Matrius-guia19-v0.2: Matriu inversa

f.u., a.g.

Març 30 2021

Definició i notacions

Calcular la inversa d'una matriu quadrada en R és ben fàcil, fem servir la funció **solve()**. Més endavant entendreu perquè calcular la inversa és el mateix que "resoldre".

```
A <- matrix(c(2, 0, 3, 4, 7, 5,  5, 4, 9), ncol = 3)
A
```

```
     [,1] [,2] [,3]
[1,]    2    4    5
[2,]    0    7    4
[3,]    3    5    9
```

```
solve(A)
```

```
           [,1]        [,2]       [,3]
[1,]  1.4827586 -0.37931034 -0.6551724
[2,]  0.4137931  0.10344828 -0.2758621
[3,] -0.7241379  0.06896552  0.4827586
```

R fa sempre els càlculs amb la versió decimal (se'n diu de coma flotant, vegeu el final de l'anterior guia) dels nombres implicats. Entre altres coses, això té l'aventatge de ser molt més eficient. I no importa massa la dimensió de la matriu.

```
A <- matrix(sample(-24:24), nrow = 7) # són els nombres des de -24 fins a 24 reordenats aleatoriament
iA <- solve(A)
round(iA,4)
```

```
        [,1]    [,2]    [,3]    [,4]    [,5]    [,6]    [,7]
[1,] -0.0006  0.0909 -0.1474 -0.0047  0.0855  0.0257 -0.0819
[2,] -0.0205 -0.0224  0.0473 -0.0237 -0.0263 -0.0039  0.0012
[3,] -0.0145  0.0558 -0.0634 -0.0083  0.0569  0.0031 -0.0081
[4,]  0.0130  0.0161  0.0146 -0.0107 -0.0234  0.0105  0.0098
[5,] -0.0091 -0.0180  0.0261 -0.0064 -0.0340  0.0104  0.0218
[6,] -0.0096 -0.0101  0.0405  0.0175 -0.0276 -0.0157  0.0236
[7,] -0.0182  0.0604 -0.0846 -0.0436  0.0489 -0.0072 -0.0322
```

```
A %*% iA
```

```
             [,1]          [,2]          [,3]          [,4]          [,5]
[1,] 1.000000e+00 -4.163336e-16  3.885781e-16 -1.110223e-16 -8.326673e-17
[2,] 1.804112e-16  1.000000e+00 -1.665335e-16 -2.220446e-16 -1.387779e-16
[3,] 0.000000e+00 -1.110223e-16  1.000000e+00  0.000000e+00  1.110223e-16
[4,] 5.551115e-17  0.000000e+00  2.220446e-16  1.000000e+00 -2.220446e-16
[5,] 0.000000e+00 -3.330669e-16  0.000000e+00  0.000000e+00  1.000000e+00
[6,] 1.110223e-16  6.661338e-16  4.440892e-16  0.000000e+00  2.220446e-16
```

```
[7,] 0.000000e+00  8.326673e-17 -1.665335e-16 -2.775558e-17  2.775558e-17
              [,6]          [,7]
[1,]  6.938894e-17  1.665335e-16
[2,]  1.249001e-16 -5.551115e-17
[3,] -1.387779e-17  0.000000e+00
[4,]  2.775558e-17  1.665335e-16
[5,] -1.387779e-17  0.000000e+00
[6,]  1.000000e+00  1.110223e-16
[7,]  6.938894e-18  1.000000e+00
```

Com veieu, això té l'inconvenient que de vegades no s'entén a la primera el què hem obtingut, aquesta matriu identitat no ho sembla gaire degut als petitissims errors que s'acumulen en els càlculs amb decimals.

Podem demanar el resultat arrodonit i ho veiem clar:

```
round(A %*% iA,4)
```

```
     [,1] [,2] [,3] [,4] [,5] [,6] [,7]
[1,]    1    0    0    0    0    0    0
[2,]    0    1    0    0    0    0    0
[3,]    0    0    1    0    0    0    0
[4,]    0    0    0    1    0    0    0
[5,]    0    0    0    0    1    0    0
[6,]    0    0    0    0    0    1    0
[7,]    0    0    0    0    0    0    1
```

Resoldre un sistema d'equacions lineals és multiplicar per la inversa.

