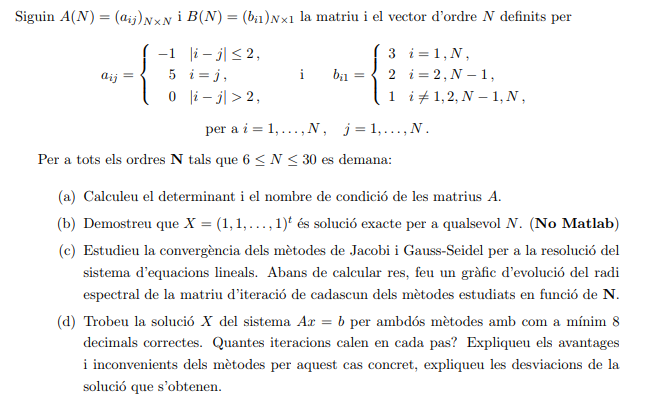
Pràctica 2 – Enunciat B  
Computació Numèrica

**David Moreno Borràs**

**2017-18 Q2**

1 | Àlgebra lineal numèrica: mètodes iteratius

1. Per començar s’inicialitzen la matriu A i el vector b amb la funció initMatriu.m (a l’annexe) segons la definició de l’enunciat.

S’inicialitzen per valors entre 6 i 30 i el resultat dels seus determinants i nombres de condició és el de la següent taula:

1. -----------------
2. Per estudiar la convergència hem de veure el radi espectral de la matriu B del sistema equivalent x = Bx + C. Els dos mètodes seran convergents si i només si el radi espectral es menor que 1.

En els dos casos calculem les matrius auxiliars necessàries (diagonal, lower triangular i upper triangular) i fem un for de 6 a 30 on calculem rho de B:

aux = inv(D);

Bj = -aux \*(L+U);

rhoJ = max(abs(eig(Bj)));

Si aquest rho es menor que 1, és convergent. Guardem els resultats a una matriu i obtenim els següents gràfics on podem veure que els dos mètodes convergeixen sempre per valors d’N entre 6 i 30.

1. Per trobar la solució amb 8 decimals correctes agafem un valor de tolerància tol = 0.000000005. Fem servir la funció “itera.m” (als annexes) a la que li passem tol, el nombre màxim d’iteracions, i les matrius necessàries.

Per cada valor d’N entre 6 i 30 es guarda a una taula el nombre d’iteracions que ha necessitat el mètode per obtenir un resultat amb 8 decimals correctes. Aquests són els resultats:

Com podem veure, el nombre d’iteracions necessàries als dos mètodes augmenta amb el tamany de la matriu i el mètode de Jacobi necessita moltes més que el de Gauss-Seidel ja que aquest té una velocitat de convergència (*-log(rho)*) major.

**Annex**

Codis Matlab Exercici 1: Àlgebra lineal numèrica: mètodes iteratius

%% Inicialització matrius

A = zeros(n,n);

b = zeros(n,1);

for c = 1:n

for r = 1:n

if r == c

A(r,c) = 5;

elseif abs(r-c) <= 2

A(r,c) = -1;

elseif r == 1 || r == n

b(c,:) = 3;

elseif r == 2 || r == n-1

b(c,:) = 2;

else %abs(r-c) > 2

A(r,c) = 0;

b(c,:) = 1;

end

end

end

% Bucle de càlcul d'iteracions

function [ x, residu, iteracions ] = itera( A, b, B, c, kMax, tol)

k = 0;

x = zeros(size(b));

residu = norm(b-A\*x);

while(k < kMax && residu >= tol)

aux = B\*x+c;

x = aux;

k = k+1;

residu = norm(b-A\*x);

end

iteracions = k;

end

%% Estudi convergència Jacobi

totalConv = zeros(30-6,2);

i = 1;

for n=6:30

[A, b] = initMatriu(n);

D = diag(diag(A));

L = tril(A,-1);

