Universidad Diego Portales
Facultad de Ingeniería y Ciencias
Instituto de Ciencias Básicas

Métodos Numéricos
1er Semestre 2018

Prof. Julio López, Ayund. Javiera Araya Plazo máximo de entrega: 11 de Junio del 2018.

Tarea computacional No 2

- La entrega del trabajo es forma grupal (a lo más 4 integrantes) y estos deben ser enviados a los emails: julio.lopez@udp.cl, javiera.arayamu@mail.udp.cl, en un archivo comprimido zip o rar, usando el siguiente formato:

  `MNT1_Apellido1_Apellido2_Apellido3_Apellido_4.`

- Pasada el plazo máximo de entrega, se recibirá los trabajos pero con un máximo a obtener de 6 puntos, el cual ira disminuyendo con relación a la fecha de entrega.

Resolución de Sistemas de Ecuaciones Lineales

Sea $A \in \mathbb{R}^{n\times n}$ una matriz invertible y $b \in \mathbb{R}^n$. Considere el siguiente problema:

$$(P) \qquad \text{Encontrar un } \mathbf{x} \in \mathbb{R}^n \text{ tal que } A\mathbf{x} = b.$$

Programación:

- Supongamos que la matriz A es triangular inferior (resp. superior). Programe el método de sustitución adelante (resp. atrás). Verifique que sus códigos funcionan, incluyendo una matriz de tamaño 4×4.
- Programe las factorizaciones LU, PA=LU (pivoteo parcial) y Cholesky de la matriz A. Tenga en cuenta cuando existen estas factorizaciones. Incluya un ejemplo de una matriz de tamaño 3×3.
- Programe los métodos iterativos: Richardson, Jacobi, Gauss-Seidel y SOR. En tales métodos incluya uno de los siguientes criterios de parada

$$\frac{\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|}{\|\mathbf{x}^{(k+1)}\|} \leq \text{Tol}_{\text{rel}}, \quad \text{ó} \quad \|r^{(k+1)}\| = \|A\mathbf{x}^{(k+1)} - b\| \leq \text{Tol}_{\text{res}}.$$

  El tipo de norma a usar es a libre elección. Para efectos de analizar la rápidez de convergencia de estos métodos, incluya el cálculo del radio espectral de la matriz de iteración. (Hint. Usar la función **eig.m** de MATLAB para el cálculo de los valores propios de una matriz). Verifique que sus códigos funcionan, incluyendo una matriz de tamaño 4×4.

Aplicación de los esquemas programados:

1. Considere la matriz $A = (a_{ij})$, definida por

$$a_{ij} = \begin{cases} i(n-j+1) & , \quad i \leq j, \\ a_{ji} & , \quad i > j, \end{cases} \qquad i,j = 1,2,\dots,n.$$

   (a) Usando la norma $\|\cdot\|_1$, cálcule el número condición de la matriz A, para $n = 5$ hasta $n = 50$. Represente en una gráfica el resultado obtenido. Que puede decir de la matriz A.

   (b) Encuentre la factorizaciones LU, PA=LU, y Cholesky (si es que existe), cuando $n = 5, 10$.

(c) Sea $b = (1, 2, \dots, n) \in \mathbb{R}^n$. Usando Eliminación Gaussiana (sin pivoteo) encuentre la solución del problema (P), cuando $n = 10, 15, 20$.

(d) Use los métodos iterativos: Jacobi, Gauss-Seidel y Relajación (con $\omega = 0.7$, $\omega = 1.5$, $\omega = 1.9$) para encontrar la solución del problema (P) cuando $n = 10, 15, 20$, y tomando como punto inicial el vector $\mathbf{x}^{(0)} = (1, \dots, 1)$. Compare sus resultados y comente sobre la convergencia.

2. Considere la función $f(t) = \dfrac{1}{1 + 25t^2}$ definida en el intervalo $[-1, 1]$, y sobre este intervalo, considere los siguientes m puntos:

$$t_i = -1 + (i-1)\frac{2}{m-1}, \quad i = 1, \dots, m.$$

Deseamos encontrar de forma aproximada la gráfica de la función f. Para esto, consideramos el siguiente polinomio de grado $n - 1$:

$$g(t) = \beta_1 + \beta_2 t + \beta_3 t^2 + \cdots + \beta_n t^{n-1}.$$

El objetivo es determinar los valores de β_i talque $g(t_i) = f(t_i) = \dfrac{1}{1 + 25t_i^2}$, para $i = 1, \dots, m$. Esta relación, nos conlleva al siguiente sistema lineal de ecuaciones:

$$\begin{bmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{n-1} \\ 1 & t_2 & t_2^2 & \cdots & t_2^{n-1} \\ \vdots & \vdots & \vdots & \vdots & \\ 1 & t_m & t_m^2 & \cdots & t_m^{n-1} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{bmatrix} = \begin{bmatrix} f(t_1) \\ f(t_2) \\ \vdots \\ f(t_m) \end{bmatrix}.$$

(a) Suponiendo que $m = n$, encuentre la solución del sistema anterior vía eliminación Gaussiana (sin pivoteo), cuando $n = 10, 20, 30$.