Com que un sistema d'equacions lineals es pot escriure matricialment com a $\mathbf{AX} = \mathbf{b}$, la solució serà $\mathbf{X} = \mathbf{A}^{-1}\mathbf{b}$. Per tant, per resoldre el sistema

$$\begin{aligned} x + 2y - z &= -5 \\ 2x - y + z &= 6 \\ x - y - 3z &= -3 \end{aligned}$$

farem:

```
A <- matrix(c(1,2,1,2,-1,-1,-1,1,-3), nrow = 3)
A
```

```
     [,1] [,2] [,3]
[1,]    1    2   -1
[2,]    2   -1    1
[3,]    1   -1   -3
```

```
b <- c(-5, 6, -3)
solve(A,b)
```

```
[1]  1 -2  2
```

Observa que la funció **solve()** que hem fet servir per calcular la inversa, si li donem un segon argument, ens torna el producte de la inversa per aquest segon argument.

```
solve(A) %*% b- solve(A,b)  # pràcticament zero, és el mateix
```

```
              [,1]
[1,] -2.220446e-16
[2,]  0.000000e+00
[3,]  0.000000e+00
```

Matrius ortogonals

La matriu de l'exemple de la guia:

```
P<- matrix(c(1,-1,1,1),nrow = 2) / sqrt(2)
P
```

```
           [,1]      [,2]
[1,]  0.7071068 0.7071068
[2,] -0.7071068 0.7071068
```

```
solve(P) # la inversa és la transposada
```

```
          [,1]       [,2]
[1,] 0.7071068 -0.7071068
[2,] 0.7071068  0.7071068
```

```
P %*% t(P)
```

```
     [,1] [,2]
[1,]    1    0
[2,]    0    1
```

```
t(P) %*% P
```

```
     [,1] [,2]
[1,]    1    0
[2,]    0    1
```

Per ortogonalitzar una matriu, R fa servir un mètode anomenat de Gram-Schmidt, disponible al paquet **pracma**. No hi entrem en aquest curs, però el podeu trobar força ben explicat a (https://ca.wikipedia.org/wiki/Proc%C3%A9s_d%27ortogonalitzaci%C3%B3_de_Gram-Schmidt).

L'apliquem a un exemple.

```
library(pracma) # potser us cal fer install.packages("pracma") si no el teniu instal·lat

A <- matrix(sample(-12:12), nrow = 5) # són els nombres des de -12 fins a 12 reordenats aleatoriament
A
```

```
     [,1] [,2] [,3] [,4] [,5]
[1,]   -1  -11    0    6    1
[2,]    9    5   12   -5   -2
[3,]    8    2   10   11   -6
[4,]   -3    3  -12    4   -9
[5,]   -8   -7  -10   -4    7
```

```
nA <- gramSchmidt(A)$Q
nA
```

```
            [,1]         [,2]        [,3]        [,4]        [,5]
[1,] -0.06757374 -0.873395298 -0.33957657 -0.09810634 -0.32814225
[2,]  0.60816364  0.009153488 -0.06571676 -0.77656943  0.15058030
[3,]  0.54058990 -0.196037198 -0.28126517  0.54904822  0.53737131
[4,] -0.20272121  0.386734861 -0.89483988 -0.08546182 -0.03602941
[5,] -0.54058990 -0.221590685  0.02281550 -0.28028091  0.76130295
```

```
round(nA %*% t(nA), 4)
```

```
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    0    0    0    0
```

```
[2,]    0    1    0    0    0
[3,]    0    0    1    0    0
[4,]    0    0    0    1    0
[5,]    0    0    0    0    1
```

Matrius de transferència de vots

Posem que en una enquesta s'ha demanat a l'entrevistat què va votar a les darreres eleccions i què vol votar ara. Disposarem les dades en una matriu $T = (t_{ij})$ on l'element i, j, fila i i columna j és el percentatge de votants del partit i que ara han decidit votar al partit j. Posem que tenim aquestes dades:

	P-A	P-B	P-C	P-D	P-E
P-A	0.5	0.2	0.1	0.1	0.1
P-B	0.2	0.6	0.05	0.05	0.1
P-C	0.05	0.15	0.4	0.2	0.2
P-D	0.05	0.05	0.05	0.8	0.05
P-E	0.15	0.2	0.2	0.15	0.3

Això vol dir que dels votants anteriors al partit A, el 20% es passen al B, el 10% al C, però el 50% resten fidels al partit A.

