Curso de Optimización, 2021

Instituto de Matemática y Estadística (IMERL)

**Práctico 0: Repaso de métodos iterativos**

Ejercicio: Resolución iterativa de un sistema de ecuaciones y estudio de la velocidad de convergencia

La idea es resolver un sistema de ecuaciones  usando el método iterativo de Jacobi. Para eso:

**a)** Genere los datos, definiendo una matriz aleatoria *A* de tamaño *n=20*, que sea diagonal dominante, y un vector *b* tal que *x*=(1,2,…,*n*) sea la solución. Eso se puede hacer de la siguiente forma:

n=10;

A0=randn(n);

rho=1.0;

d0=sum(abs(A0'));

A=A0+rho\*diag(d0);

xstar=(1:n)';

b=A\*xstar;

**b)** Haga un programa que itere a partir de un  arbitrario, generando una sucesión  con la regla de Jacobi: , siendo *D* la diagonal de *A,* y *E=A-D*.

Construya dos vectores Rx y Ex para ir guardando la evolución del residuo y del error en x (Obs: acá conocemos el error exacto porque sabemos que la solución es xstar).

Un código Matlab/Octave puede ser:

d=diag(A); D=diag(d);

invD=diag(1 ./d);

E=A-D;

x=randn(n,1);

Ex=norm(x-xstar); Rx=norm(A\*x-b);

iter=0;

Niter=20;

while iter<Niter,

iter=iter+1;

x=invD\*(b-E\*x);

Rx=[Rx; norm(A\*x-b)];

Ex=[Ex; norm(x-xstar)];

end

figure(1),semilogy(Rx), title('Residuo')

figure(2),semilogy(Ex),title('Error en x')

**c)** Pruebe otras condiciones en el while, de forma de parar cuando “ya convergió lo suficiente”.

**d)** Estudie la velocidad de convergencia, variando n (por ej. 100, 1000) y el parámetro rho

(por ej. 2, 1.1, 0.8). Observe la cantidad de iteraciones y el cociente .