MA-0501 Análisis Numérico I
TAREA 1

Fecha de entrega: Lunes 07 de octubre, 11:59pm.
Instrucciones:

- La entrega de la tarea debe realizarse en la plataforma de Mediación Virtual, antes de la fecha y hora establecida.
- Debe subir dos archivos:
  - Un archivo en formato pdf con la resolución de los ejercicios y la discusión de los resultados. El documento debe llamarse `ApellidosNombre.pdf`
  - Un archivo `MATLAB` (`ApellidosNombre.m` o `ApellidosNombre.mlx`) con todos los códigos implementados, debidamente documentados y listos para ser ejecutados.
- Recuerde que *una imagen dice más que mil palabras*. Al mostrar resultados, presente gráficos claros debidamente rotulados, con escalas y leyendas adecuadas.
- El trabajo se puede realizar individualmente o en parejas. Si aplica, debe citar cualquier colaboración o referencia utilizada. En caso de ser en parejas, incluir en el nombre de los archivos `Apellido1Nombre1_Apellido2Nombre2.m`. Basta subir una entrega por pareja.
- Se evaluará sobre un total de 100 puntos; en caso de obtener un mayor puntaje, dichos puntos no serán acumulables ni transferibles.

Respuesta corta.

1. (2 puntos) Describa un algoritmo determinar los pesos w_i de la fórmula de Newton-Cotes para $n = 3$ en el intervalo $[-1, 1]$, la cual viene dada por

$$\int_{-1}^{1} f(x)\, dx \approx w_0 f(-1) + w_1 f(-1/3) + w_2 f(1/3) + w_3 f(1).$$

   No es necesario calcular los valores exactos.

2. (2 puntos) Describa un método, con base en lo estudiado en clase, para aproximar la integral impropia

$$I = \int_0^1 \frac{\sin(x)}{x^{3/2}}\, dx$$

   de forma tal que pueda garantizar que el error es menor a una tolerancia dada.

3. (2 puntos) En una regla de cuadratura en el intervalo $[0, 1]$ con $n+1$ nodos, se fijan 3 nodos internos. ¿Cuál es el mayor grado de un polinomio que se puede integrar de manera exacta? Justifique de forma general.

Desarrollo

1. (6 puntos) Considere $n+2$ puntos $\{(x_i, y_i)\}_{i=0}^{n+1}$ (donde los x_i son distintos). Sea q el polinomio de Lagrange de grado n que interpola los puntos $\{(x_i, y_i)\}_{i=0}^{n}$ y sea r el polinomio de Lagrange de grado n que interpola los puntos $\{(x_i, y_i)\}_{i=1}^{n+1}$. Defina

$$p(x) := \frac{(x - x_0)r(x) - (x - x_{n+1})q(x)}{x_{n+1} - x_0}.$$

Demuestre que p es el polinomio de grado $n+1$ que interpola todo el conjunto de datos.

2. En este ejercicio se demostrará cómo obtener la fórmula de Gauss de una forma diferente a la vista en clase. Dados n natural, un intervalo $[a, b]$ y una función peso $w : (a, b) \to \mathbb{R}^+$, considere el polinomio ortogonal $\phi_{n+1}(x)$ de grado $n+1$. En clase demostramos que sus $n+1$ ceros, los cuales denotamos por $\{x_0, \ldots, x_n\}$, son distintos y están en el intervalo (a, b).

   a) (3 puntos) Sea p un polinomio de grado a lo sumo $2n+1$. Por el algoritmo de la división, es posible escribir $p(x) = \phi_{n+1}(x)q(x) + r(x)$, para polinomios apropiados q, r. Demuestre que

   $$\int_a^b p(x)w(x)\ dx = \int_a^b r(x)w(x)\ dx.$$

   b) (3 puntos) Exprese r como combinación lineal de los polinomios de Lagrange $\{L_i\}_{i=0}^{n}$ (asociados a los nodos $\{x_0, \ldots, x_n\}$). Justifique por qué esto es posible.

   c) (3 puntos) Demuestre que

   $$\int_a^b p(x)w(x)\ dx = \sum_{i=0}^{n} w_i p(x_i)$$

   para valores apropiados de $\{w_i\}_{i=0}^{n}$. Dé explícitamente la fórmula para cada w_i. ¿Coincide con la fórmula dada en clase?

   d) (1 punto) Concluya que la cuadratura $\displaystyle\sum_{i=0}^{n} w_i p(x_i)$ es exacta para cualquier polinomio de grado a lo sumo $2n+1$.

