Ecuaciones y sistemas lineales

Imanol

11/3/2021

R

Sistemas compatibles determinados.

Sistemas compatibles determinados con *solve()*

```
A = rbind(c(1,1,2),c(2,4,-3),c(3,6,-5))
b = c(9,1,0)
AB = cbind(A,b) # Unimos matriz A y b

# Comprobar si es compatible determinado
qr(A)$rank==qr(AB)$rank # Compatible
```

```
## [1] TRUE
```

```
qr(A)$rank==3 # Determinado
```

```
## [1] TRUE
```

```
# Resolver el sistema
solve(A,b)
```

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

```
# Comprobacion
solution = c(1,2,3)
A%*%solution == b # Confirmado, nos da el vector de terminos independientes.
```

```
##      [,1]
## [1,] TRUE
## [2,] TRUE
## [3,] TRUE
```

Sistemas compatibles determinados con *matlib*

```r
A = rbind(c(2,2),c(-1,1))
b = c(1,2)
AB = cbind(A,b) # Unimos matriz A y b

library(matlib)

# Una vez en forma de matriz se puede mostrar
showEqn(A, b)
```

```
##  2*x1 + 2*x2  =  1
## -1*x1 + 1*x2  =  2
```

```r
# Calcular rango de la matriz con matlib
R(A)
```

```
## [1] 2
```

```r
R(AB)
```

```
## [1] 2
```

```r
# Comprobar si el sistema es compatible con matlib
all.equal(R(A),R(AB))
```

```
## [1] TRUE
```

```r
# Resolver el sistema son matlib
Solve(A,b, fractions = TRUE) # En este cas la s es mayuscula.
```

```
## x1    =  -3/4
##   x2  =   5/4
```

```r
                              # Si fractions es TRUE nos lo muestra en fracciones
```

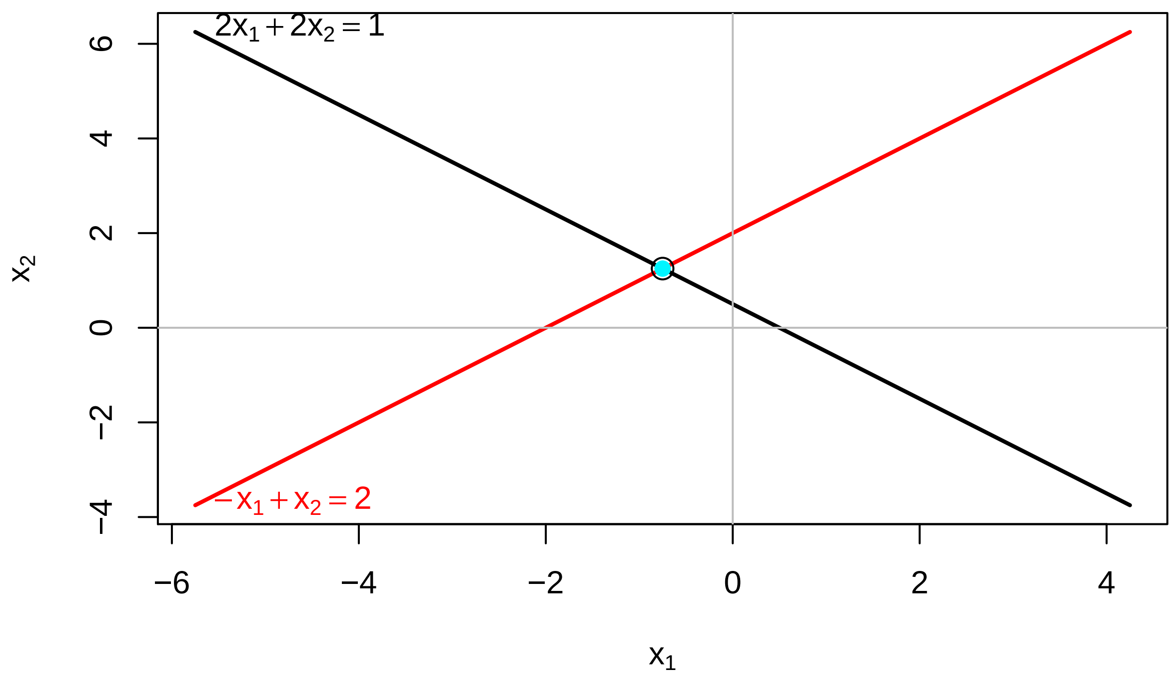
Representación de Sistemas.