U = triu(A,1);

aux = inv(D);

Bj = -aux \*(L+U);

rhoJ = max(abs(eig(Bj)));

totalConv(i,:) = [n, rhoJ];

if(rhoJ < 1)

fprintf('Mètode de Jacobi Convergent\n ')

else

fprintf('Mètode de Jacobi divergent\n ')

end

i = i + 1;

end

filename = 'JacobiEstudiConvergencia.xlsx';

xlswrite(filename,totalConv);

%% Mètode de Gauss-Seidel

i = 1;

for n=6:30

[A, b] = initMatriu(n);

D = diag(diag(A));

L = tril(A,-1);

U = triu(A,1);

aux=inv(L+D);

Bs=-aux\*(U);

cs = aux\*b;

rhoS = max(abs(eig(Bs)));

if(rhoS < 1)

%fprintf('Mètode de Gauss-Seidel Convergent\n ');

kMax = 100; tol = 0.000000005;

[ xS, residuS, it ] = itera( A, b, Bs, cs, kMax, tol);

iteracionsGauss(i,:) = [n, it];

else

fprintf('Mètode de Gauss-Seidel divergent\n ');

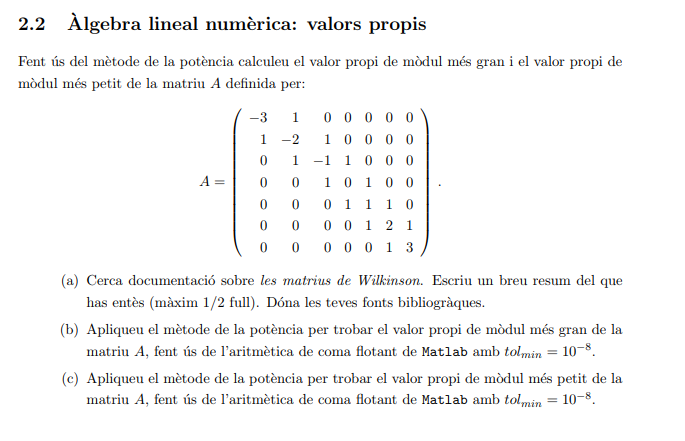
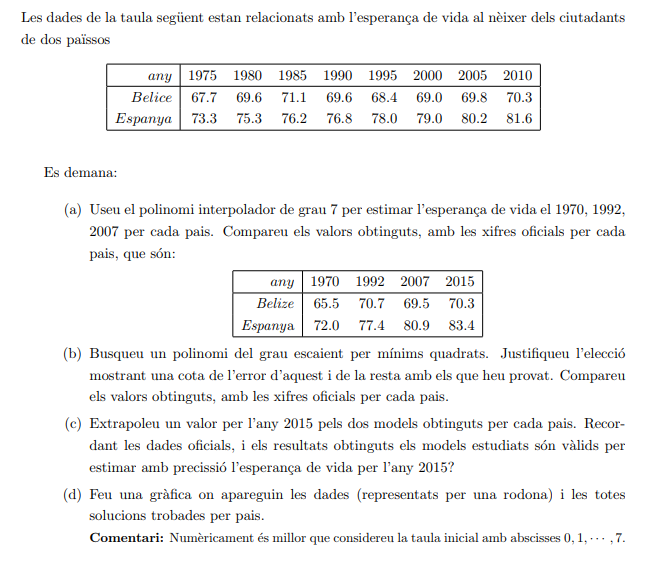
end

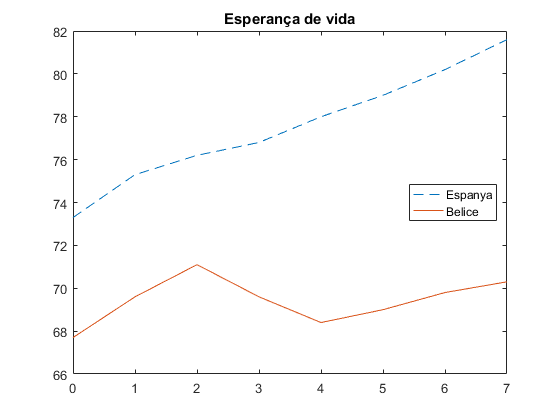
i = i + 1;

end

filename = 'GaussSeidelIteracions.xlsx';

xlswrite(filename,iteracionsGauss);

