EJERCICIOS DE LABORATORIO RESUELTOS

Prof. Dr. Fredy Vides
Scientific Computing Innovation Center, UNAH &
Centre for Analysis of Data-Driven Systems
E-mail: fredy.vides@unah.edu.hn

ÍNDICE

1. Ejercicios Resueltos 1
Ejercicio 1 1
Ejercicio 2 2

1. EJERCICIOS RESUELTOS

Ejercicio 1. Desarrollar un programa Octave llamado Arnoldi.m (**puede utilizar el comando** `krylov` o desarrollar su programa sin utilizar el comando `krylov`), que para cualquier matriz $A \in \mathbb{C}^{n\times n}$, cualquier vector $v \in \mathbb{C}^n$, y cualquier entero positivo $k \leq n$, calcule una base ortonormal de $\{u_1, \ldots, u_k\}$ de $\mathcal{K}_k(A, v)$, y la matriz de Hessenberg H_k correspondiente.

Solución. El siguiente programa de Octave que ha sido llamado `Arnoldi.m` y puede aplicarse para resolver el problema:

```
function [U,V,H0,H1]=Arnoldi(A,v,k)
  [V, H, N] = krylov (A, v, k+1);
  U=V(:,1:k);
  H0=H(1:k,1:k);
  H1=H(:,1:k);
endfunction
```

Para ejecutar el programa `Arnoldi.m` basta escribir la siguiente secuencia de comandos:

```
>> A=speye(5000)+sprandn (5000,5000,.0001);
>> A=(A*A');
>> v=randn(5000,1);
>> tic,[U,H0,H1]=Arnoldi(A,v,50);toc
Elapsed time is 0.275203 seconds.
>> norm(A*U-V*H1)
ans =    7.0702e-14
```

Ejercicio 2. Desarrollar un programa Octave que para una matriz $A \in \mathbb{C}^{n\times n}$ y un vector $v \in \mathbb{C}^n$, permita calcular el polinomio p_k determinado por el problema

$$p_k = \operatorname{argmin}_{p\in\mathcal{P}_k, p(0)=1} \|p(A)v\|_2 \tag{1.1}$$

para el subespacio $\mathcal{K}_k(A, b)$, resolviendo directamente el problema de mínimos cuadrados (1.1). Este nuevo programa puede también utilizar el comando `krylov`, en caso de ser necesario.

Desarrollar además un programa Octave que para dos enteros positivos k, n tales que $k \leq n$ genere una matriz $A \in \mathbb{C}^{n\times n}$, un vector $v \in \mathbb{C}^n$, y calcule el polinomio p_k determinado por (1.1) para el subespacio $\mathcal{K}_k(A, v)$, produciendo una salida gráfica que permite verificar aproximadamente que el polinomio solución p_k cumple (1.1).

Solución. El programa (script) Octave que hemos llamado `Lab7b.m` cuyo código se muestra a continuación, puede aplicarse para resolver el problema.

```
% Definir valores de n,k:
n=100;
k=10;
% Generar matriz de prueba A en C^nxn de rango r:
A=randn(n)+i*randn(n);
% Generar vector de prueba y en C^n:
v=randn(n,1)+i*randn(n,1);
% Calcular proyector Q y el polinomio minimizador p
% que cumplen las restricciones del problema:
tic
K=A*v;
  for j=1:(k-1)
      K=[K A*K(:,j)];
  end
  Kp=pinv(K);
  p=-Kp*v;
  p=[fliplr(p.') 1];
  Q=eye(n)-K*Kp;
  q=[-2 3 0 0];
  for k=1:5, Q=polyvalm(q,Q);end
% Generar polinomios para verificar aproximadamente
% la identidad: ||Qv||=min { ||p(A)v|| : p en P_m^1 }
ptest=[p(1:k)+1e-4*randn(10*n,k) ones(10*n,1)];
% Verificar aproximadamente la identidad:
% ||Qv||=min { ||p(A)v|| : p en P_m^1 }
for j=1:(10*n)
   test(j)=norm(polyvalm(ptest(j,:),A)*v);
end
toc
L=1:(10*n);
NQv=norm(Q*v);
NpAv=norm(polyvalm(p,A)*v);
plot(L,NQv*ones(1,10*n),'k.',L,...
NpAv*ones(1,10*n),'r.',...
```

```
'markersize',12,L,test,'b.','markersize',12);
grid on
axis tight
disp('Valor de ||p(A)v||:');
disp(NpAv);
disp('Valor de ||Qv||:');
disp(NQv);
disp('Valor del error ||Q^2-Q||:');
disp(norm(Q^2-Q));
disp('Valor del error ||Q*-Q||:');
disp(norm(Q'-Q));
```

El código de `Lab7b.m` aplica las ideas implementadas en la solución del ejercicio para el lector 3, utilizando el estándar utilizado en Octave para la definición del comando `pinv`. Además se aplican **pricipios de topolgía matricial** para "refinar" el proyector (aproximado) `Q` calculado por el programa. Para el desarrollo de `Lab7b.m` no se ha utilizado el comando `krylov`.

Para ejecutar el programa `Lab7b.m` basta ejecutar la siguiente secuencia de comandos.

```
>> Lab7b
Elapsed time is 6.34053 seconds.
Valor de ||p(A)v||:
 11.737
Valor de ||Qv||:
 11.737
Valor del error ||Q^2-Q||:
   1.4881e-15
Valor del error ||Q*-Q||:
   1.7188e-15
```

La salida gráfica de verificación aproximada de la solución del problema de mínimos cuadrados se muestra en la figura 1.1.

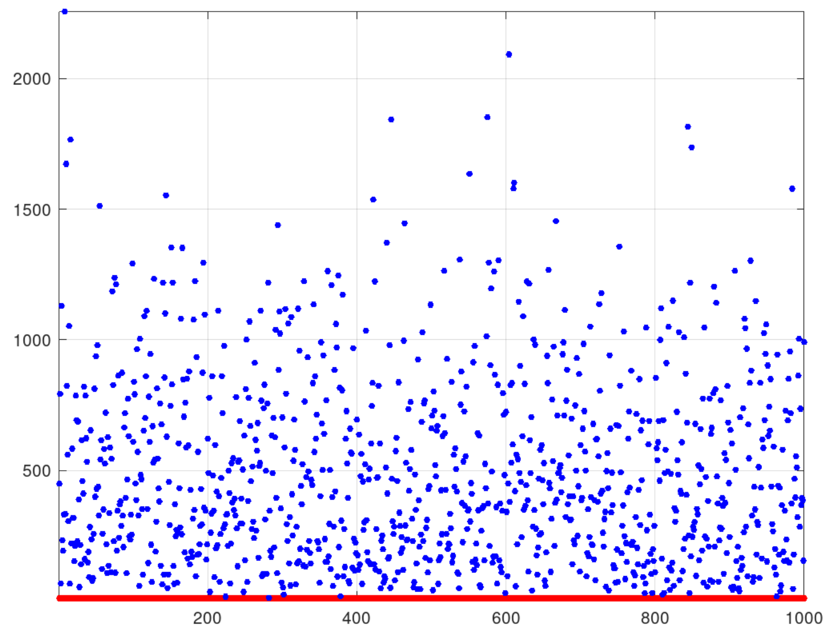

FIGURA 1.1. Salida gráfica del programa.