Se hace uso de la libreria `matlib()` y mediante *plotEqn()* si tenemos 2 incognitas o *plotEqn3()* si tenemos 3 incognitas.

```r
library(matlib)

# Representacion con plotEqn()

## Dos Incognitas

### Dos ecuaciones
A = rbind(c(2,2),c(-1,1))
b = c(1,2)
AB = cbind(A,b) # Unimos matriz A y b
plotEqn(A,b)
```

```
## 2*x[1] + 2*x[2]  =  1
##  -x[1]   + x[2]  =  2
```

```
points(-3/4,5/4, col = "turquoise1", pch = 19) # Para dibujar puntos con ciertas caracteristicas
```

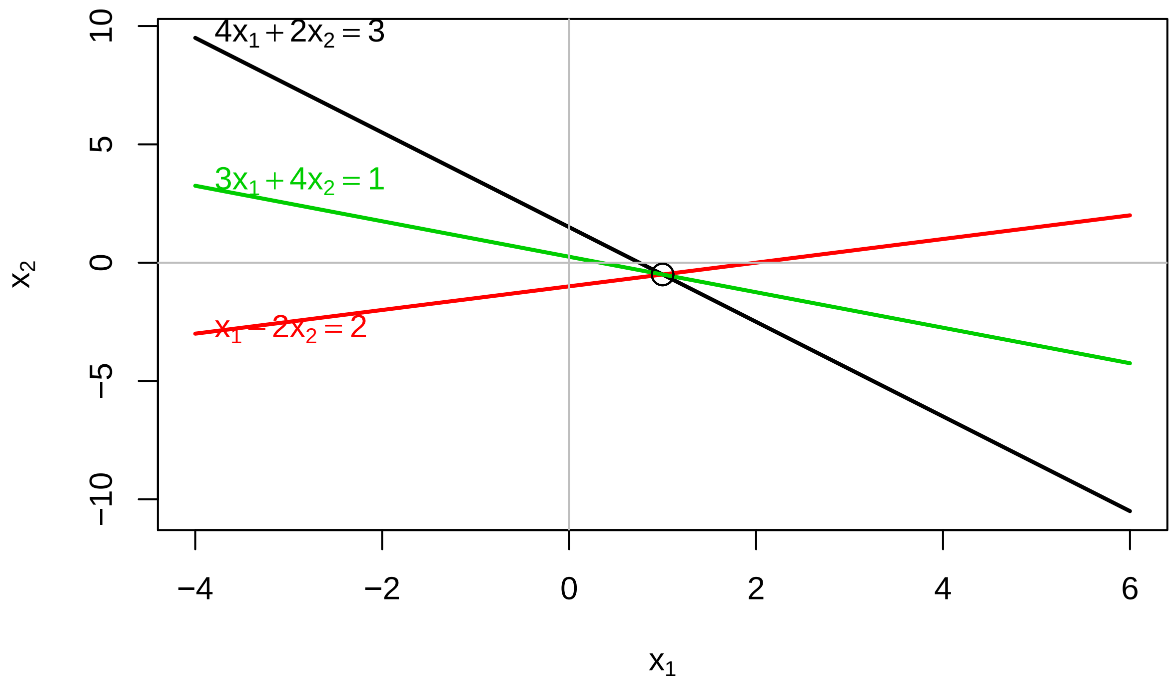



```
#### Tres ecuaciones
A = rbind(c(4,2),c(1,-2),c(3,4))
b = c(3,2,1)
showEqn(A,b)
```

```
## 4*x1 + 2*x2  =  3
## 1*x1 - 2*x2  =  2
## 3*x1 + 4*x2  =  1
```

```
plotEqn(A,b)
```

```
## 4*x[1] + 2*x[2]  =  3
##   x[1] - 2*x[2]  =  2
## 3*x[1] + 4*x[2]  =  1
```




```
## Tres Incognitas

A = rbind(c(1,1,2),c(2,4,-3),c(3,6,-5))
b = c(9,1,0)
showEqn(A,b)
```

```
## 1*x1 + 1*x2 + 2*x3  =  9
## 2*x1 + 4*x2 - 3*x3  =  1
## 3*x1 + 6*x2 - 5*x3  =  0
```

```
plotEqn3d(A,b, xlim = c(-3,3), ylim = c(0,6))
```