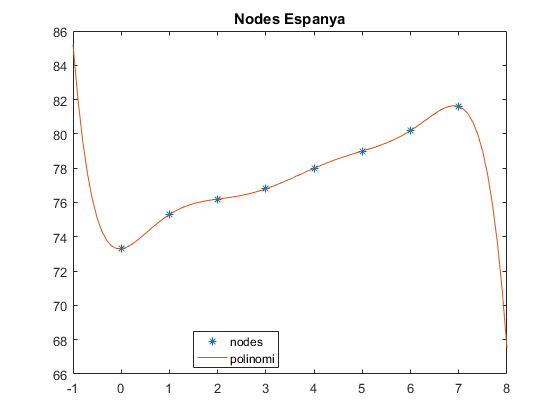
1. | Àlgebra lineal numèrica: valors propis
2. Les matrius de Wilkinson (anomenades així pel matemàtic angles James H. Wilkinson) són unes matrius simètriques, tridiagonals d’ordre N amb parells de valors propis casi iguals.
3. | Expressions recurrents
4. | Aproximació de dades
5. Per començar guardem les dades a Matlab. Aquí podem veure el plot de la progressió de l’esperança de vida dels dos països:

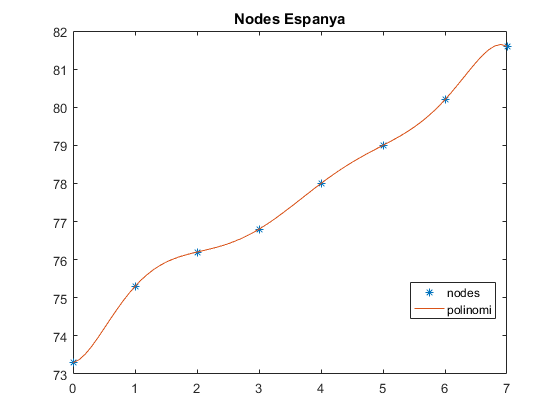


Com es comenta a l’enunciat, s’ha considerat en tot moment la taula d’abscisses 0:7 enlloc dels anys, ja que aquesta donava problemes.

Una vegada tenim les dades, procedim a buscar el polinomi interpolador.

Per obtenir el polinomi de grau 7 hem de cridar la funció polyfit. El resultat transposat són els coeficients. Els quals els fem servir per obtenir el polyval. El resultat, si fem un plot de 0 a 7, és la imatge de l’esquerra:



**Polinomi Interpolador Fenomen de Runge**

Com podem observar si fem el plot amb un interval més ampli, la interpolació resultat oscil·la molt als extrems de l’interval, així que podem observar el Fenomen de Runge.

Per obtenir els valors que demana l’enunciat: 1970, 1992 i 2007 hem de veure el valor que pren el polinomi en aquests punts, però com hem comentat abans la taula d’abscisses no són els anys, és 0:7.

