Universidad de Granada
Facultad de Ciencias
Dept. Matemática Aplicada

MÉTODOS NUMÉRICOS
Grupo A
Licenciatura en Matemáticas

1ª prueba de prácticas

10-feb-06

Ecuaciones de una variable - S.E.L. - Interpolación (polinómica)

1. a) Calcula todas las soluciones reales de la ecuación $f(x) = 0$ siendo

$$f(x) = 1/4 - \sum_{n=1}^{10} \frac{(-1)^{n+1}}{n} \left(\frac{x}{5}\right)^n.$$

Solución: Dos soluciones reales en $x = 1.4201275$ y $x = 6.333979$.

```
In[1]   f[x_]:=1/4-Sum[(-1)^(k+1)*(x/5)^k/k,{k,1,10}]
        NSolve[f[x]==0,x]
```

b) Aplica el método iterativo siguiente

$$x_{n+1} = x_n - \left(1 + \frac{f(x_n)\,f''(x_n)}{2(f'(x_n))^2}\right)\frac{f(x_n)}{f'(x_n)}$$

para aproximar la menor de las soluciones positivas calculadas, comenzando con $x_0 = 4.1$ y deteniendo el proceso cuando la distancia entre iteraciones sucesivas sea menor que 10^{-6}.

Solución: En 5 iteraciones se detiene, dando como resultado 1.4201275

```
In[3]   x=4.1;
        g[x_]:=x-(1+f[x]*f''[x]/(2f'[x]^2))*(f[x]/f'[x]);
        For[i=1,i<=100,i++,y=g[x];If[Abs[x-y]<10^(-6),Break[],x=y]]
        Print["sol.=", y, "    iteraciones=", i]
```

2. Aproxima el valor de $I = \int_{-1}^{1} e^{-x^2}\,dx$ por el de la integral del polinomio que interpola a $h(x) = e^{-x^2}$ usando datos:

a) ... de tipo Lagrange en los nodos $\left\{-\frac{2}{3},\ 0,\ \frac{2}{3}\right\}$

Solución: Valor de la integral aproximada $= 1.4617705826$.

```
In[7]   nodos={-2/3,0,2/3};
        h[z_]:=E^(-z^2);
        datosL=Transpose[{nodos, g[nodos]}];
        p1=InterpolatingPolynomial[datosL,z]
        int1=Integrate[p1,{z,-1,1}]//N
```

b) ... y de tipo Hermite en los mismos nodos.

Solución: Valor de la integral aproximada $= 1.5005421214188$

```
In[12]  datosH=Table[{nodos[[i]],{g[nodos[[i]]],g'[nodos[[i]]]}},{i,3}];
        p2=InterpolatingPolynomial[datosH,z]
        int2=Integrate[p2,{z,-1,1}]//N
```

c) Dibuja conjuntamente las gráficas de ambos polinomios y de la propia función e^{-x^2} en el intervalo $[-2,\ 2]$ identificando cada una de ellas.

Solución: h en negro, p1 en rojo, p2 en azul.

```
In[15] Plot[{p1,p2,g[z]},{z,-2,2},
       PlotStyle->{RGBColor[1,0,0],RGBColor[0,0,1],RGBColor[0,0,0]}]
```

d) Mediante el comando `NIntegrate[E^(-z^2),{z,-1,1}]` calcula el valor de I e indica el error relativo cometido al aproximar I por la integral de los dos polinomios de interpolación.

Solución: Error con Lagrange $= 0.021342$; error con Hermite $= 0.004615$.

```
In[17] sol=NIntegrate[E^(-z^2),{z,-1,1}]
       eL=Abs[int1-sol]/Abs[sol]
       eH=Abs[int2-sol]/Abs[sol]
```

$\boxed{\textbf{3.}}$ a) Calcula la solución del sistema $Ax = b$ siendo:

$$A = \begin{pmatrix} 0 & \cdots & \cdots & \cdots & \cdots & 0 & 1 \\ 1 & 0 & & & & \vdots & \frac{60}{61} \\ 0 & 1 & 0 & & & \vdots & \frac{59}{61} \\ \vdots & \ddots & \ddots & \ddots & & \vdots & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & & & \ddots & 1 & 0 & \frac{2}{61} \\ 0 & \cdots & \cdots & \cdots & 0 & 1 & \frac{1}{61} \end{pmatrix}_{61\times 61} \qquad b = \frac{1}{61}\begin{pmatrix} 30 \\ 29 \\ \vdots \\ 2 \\ 1 \\ 0 \\ 1 \\ 2 \\ \vdots \\ 29 \\ 30 \end{pmatrix}_{61\times 1}.$$

Solución:

```
In[20] n=61;
       A=Table[0,{n},{n}];
       Do[A[[i+1,i]]=1,{i,n-1}]
       Do[A[[i, n]]=(62-i)/n,{i,n}]
       b=(1/n)*Table[Abs[31-i],{i,n}];
       x=LinearSolve[A,b];
```

b) Llamando $x = (x_1, x_2, \ldots, x_{61})$ a la solución obtenida, crea una matriz 61×2 llamada *lista* cuyas filas sean los pares de la forma (i, x_i), para cada $i = 1, \ldots, 61$, y dibuja esos pares usando el comando **ListPlot[***lista***]**.

Solución:

```
In[26] lista=Table[{i,x[[i]]},{i,n}];
       ListPlot[lista]
```

c) Calcula el condicionamiento de A usando tanto la norma matricial $\|\cdot\|_\infty$ como la $\|\cdot\|_1$.

Solución: $K_\infty(A) = 14641/3721 = 3.93469$ y $K_1(A) = 961$.

```
In[28] B=Inverse[A];
       normainfA=Max[Table[Sum[Abs[A[[i,j]]],{j,n}],{i,n}]];
       normainfB=Max[Table[Sum[Abs[B[[i,j]]],{j,n}],{i,n}]];
       condicinfinito=normainfA*normainfB


In[32] norma1A=Max[Table[Sum[Abs[A[[i,j]]],{i,n}],{j,n}]]
       norma1B=Max[Table[Sum[Abs[B[[i,j]]],{i,n}],{j,n}]]
       condicuno=norma1A*norma1B
```