1 Metodes numerics aplicats als conceptes basics de l'algebra lineal (continuacio)

Amelia Martínez Sequera

26/9/2021

Nota: He vist que que a la PAC1 no vaig copiar bé la matriu L, canviant el 10 per un 0. Per tant, quan faig referència a que els resultats coincideixen amb la pràctica anterior, em refereixo als resultats correctes, no els que jo vaig obtenir.

Prèviament, haviem calculat la suma deles potències dels valors propis mitjançant el càlcul de la traça de la potència de la matriu L i havíem obtingut els coeficients del polinomi característic.

```
##      [,1] [,2] [,3] [,4]
## [1,]   10  -35   50  -24
## [2,]    1    0    0    0
## [3,]    0    1    0    0
## [4,]    0    0    1    0
```

Alternativa al càlcul dels coeficients del polinomi característic.

Tenint en compte l´expressió del polinomi característic introduïda en la pràctica anterior:

$$P(\Lambda) = \Lambda^4 + c_1\Lambda^3 + c_2\Lambda^2 + c_3\Lambda + c_4$$

existeix una variant del mètode de Leverrier, coneguda com el mètode de Leverrier-Faddeev, que també ens permet obtenir els coeficients del polinomi a partir del càlcul de la traça de certes matrius, de la següent manera:

$c_1 = -Tr(B_1), B_1 = L$

$c_2 = -1/2Tr(B_2), B_2 = L(B_1 + c_1 I)$

$c_3 = -1/3Tr(B_3), B_3 = L(B_2 + c_2 I)$

$c_4 = -1/4Tr(B_4), B_4 = L(B_3 + c_3 I)$

i, en general, $c_n = -1/nTr(B_n), B_n = L(B_{n-1} + c_{n-1} I)$

Calculem els coeficients del polinomi característic seguint les expressions anteriors i comprovem que, efectivament, s´obtenen els mateixos valors que resolent el sistema de la pràctica anterior.

```
#Ordre de la matriu L
(N = dim(L))
```

```
## [1] 4 4

#Alternativa en el calcul dels coeficients del polinomi caracteristic

I<-diag(rep(1,4)) #Matriu identitat de dimensio igual que L
B1 = L
c1 = -sum(diag(B1))


B2 = L%*%(B1 + c1*I)
c2 = -(1/2)*sum(diag(B2))


B3 =  L%*%(B2 + c2*I)
c3 = -(1/3)*sum(diag(B3))


B4 =  L%*%(B3 + c3*I)
c4 = -(1/4)*sum(diag(B4))

c1;c2;c3;c4

## [1] -10

## [1] 35

## [1] -50

## [1] 24
```

Tercera part de l´algorisme de Leverrier: Càlcul dels valors propis de L (I).

Leverrier va trobar diverses relacions molt útils per al càlcul dels valors propis de la matriu L. Definim, a partir de les s_n de la pràctica anterior,

$$\sigma = \frac{s_n}{s_{n-1}}, n \geq 2, \sigma_1 = 0$$

$$\delta = \frac{\sigma_n - \sigma_{n-1}}{\sigma_{n-1} - \sigma_{n-2}}, n \geq 3$$

Fent ús d´aquestes dues successions, es defineixen dues noves successions a_n i b_n:

$$a_n + b_n = \sigma_n + \delta_n \sigma_{n-2}$$

$$a_n b_n = \sigma_{n-1} \sigma_{n-2} \delta_n$$

amb $|a_n| \geq |b_n|$ i $n \geq 3$.

Es pot demostrar que les successions a_n i b_n convergeixen cap als dos valors propis de L amb major mòdul. Per tant, si els calculem, tindrem els valors de Λ_3 i Λ_4.

Calculem, per tant, els termes a_n i b_n amb $3 \leq n \leq 22$, i els presentem en una taula. Els últims termes seran els valors propis buscats: $\Lambda_3 = b_{22}$ i $\Lambda_4 = a_{22}$.

Per tant, hem de calcular s1, s2, ...,s22:

```
##  [1] 1.000000e+01 3.000000e+01 1.000000e+02 3.540000e+02 1.300000e+03
##  [6] 4.890000e+03 1.870000e+04 7.235400e+04 2.823400e+05 1.108650e+06
## [11] 4.373500e+06 1.731275e+07 6.871138e+07 2.732348e+08 1.088124e+09
## [16] 4.338080e+09 1.730914e+10 6.910716e+10 2.760407e+11 1.102999e+12
## [21] 4.408509e+12 1.762357e+13
```

Veiem que coincideix amb el resultats de la pràctica anterior.