3. En este ejercicio se deducirá una manera diferente para definir los nodos y pesos de la cuadratura de Gauss. Considere un conjunto de polinomios ortogonales mónicos[1] $\{\phi_j\}_{j=0}^{\infty}$ en el intervalo $[a,b]$, correspondientes a una función peso $w(x)$.

   a) (6 puntos) Demuestre que existen sucesiones $\{a_k\}_{k=1}^{\infty}$, $\{b_k\}_{k=0}^{\infty}$, tales que los polinomios ortogonales satisfacen la relación de recurrencia

   $$\begin{aligned} \phi_1(x) &= (x-b_0)\phi_0(x), \\ \phi_{k+1}(x) &= (x-b_k)\phi_k(x) - a_k\phi_{k-1}(x) \ \forall\ k \geq 1. \end{aligned}$$

   Dé fórmulas explícitas para el término general de cada sucesión[2]. Sugerencia: Por algoritmo de la división, para cada $k \geq 1$ se tiene que $\phi_{k+1}(x) - x\phi_k(x)$ es un polinomio de grado k.

   b) (4 puntos) Considere ahora polinomios ortonormales $\{\widetilde{\phi}_j\}_{j=0}^{\infty}$. Escribiendo $\phi_k = \|\phi_k\|\widetilde{\phi}_k$ y las fórmulas de los coeficientes del inciso anterior, deduzca que existen sucesiones $\{\alpha_k\}_{k=1}^{\infty}$, $\{\beta_k\}_{k=0}^{\infty}$ tales que

   $$\begin{aligned} \alpha_1\widetilde{\phi}_1(x) + \beta_0\widetilde{\phi}_0(x) &= x\widetilde{\phi}_0(x), \\ \alpha_{k+1}\widetilde{\phi}_{k+1}(x) + \beta_k\widetilde{\phi}_k(x) + \alpha_k\widetilde{\phi}_{k-1}(x) &= x\widetilde{\phi}_k(x) \ \forall\ k \geq 1. \end{aligned} \tag{1}$$

   c) (4 puntos) Para n fijo, considere los ceros $\{x_0, \ldots, x_n\}$ del polinomio $\widetilde{\phi}_{n+1}$. Evalúe (1) (para $1 \leq k \leq n$) en un cero x_j . Concluya que x_j es un cero del polinomio ortogonal $\widetilde{\phi}_{n+1}$ si y solo si es un valor propio de la matriz T de tamaño $(n+1) \times (n+1)$ dada por

   $$T = \begin{bmatrix} \beta_0 & \alpha_1 & & & & \\ \alpha_1 & \beta_1 & \alpha_2 & & & \\ & \alpha_2 & \beta_2 & \alpha_3 & & \\ & & \ddots & \ddots & \ddots & \\ & & & \alpha_{n-1} & \beta_{n-1} & \alpha_n \\ & & & & \alpha_n & \beta_n \end{bmatrix}.$$

   d) (4 puntos) Para $i, j \in \{0, \ldots, n\}$, utilice el hecho que la cuadratura de Gauss integra de forma exacta $\widetilde{\phi}_i\widetilde{\phi}_j$ para deducir que

   $$\delta_{ij} = \sum_{k=0}^{n} w_k\widetilde{\phi}_i(x_k)\widetilde{\phi}_j(x_k).$$

[1] En este caso, primero normalizamos los polinomios de forma tal que el coeficiente del término de mayor grado es 1.

[2] Deben quedar en términos del producto interno $(f,g)_w = \int_a^b fgw$ o la norma asociada $\|f\| = ((f,f)_w)^{1/2}$.