(b) Gráfique la función f versus el polinomio g para $n = 10, 20, 30$. Comente sus resultados.

Método de Gradiente conjugado (Hestenes-Stiefel version):

Es un método iterativo para encontrar de forma aproximada la solución de un sistema lineal de la forma (P), cuando la matriz A es simétrica y definida positiva. El algoritmo para el método de gradiente conjugado es el siguiente:

---

procedure ALGORITMO CGM$(A, b, \mathbf{x}_0, \varepsilon)$
    $\mathbf{r_0} := \mathbf{b} - \mathbf{A}\mathbf{x_0}$.
    $\mathbf{p_0} := \mathbf{r_0}$.
    $k := 0$.
    repeat
        $\alpha_k := \dfrac{\mathbf{r_k}^\top \mathbf{r_k}}{\mathbf{p_k}^\top \mathbf{A} \mathbf{p_k}}$
        $\mathbf{x_{k+1}} := \mathbf{x_k} + \alpha_\mathbf{k} \mathbf{p_k}$
        $\mathbf{r_{k+1}} := \mathbf{r_k} - \alpha_\mathbf{k} \mathbf{A} \mathbf{p_k}$
        if r_{k+1} is sufficiently small **then** $(\|r_{k+1}\| < \varepsilon)$
            return The result is $\mathbf{x_{k+1}}$ exit loop
        end if
        $\beta_k := \dfrac{\mathbf{r_{k+1}}^\top \mathbf{r_{k+1}}}{\mathbf{r_k}^\top \mathbf{r_k}}$
        $\mathbf{p_{k+1}} := \mathbf{r_{k+1}} + \beta_\mathbf{k} \mathbf{p_k}$
        $k := k + 1$
    until
end procedure

(a) Programa el algoritmo descrito anteriormente. Luego, verifique que el código funciona, tomando como datos:

$$A = \begin{pmatrix} 10 & 4 & 5 \\ 4 & 6 & 1 \\ 5 & 1 & 4 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 4 \\ 5 \end{pmatrix}.$$

(b) En modelos de Regresión, un modelo que se resuelve el siguiente problema de optimización convexa sin restricciones (Least-squares linear regression, with ℓ_2-regularization or Tikhonov regularization):

$$\min_{\mathbf{x}\in\mathbb{R}^n} \left\{ \frac{1}{2}\|A\mathbf{x} - b\|^2 + \frac{\lambda}{2}\|\mathbf{x}\|^2 \right\},$$

donde $\mathbf{x} \in \mathbb{R}^n$ es la variable, $A \in \mathbb{R}^{m\times n}$ y $b \in \mathbb{R}^m$ representan los datos de entrada y salida, $\lambda > 0$ es el parámetro de regularización. La solución de este problema de optimización, puede ser encontrada, resolviendo el siguiente sistema lineal:

$$(A^\top A + \lambda I)x = A^\top b. \tag{0.1}$$

(i) Encuentre la solución del problema de optimización, usando el algoritmo CGM, tomando las siguientes bases de datos[1]: Forest Fires, Concrete Compressive Strength, Wine Quality, con $\lambda = 2^{-1}, 2^1$ (En dichas bases, la última columna representa el vector b, y lo restante la matriz A).

(ii) Calcule $f(x^*) = Ax^*$, donde x^* denota la solución encontrada en el item (i). Luego, compare estos valores con el vector b, mediante el cálculo del error cuadrático medio (MSE):

$$\text{MSE} = \frac{\sum_{i=1}^m ((f(x^*))_i - b_i)^2}{m},$$

donde m denota el número de datos.

(iii) Use el método de Gauss-Seidel, para encontrar la solución del sistema (0.1).

(iv) Calcule $f(\tilde{x}) = A\tilde{x}$, donde $\tilde{x}$ denota la solución encontrada en el item (iii). Luego, compare estos valores con el vector b, mediante el cálculo de MSE.