Si tenim les dades en un full Excel Transf-Vots.xlxs, podem llegir-les en R:

```r
library(readxl)
t.data <- read_excel("Transf-Vots.xlsx", col_names = TRUE)
```

```
New names:
* `` -> ...1
```

```r
T <- as.matrix(t.data[,-1]) # suprimim la primera columna que son els noms
T
```

```
      P-A  P-B  P-C  P-D  P-E
[1,] 0.50 0.20 0.10 0.10 0.10
[2,] 0.20 0.60 0.05 0.05 0.10
[3,] 0.05 0.15 0.40 0.20 0.20
[4,] 0.05 0.05 0.05 0.80 0.05
[5,] 0.15 0.20 0.20 0.15 0.30
```

Per suposat, també podem entrar les dades "a mà", en columnes:

```r
T <- matrix(c(0.5, 0.2, 0.05, 0.05, 0.15, 0.2, 0.6, 0.15, 0.05, 0.2, 0.1, 0.05, 0.4, 0.05, 0.2, 0.1, 0.0
T
```

```
     [,1] [,2] [,3] [,4] [,5]
[1,] 0.50 0.20 0.10 0.10 0.10
[2,] 0.20 0.60 0.05 0.05 0.10
[3,] 0.05 0.15 0.40 0.20 0.20
[4,] 0.05 0.05 0.05 0.80 0.05
[5,] 0.15 0.20 0.20 0.15 0.30
```

En qualsevol cas tenim la mateix matriu que en direm la "matriu de transferència de vots".

Aquesta matriu ens permet passar de la distribució de vots de les anteriors eleccions als resultats previstos per a la següent. Com ho hem de fer?

Posem que a les anteriors la distribució dels vots va ser A: 0.35, B: 0.25, C:0.20, D: 0.15, E: 0.05. Això ho podem representar en un vector:

```r
V <- c(0.35, 0.25, 0.20, 0.15, 0.05)
V
```

```
[1] 0.35 0.25 0.20 0.15 0.05
```

Lavors, per calcular quina seria la distribució de vots per les properes elecions, quin càlcul matricial hem de fer? Creus que hauria de ser $T \cdot V$? O bé $V \cdot T$? Raona la teva resposta.

Cadena de Markov pels usos de transport públic

Aquest tipus de model reben el nom de Cadenes de Markov. Ara veiem un altre exemple: els viatges a la ciutat de Barcelona es poden realitzar en cotxe privat (P), transport públic (M), bicicleta (B) o a peu (P). Posem que actualment les proporcions siguin, respectivament, 0.18, 0.38, 0.12, 0.32.

Podeu consultar dades reals del 2018 a [https://www.idescat.cat/serveis/biblioteca/docs/bib/pec/paae2018/a07322018.pdf]

Segons els estudis de l'autoritat metropolitana del transport, la gent que va ara en cotxe seguirà usant-lo en un 60% dels casos però canviarà a M, B, P en proporcions 25%, 5%, 10% respectivament. Els que van en transport públic passaran a C, B, P en proporcions 10%, 12%, 5%. Els de la bicicleta canviaran a C, M, P amb proporcions 2%, 5%, 5%. I els que van a peu canviaran a C, M, B en proporcions 2%, 10%, 5%.

Calcula la matriu de transició T i les proporcions previstes segons auqestes dades per l'any següent.

```
       [,1]   [,2]   [,3]   [,4]
[1,] 0.1644 0.3528 0.1582 0.3246
```

Resposta: el vector de les proporcions actuals és 0.18, 0.38, 0.12, 0.32, i el resultant d'aplicar la matriu de transició serà 0.1644, 0.3528, 0.1582, 0.3246.