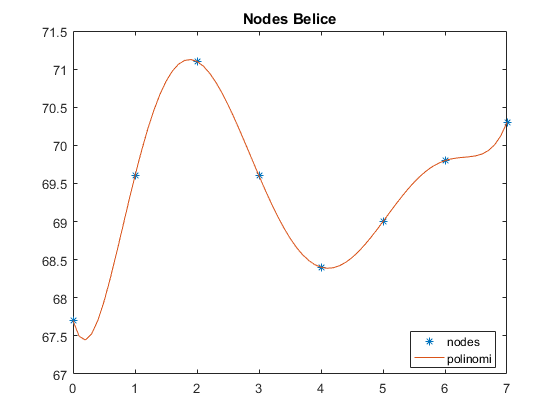
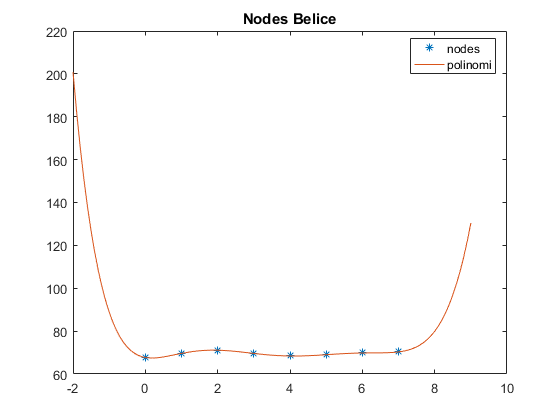
L’equivalència de “passos” és *1 pas = 5 anys*, per tant 1970 correspondria a -1, tenint en compte que l’inicial es 0. Com està “fora” hem d’ampliar el rang: t = -2:0.1:9 i tenint en compte que ara cada pas són 10 elements de la matriu i a Matlab es comença a indexar a 1, podem veure que v(21) = 73.3000, el primer valor, per tant, **v(11) = 85.200** equival a l’estimació per l’any 1970. De fet, per comprovar, podem fer “data cursor” al plot i veure que efectivament el valor corresponent a -1 és **85,2.**

Per l’any 2007 tenim que v(91) = 81.0, l’any 2010. I v(81) = 80.2, lany 2005, per tant, com entre 5 anys hi ha 10 indexs, cada any són 2 indexs. Per tant, si volem l’any 2007 hem de buscar v(81 + 2 + 2) = **v(85) = 80.9803**.

Finalment, per l’any 1992 hem d’observar quin index correspon a 1990 i sumar 4, com abans. Li correspon v(51), per tant l’estimació per 1992 és v(51 + 4) = **v(55) = 77.2448.**

Com podem observar, els valors de l’any 2007 i 1992, al estar dins l’interval, tenen una estimació força aproximada, concretament l’error absolut en el cas de l’any 2007 és **0.083** i l’any 1992 és **0.1552**.L’any 1970, al estar fora i degut al fenomen de Runge té un error absolut de **13.2000**.

Ara fem el mateix amb Belice:

****

**Polinomi Interpolador Fenomen de Runge**

De nou podem observar el fenomen de Runge quan estudiem el polinomi fora de l’interval.

Seguint els mateixos passos que amb Espanya, obtenim que l’any 1970 té una estimació de v(11) = 88.90, l’any 2007 v(85) = 69.8506 i l’any 1992 v(55) = 68.9179

Si calculem els valors absoluts obtenim:

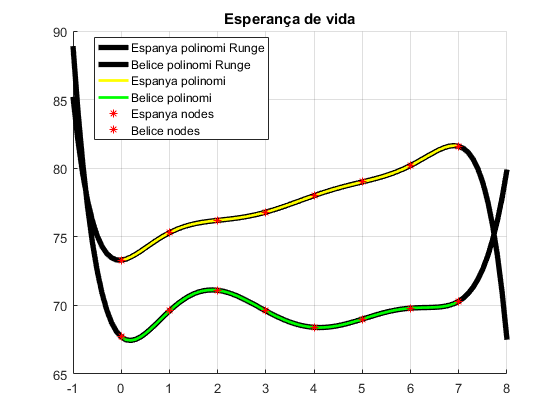
1970 -> 23.40

1992 -> 1.7821

2007 -> 0.3506

Així que de nou veiem que els dos que estan dins dels intervals fan estimacions bastants correctes però la de 1970 és molt diferent.

La següent imatge és un plot amb tots els elements vists:



Els codis usats es troben a l’annex.

b)