Tema 2 Ejercicio 1.

```
A = rbind(c(4,2),c(1,-2),c(3,4))
b = c(3,2,1)
showEqn(A,b)
```

Solución (a)

```
## 4*x1 + 2*x2  =  3
## 1*x1 - 2*x2  =  2
## 3*x1 + 4*x2  =  1
```

```r
AB = cbind(A,b)

all.equal(R(A),R(AB))
```

```
## [1] TRUE
```

```r
# Resolvemos
Solve(A,b,fractions = T)
```

Solución (b)

```
## x1    =     1
##   x2  =  -1/2
##    0  =     0
```

```r
#Comprobamos
s = c(1,-1/2)
A%*%s == b # Efectivamente es la correcta
```

```
##      [,1]
## [1,] TRUE
## [2,] TRUE
## [3,] TRUE
```

Metodo de GAUSS.

```r
library(matlib)
A = rbind(c(1,1,2),c(2,4,-3),c(3,6,-5))
b = c(9,1,0)
AB = cbind(A,b)

echelon(AB) # Escalona la matriz
```

```
##            b
## [1,] 1 0 0 1
## [2,] 0 1 0 2
## [3,] 0 0 1 3
```

```r
echelon(AB, verbose = T, fraction = T) # Verbose = True nos da los pasos que ha hecho para escalonarla
```

```
##
## Initial matrix:
##               b
## [1,]  1  1  2  9
## [2,]  2  4 -3  1
## [3,]  3  6 -5  0
```

```
##
## row: 1
##
##  exchange rows 1 and 3
##              b
## [1,]  3  6 -5  0
## [2,]  2  4 -3  1
## [3,]  1  1  2  9
##
##  multiply row 1 by 1/3
##                    b
## [1,]    1    2 -5/3    0
## [2,]    2    4   -3    1
## [3,]    1    1    2    9
##
##  multiply row 1 by 2 and subtract from row 2
##                    b
## [1,]    1    2 -5/3    0
## [2,]    0    0  1/3    1
## [3,]    1    1    2    9
##
##  subtract row 1 from row 3
##                    b
## [1,]    1    2 -5/3    0
## [2,]    0    0  1/3    1
## [3,]    0   -1 11/3    9
##
## row: 2
##
##  exchange rows 2 and 3
##                    b
## [1,]    1    2 -5/3    0
## [2,]    0   -1 11/3    9
## [3,]    0    0  1/3    1
##
##  multiply row 2 by -1
##                       b
## [1,]     1     2  -5/3     0
## [2,]     0     1 -11/3    -9
## [3,]     0     0   1/3     1
##
##  multiply row 2 by 2 and subtract from row 1
##                       b
## [1,]     1     0  17/3    18
## [2,]     0     1 -11/3    -9
## [3,]     0     0   1/3     1
##
## row: 3
##
##  multiply row 3 by 3
##                       b
## [1,]     1     0  17/3    18
## [2,]     0     1 -11/3    -9
## [3,]     0     0     1     3
```

```
##
##  multiply row 3 by 17/3 and subtract from row 1
##                        b
## [1,]     1     0     0     1
## [2,]     0     1 -11/3    -9
## [3,]     0     0     1     3
##
##  multiply row 3 by 11/3 and add to row 2
##            b
## [1,] 1 0 0 1
## [2,] 0 1 0 2
## [3,] 0 0 1 3
```

Sistemas compatibles indeterminados.

La función solve no funcionara.