Integración de funciones:

Corriente arriba de una presa, el agua ejerce una presión $p(z) = pg(D - z)$ medida en N/m^2 y ejercida a una elevación z metros por encima del fondo fluvial. Si se omite la presión atmosférica, la fuerza puede ser determinada al multiplicar la presión por el área de la cara de la presa. Esta área se obtiene integrando la función $w(z)$ que proporciona el ancho del río a una altura z del lecho y toma valores según la figura

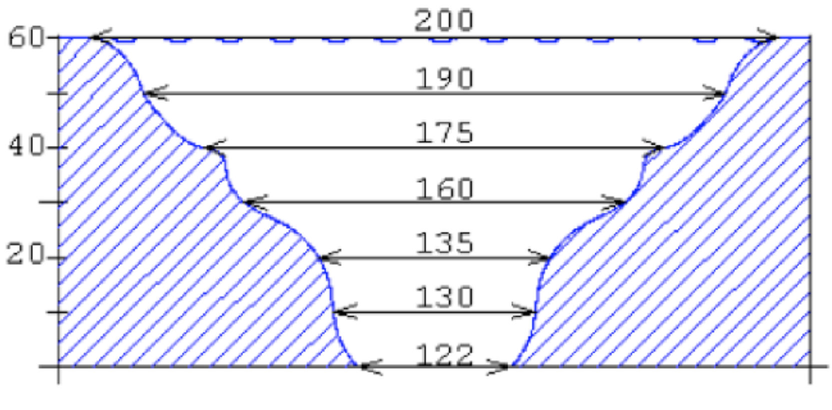

Fig. 1: Ancho del rio, w, respecto a altura z.

[1] http://archive.ics.uci.edu/ml/datasets.html

Debido a que la presión y el área varián con la elevación, la fuerza total se obtiene al calcular

$$f = \rho g \int_0^D w(z)(D-z)dz,$$

donde $\rho = 10^3 kg/m^3$ es la densidad del agua, $g = 9.8m/s^2$ es la aceleración debido a la gravedad y $D = 60$ es la elevación en metros de la superficie del agua por encima del fondo.

(a) Use la formula del trapecio compuesta (para 6 sub-intervalos) para calcular la fuerza total ejercida por el agua en la cara de la presa. **Hint.** usar los datos proporcionados en la figura, por ejemplo: $w(0) = 122$, $w(10) = 130$, $w(20) = 135$.

(b) Repita lo mismo del item (a), teniendo en cuenta la formula Simpson 1/3 compuesta (para 6 sub-intervalos).

(c) Usando interpolación de Lagrange, dibuje las curvas laterales de la Fig.1.

(d) Use interpolación por Spline cúbico, para obtener las curvas laterales de la Fig.1. Cuales de las dos curvas fue la más adecuada?.

Comandos: Comandos del Spline Toolbox en Matlab (cualquiera de los siguientes comandos puede usar para la interpolación con Spline):

- **spline**: Interpolación spline cúbica con condiciones frontera sujeta.
- **csape**: Interpolación spline cúbica con varias condiciones de frontera: completa, not-a-knot, periodic, second, variational.
- **csapi:** Interpolación spline cúbica con condiciones frontera not-a-knot.
- **cpapi:** Interpolación spline cúbica general (lineal, cuadrática, cúbica, cuártica, etc).
- **csaps:** Interpolación spline cúbica suavizante.

Integración Doble:

Considere una función $f : \mathbb{R}^2 \to \mathbb{R}$, integrable en una región rectangular $R \subset \mathbb{R}^2$, la cual es definida por:

$$R = \{(x, y) \in \mathbb{R}^2 : a \leq x \leq b, c \leq y \leq d\}, \text{ con } a, b, c, d \in \mathbb{R}.$$

El propósito del siguiente problema, es aproximar el valor de la siguiente integral doble:

$$I = \int\int_R f(x, y)dA = \int_a^b \left[\int_c^d f(x, y)dy \right] dx.$$

1. Describa la forma del método de Trapecio compuesto para el calculo de la integral doble I.
2. Use la formula encontrada en el item anterior (con 8 sub-intervalos en cada eje), para calcular de forma aproximada, el valor de las siguientes dos integrales dobles:

$$\int_0^1 \int_0^1 \frac{1}{1 - 0.5xy} dydx, \quad \int_{-1}^1 \int_{-1}^1 \cos(\frac{\pi x}{2}) \sin(\frac{\pi x}{2}) dydx.$$

   Compare sus resultados, con el valor exacto (**Hint.** Use Wolframalpha, para el calculo del valor exacto).