# Numerical Analysis

Instructor: Wei-Yang Lin

June 25, 2007

**Name** : ______________ **Student ID** : __________________

Final Examination
Monday, June 25, 2007
from 2:45 to 4:00 PM

1. (20 %) Write a `usolve` function to solve $\mathbf{Ax} = \mathbf{b}$ when $\mathbf{A}$ is an upper triangular matrix.

   Answer:

```
function x = usolve(A,b)
[m,n] = size(A);
x = zeros(n,1);
x(n) = b(n)/A(n,n);
for i = n-1:-1:1
   x(i)=(b(i)-A(i,i+1:n)*x(i+1:n))/A(i,i);
end
```

2. (20 %) Write a `powFit` function that calls `linefit` to fit data to $y = c_1 x^{c_2}$.

Answer:

$y = c_1 x^{c_2}$

$\log y = \log c_1 + c_2 \log x$

Let $u = \log x$, $v = \log y$, $\alpha = c_2$, $\beta = \log c_1$

$v = \alpha u + \beta$

```
function c = powFit(x,y)
u = log(x);
v = log(y);
c = linefit(u, v);
c = [exp(c(2)) ; c(1)];
```

3. (20 %) Setup the $4 \times 4$ system of equations for coefficients of the cubic interpolating polynomial in Newton form. Solve this system by hand, showing all intermediate steps, to obtain the coefficients.

Answer:

$$P_3(x) = C_1 + C_2(x - x_1) + C_3(x - x_1)(x - x_2) + C_4(x - x_1)(x - x_2)(x - x_3)$$

$$P_3(x_1) = C_1 = y_1$$
$$P_3(x_2) = C_1 + C_2(x_2 - x_1) = y_2$$
$$P_3(x_3) = C_1 + C_2(x_3 - x_1) + C_3(x_3 - x_1)(x_3 - x_2) = y_3$$
$$P_3(x_4) = C_1 + C_2(x_4 - x_1) + C_3(x_4 - x_1)(x_4 - x_2) + C_4(x_4 - x_1)(x_4 - x_2)(x_4 - x_3) = y_4$$

$$\Rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & (x_2 - x_1) & 0 & 0 \\ 1 & (x_3 - x_1) & (x_3 - x_1)(x_3 - x_2) & 0 \\ 1 & (x_4 - x_1) & (x_4 - x_1)(x_4 - x_2) & (x_4 - x_1)(x_4 - x_2)(x_4 - x_3) \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix}$$
$$= \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & (x_2 - x_1) & 0 & 0 \\ 0 & (x_3 - x_1) & (x_3 - x_1)(x_3 - x_2) & 0 \\ 0 & (x_4 - x_1) & (x_4 - x_1)(x_4 - x_2) & (x_4 - x_1)(x_4 - x_2)(x_4 - x_3) \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix}$$
$$= \begin{bmatrix} y_1 \\ y_2 - y_1 \\ y_3 - y_1 \\ y_4 - y_1 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & (x_3 - x_2) & 0 \\ 0 & 1 & (x_4 - x_2) & (x_4 - x_2)(x_4 - x_3) \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} y_1 \\ f[x_1, x_2] \\ f[x_1, x_3] \\ f[x_1, x_4] \end{bmatrix}$$

$$\because f[x_a, x_b] = \frac{y_b - y_a}{x_b - x_a}$$

$$\Rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & (x_4 - x_3) \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} y_1 \\ f[x_1, x_2] \\ \frac{f[x_1,x_3] - f[x_1,x_2]}{x_3 - x_2} \\ \frac{f[x_1,x_4] - f[x_1,x_2]}{x_4 - x_2} \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} f[x_1] = y_1 \\ f[x_1, x_2] \\ \frac{f[x_1,x_3] - f[x_1,x_2]}{x_3 - x_2} = f[x_1, x_2, x_3] \\ \frac{f[x_1,x_2,x_4] - f[x_1,x_2,x_3]}{x_4 - x_3} = f[x_1, x_2, x_3, x_4] \end{bmatrix}$$

4. (20 %) Manually evaluate the following integral with the trapezoid rule using two panels, and compare the result with the exact analytical value of the integral.

$$\int_0^2 xe^{-x}dx$$

*Hint*: The integration by parts rule states that

$$\int_a^b f(x)g'(x)dx = f(x)g(x)|_a^b - \int_a^b f'(x)g(x)dx$$

Answer:

$f(x) = x$
$g'(x)dx = e^{-x}dx \Rightarrow g(x) = -e^{-x}$

$$\int_0^2 xe^{-x}dx = (-x)\cdot e^{-x}|_0^2 + \int_0^2 e^{-x}dx = (-x)\cdot e^{-x}|_0^2 + (-e^{-x})|_0^2$$
$$= -2e^{-2} - e^{-2} + 1 = 1 - 3e^{-2}$$

$n = 2 + 1 = 3,\ h = \dfrac{2-0}{3-1} = 1,\ x = 0, 1, 2;$

$f_1 = f(0) = 0 \times e{-}0 = 0$
$f_2 = f(1) = 1 \times e^{-1} = e^{-1}$
$f_3 = f(2) = 2 \times e^{-2} = 2e^{-2}$
$I = 1 \times (0.5 \times 0 + e^{-1} + 0.5 \times (2e^{-2})) = e^{-1} + e^{-2}$

5. (20 %) Write an m-file to return the value of

$$p(x) = \frac{1}{\sqrt{2\pi}} \int_{-s}^{s} e^{-z^2/2} dz$$

Use composite Gauss-Legendre quadrature with six panels and four nodes per panel. Note that you have to implement the procedure for computing the nodes and weights.

Answer:

```
function I = gaussQuad(fun, a, b, upanel, nnode)
[z,wt] = GLNodeWt(nnode);
H = (b-a)/npanel;
H2 = H/2;
x = a:H:b;
I = 0;
for i = 1:npanel
   xstar = 0.5*(x(i)+x(i+1)) + H2*z;
   f = feval(fun, xstar);
   I = I + sum(wt.*f);
end
I = I*H2;

function [x,w] = GLNodeWt(n)
beta = (1:n-1)./sqrt(4*(1:n-1).^2 - 1);
J = diag(beta,-1) + diag(beta,1);
[V,D] = eig(J);
[x,ix] = sort(diag(D));
w = 2*V(1,ix)'.^2;

function f = fun(x)
f = exp(-0.5*x.^2) / sqrt(2*pi);
```