Concluya que $P^TWP = I$, donde I es la matriz identidad de tamaño $(n+1) \times (n+1)$, y las matrices W, P vienen dadas por

$$W = \begin{bmatrix} w_0 & 0 & \dots & 0 \\ 0 & w_1 & \ddots & \vdots \\ \vdots & \ddots & & 0 \\ 0 & \dots & 0 & w_n \end{bmatrix}, P = \begin{bmatrix} \widetilde{\phi}_0(x_0) & \dots & \widetilde{\phi}_n(x_0) \\ \vdots & \ddots & \vdots \\ \widetilde{\phi}_0(x_n) & \dots & \widetilde{\phi}_n(x_n) \end{bmatrix}.$$

e) (2 puntos) Es posible demostrar que $W^{-1} = PP^T$. Deduzca así que

$$\frac{1}{w_j} = \sum_{i=0}^{n} (\widetilde{\phi}_i(x_j))^2.$$

f) (2 puntos) Considere ahora el vector propio unitario $v^{(j)} = [v_1^{(j)}, \dots, v_{n+1}^{(j)}]^T$ de la matriz T asociado al valor propio (nodo) x_j. Demuestre que existe una constante C tal que $v^{(j)} = C[\widetilde{\phi}_0(x_j), \dots, \widetilde{\phi}_n(x_j)]^T$.

g) (2 puntos) Demuestre que $C = \sqrt{\mu} v_1^{(j)}$, con $\mu = \int_a^b w(x)\, dx$. Sugerencia: utilice el hecho que $(\widetilde{\phi}_0, \widetilde{\phi}_0)_w = 1$.

h) (2 puntos) Concluya que los pesos de la cuadratura se pueden escribir como $w_j = \mu(v_1^{(j)})^2$.

i) (8 puntos) [`MATLAB`] Para el caso $w(x) = 1$ en $[-1, 1]$, escriba una función que calcule la matriz T para n dado, y sus valores y vectores propios. Utilice las fórmulas demostradas anteriormente para que la función retorne los pesos y nodos de la cuadratura de Gauss. Sugerencia: para calcular los valores y vectores propios utilice la función `[V,D]=eig(T)`.

j) (2 puntos) [`MATLAB`]Verifique que para $n = 0$ su programa retorna $x_0 = 0, w_0 = 2$, y que para $n = 1$ retorna $x_0 = -1/\sqrt{3}, x_1 = 1/\sqrt{3}, w_0 = w_1 = 1$.

k) (4 puntos) [`MATLAB`] Construya una cuadratura que permita integrar $f(x) = x^8 + 2x^2 + x$ de manera exacta. Utilice el mínimo valor de n posible. Escriba los nodos y pesos que utilizó, y verifique que obtiene el resultado correcto.

l) (6 puntos) [`MATLAB`] Implemente una función que calcule la fórmula compuesta de la cuadratura de Gauss. Para ello, considere como entrada el intervalo $[a, b]$, el número de subintervalos m, el valor n para la cuadratura de Gauss y la función f. Dicha función debe calcular una partición uniforme $\{x_j\}_{j=0}^m$ dada por

$$x_j := a + jh \ (j = 0, \dots, m), \quad \text{con} \ \ h = \frac{b-a}{m},$$

y calcular la fórmula

$$\int_a^b f(x)\,dx \approx \frac{h}{2}\sum_{j=1}^{m}\sum_{k=0}^{n} W_k f\left(\frac{1}{2}(x_{j-1}+x_j)+\frac{1}{2}h\widetilde{x}_k\right)\,dt,$$

donde $\{\widetilde{x}_k\}_{k=0}^{n}$ son los ceros de un polinomio ortogonal de grado $n+1$ en el intervalo $[-1,1]$ y w_k los pesos asociados (que calculó anteriormente).

m) (4 puntos) [`MATLAB`] Considere $g(x) = \operatorname{sen}(100\pi x)(1-x)^{1/2}\log(1-x)$ en el intervalo $[0,1]$. Grafique, para $n \in \{0,2,5,8\}$, el error absoluto de esta cuadratura compuesta en función de m (una curva para cada valor de n). Comente sus resultados. Utilice $I_g = 0.003940021793519$ como valor exacto.

4. En este ejercicio se busca construir una aproximación para la derivada de una función, a partir de su polinomio de interpolación de una manera diferente a las fórmulas estudiadas de $n+1$ puntos del Capítulo 3.

   a) (4 puntos) Demuestre que

   $$L_k(x) = \frac{\lambda_k}{x-x_k}\Bigg/\sum_{j=0}^{n}\frac{\lambda_j}{x-x_j}, \text{para } x \notin \{x_0,\ldots,x_n\},$$

   donde $\{L_k\}$ es la base de polinomios de interpolación de Lagrange. Deduzca que

   $$L_k(x)s(x) = \lambda_k\frac{x-x_i}{x-x_k},$$

   donde $s(x) = \sum_{j=0}^{n}\lambda_j\frac{x-x_i}{x-x_j}$, $i \in \{0,1,\ldots,n\}$.

