ \phi(x_n,x_1) & . & . & . & \phi(x_n,x_n) \end{bmatrix} \begin{bmatrix} a_1 \\ . \\ . \\ . \\ a_n \end{bmatrix} = \begin{bmatrix} F(x_1) \\ . \\ . \\ . \\ F(x_n) \end{bmatrix} \quad , \phi(x_k,x_r) = ((x_k - x_r)^2 + c_r^2)^\beta$$

b) $\phi(x_1,x_1) = \phi(x_2,x_2) = \phi(x_3,x_3) = 1; \phi(x_1,x_2) = \phi(x_2,x_3) = \sqrt{1/4+1}; \phi(x_1,x_3) = \sqrt{2}$

$$\begin{bmatrix} 1 & \sqrt{5}/2 & \sqrt{2} \\ \sqrt{5}/2 & 1 & \sqrt{5}/2 \\ \sqrt{2} & \sqrt{5}/2 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 13.0328 \\ -28.1421 \\ 13.0328 \end{bmatrix}$$

$$F^*(x) = a_1\sqrt{x^2+1} + a_2\sqrt{(x-0.5)^2+1} + a_3\sqrt{(x-1)^2+1}$$

c)Error en x = 0.25 = Error en 0.75 = 0.0334

d) $\phi(x_1,x_1) = \phi(x_2,x_2) = \phi(x_3,x_3) = 5; \phi(x_1,x_2) = \phi(x_2,x_3) = \sqrt{1/4+25}; \phi(x_1,x_3) = \sqrt{26}$

$$\begin{bmatrix} 5 & \sqrt{101}/2 & \sqrt{26} \\ \sqrt{101}/2 & 5 & \sqrt{101}/2 \\ \sqrt{26} & \sqrt{101}/2 & 5 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 1025.0 \\ -2060.0 \\ 1025.0 \end{bmatrix}$$

$$F^*(x) = a_1\sqrt{x^2+25} + a_2\sqrt{(x-0.5)^2+25} + a_3\sqrt{(x-1)^2+25}$$

Error en x = 0.25 = Error en 0.75 = 0.0018

Con c_j = 100 resulta:

$$\phi(x_1,x_1) = \phi(x_2,x_2) = \phi(x_3,x_3) = 100; \phi(x_1,x_2) = \phi(x_2,x_3) = \sqrt{1/4+10000}; \phi(x_1,x_3) = \sqrt{10001}$$

$$\begin{bmatrix} 100 & \sqrt{40001}/2 & \sqrt{10001} \\ \sqrt{40001}/2 & 100 & \sqrt{40001}/2 \\ \sqrt{10001} & \sqrt{40001}/2 & 100 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} 0.8001*10^7 \\ -1.6001*10^7 \\ 0.8001*10^7 \end{bmatrix}$$

$$F^*(x) = a_1\sqrt{x^2+10000} + a_2\sqrt{(x-0.5)^2+10000} + a_3\sqrt{(x-1)^2+10000}$$

Error en x = 0.25 = $4.7*10^{-6}$ = Error en 0.75

A medida que c_j crece el error disminuye. Sin embargo las matrices que van quedando son cada vez peor condicionadas. Esto se ve en que las columnas de estas matrices se van poniendo cada vez más colineales con el vector $(1\ 1\ 1)^t$ y por tanto tienden a una matriz singular de rango 1