Ara calculem sigma:

```
##  [1] 0.000000 3.000000 3.333333 3.540000 3.672316 3.761538 3.824131
3.869198
##  [9] 3.902203 3.926649 3.944888 3.958558 3.968830 3.976558 3.982375
3.986753
## [17] 3.990047 3.992524 3.994386 3.995786 3.996837 3.997626
```

i delta:

```
##  [1] 0.0000000 0.0000000 0.1111111 0.6200000 0.6402406 0.6743086
0.7015351
##  [8] 0.7200070 0.7323587 0.7406611 0.7461121 0.7494958 0.7514160
0.7523526
## [15] 0.7526697 0.7526257 0.7523935 0.7520814 0.7517531 0.7514433
0.7511682
## [22] 0.7509334
```

A continuacio calculem les successions a_n i b_n:

```r
#Resolem el següent sistema d'equacions:
#a[i] + b[i]  = sigma[i] + delta[i]*sigma[i-2]
#a[i]*b[i] = sigma[i-1]*sigma[i-2]*delta[i]

#Definim yn = an + bn, i zn = an*bn


yn = c(0, 0, 0)
zn = c(0, 0, 0)

for (i in 3:n){
  yn[i] = sigma[i] + (delta[i]*sigma[i-2])
}



for (i in 3:n){
  zn[i] = sigma[i-1]*sigma[i-2]*delta[i]
  }
```

```r
print(zn)

##  [1]  0.000000  0.000000  0.000000  6.200000  7.554839  8.766012
9.690697
##  [8] 10.357024 10.836213 11.182794 11.432363 11.609831 11.734164
11.820096
## [15] 11.878847 11.918692 11.945560 11.963607 11.975696 11.983780
11.989180
## [22] 11.992784

print(yn)

##  [1] 0.000000 0.000000 3.333333 5.400000 5.806452 6.148591 6.400390
6.577532
##  [9] 6.702839 6.792413 6.856369 6.901565 6.933082 6.954790 6.969593
6.979613
## [17] 6.986360 6.990887 6.993916 6.995941 6.997293 6.998195

library(pracma)

#Els valors de an i bn s'obtenen al resoldre l'equacio:
#    x^2 - yn[i]x + zn[i]=0 per a cada valor de i.

for (i in 3:n){
  lambda = (polyroot(c(zn[i], -yn[i], 1)))
}

print(round(lambda,1))

## [1] 3+0i 4+0i

#La funcio Re torna valors complexos i ens quedem amb la part real

for (i in 3:n){
  lambda = Re(polyroot(c(zn[i], -yn[i], 1)))
}

## El valor de lambda 4 és 4

## El valor de lambda 3 és 3
```

que coincideix amb els resultats de Λ_4 i Λ_3 obtinguts a la pràctica anterior:

```r
lambdaPAC1 <-eigen(L)

lambdaPAC1$values

## [1] 4 3 2 1
```

Quarta part de l´algorisme de Leverrier: Càlcul dels valors propis de L (II).

A partir de l´anterior, podem calcular un valor aproximat de les dues arrels de $P(\Lambda)$ restants. Una vegada es tenen dues arrels aproximades del polinomi característic, es poden calcular les altres dues de la manera següent:

$$\frac{P_\Lambda}{(\Lambda - a_{22})(\Lambda - b_{22})} = 0$$

$\Lambda_3 = b_{22}$ i $\Lambda_4 = a_{22}$.

Si es fa aquesta divisió, i es calculen les arrels del polinomi de grau dos resultant, s´obtenen les dues arrels que falten del polinomi característic.És a dir, ja tindrem els quatre valors propis de la matriu L.

```
#Calcul dels dos valors propis lambda_1 i lambda_2 de la matriu L
num = c(1, c1, c2, c3, c4)
den = c(1, -(lambda[1] + lambda[2]), lambda[1]*lambda[2]) #(x-
lambda3)*(x-lambda4)

div = deconv(num, den)
print(div[[1]])

## [1]  1.000000 -3.001805  1.999999

#Calculem les arrels del polinomi obtingut
lambda = (polyroot(c(rev(div[[1]]))))

print(round(lambda,1))

## [1] 1+0i 2+0i

## El valor de lambda 2 és 2

## El valor de lambda 1 és 1
```

Veiem que s´obtenen els mateixos resultats que a la pràctica anterior.