   b) (2 puntos) Del inciso (a), deduzca que

   $$L_k'(x)s(x) + L_k(x)s'(x) = \lambda_k\left(\frac{x-x_i}{x-x_k}\right)'.$$

   c) (2 puntos) Verifique que $s(x_i) = \lambda_i$ y deduzca que

   $$L_k'(x_i) = \frac{\lambda_k}{\lambda_i}\frac{1}{x_i-x_k}(i \neq k).$$

   d) (2 puntos) Demuestre que

   $$L_k'(x_k) = -\sum_{i\neq k} L_i'(x_k).$$

e) (6 puntos) Denote la matriz $D \in \mathbb{R}^{n+1\times n+1}$ cuyas entradas vienen dadas por $D_{ik} = L'_k(x_i)$, para $i, j \in \{0, 1, \ldots, n\}$. ¿Qué representa el vector $Df(\hat{\boldsymbol{x}})$, donde $\hat{\boldsymbol{x}}$ es el vector de nodos de Chebyshev y $f(\hat{\boldsymbol{x}})$ el vector con las imágenes de dichos nodos?

f) (6 puntos) [`MATLAB`] Implemente una función que reciba n y calcule la matriz D del inciso anterior. Para $n = 20, f(x) = 1/(1 + 16x^2)$, calcule y grafique $f'(x)$, junto a los puntos $(\hat{\boldsymbol{x}}, Df(\hat{\boldsymbol{x}}))$. Cuantifique el error en los nodos de interpolación en esta gráfica.

g) (3 puntos) [`MATLAB`] Grafique $\|Df(\hat{\boldsymbol{x}}) - f'(\hat{\boldsymbol{x}})\|_\infty$ en función de n. ¿Cuál es el menor error posible?

5. Es posible escribir el polinomio de interpolación de Lagrange p en la base de los polinomios de Chebyshev,

$$p(x) = \sum_{k=0}^{n} c_k T_k(x), \quad |x| \leq 1,$$

donde $T_k(x) = \cos(k \arccos x)$. Sabemos que los polinomios de Chebyshev satisfacen la relación de recurrencia

$$T_{k+1}(x) = \alpha_k(x) T_k(x) + \beta_k(x) T_{k-1}(x),$$

para $\alpha_k(x) = 2x$ y $\beta_k(x) = -1$. Deseamos calcular $p(x_0)$ para x_0 dado (asumiendo que lo tenemos expresado en esta base). Para ello, considere el siguiente algoritmo:

Alg.1– Defina la sucesión $\{b_k(x_0)\}_{k=0}^{n+2}$ mediante la recurrencia hacia atrás:

$$\begin{aligned} b_{n+2}(x_0) &= b_{n+1}(x_0) = 0, \\ b_k(x_0) &= c_k + \alpha_k(x_0) b_{k+1}(x_0) + \beta_{k+1}(x_0) b_{k+2}(x_0) \quad \text{para } k = n, n-1, \ldots, 1. \end{aligned}$$

Se tiene entonces que el valor deseado es

$$p(x_0) = T_0(x_0)c_0 + T_1(x_0)b_1(x_0) + \beta_1(x_0)T_0(x_0)b_2(x_0). \tag{2}$$

a) (2 puntos) Determine el número de operaciones (sumas y multiplicaciones) que se deben realizar en el Algoritmo 4.

b) (4 puntos) Escriba una función `y0 = evalCheb(c,x0)` en `MATLAB` que calcule $y_0 = p(x_0)$ mediante la fórmula (2), donde la entrada es el vector de coeficientes $\boldsymbol{c}$ y el valor x_0.

c) (3 puntos) Considere el polinomio $p(x) = x^3 - 3x^2 + 1$. Exprese $p(x)$ como combinación lineal de polinomios de Chebyshev

$$p(x) = c_0T_0(x) + c_1T_1(x) + c_2T_2(x) + c_3T_3(x)$$

y verifique el comportamiento del algoritmo para $x_0 = 1$.

d) (3 puntos) Utilice `c = rand(n,1)`, $n = 10^8$, $x_0 = 0.1$ y reporte el tiempo de ejecución (puede realizar varias corridas y mostrar un tiempo promedio). Compare con el tiempo que requiere `polyval` para evaluar un polinomio del mismo grado. Comente sus resultados.