Linear Algebra PS-5

Srijan Sen

2022-11-16

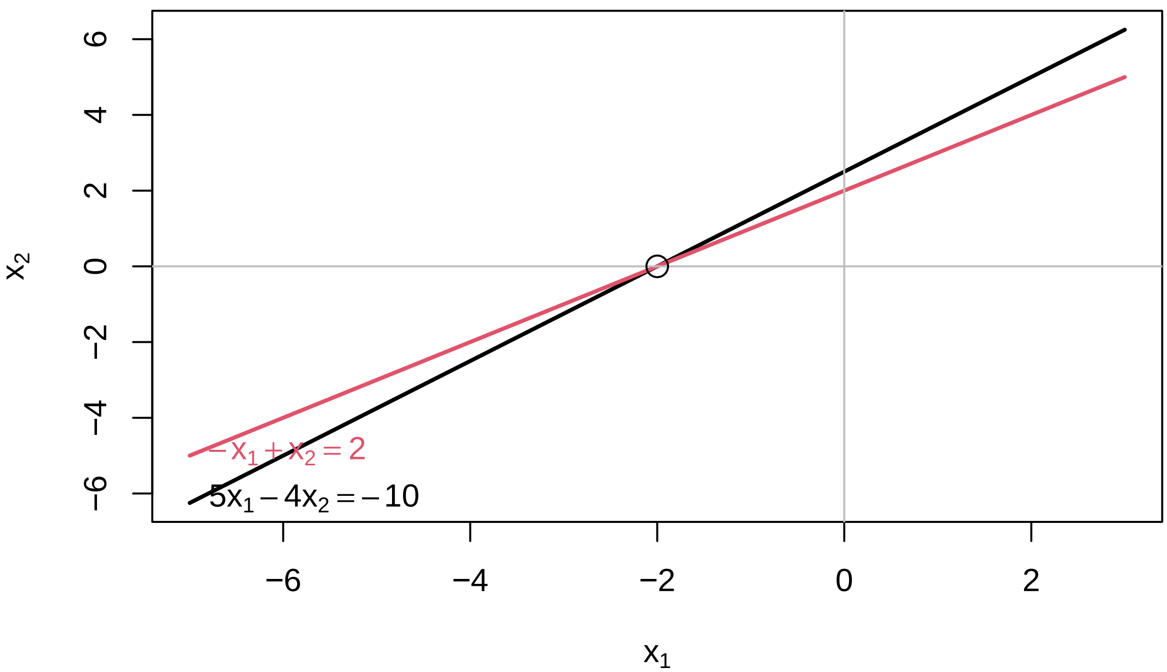
```
library(matlib)
```

Problem 1 (i)

```
a1 = matrix(c(5,-4,-1,1),2,2,byrow=T)
b1 = c(-10,2)

plotEqn(a1,b1)
```

```
## 5*x[1] - 4*x[2]  =  -10
##  -x[1]   + x[2]  =    2
```



Comment

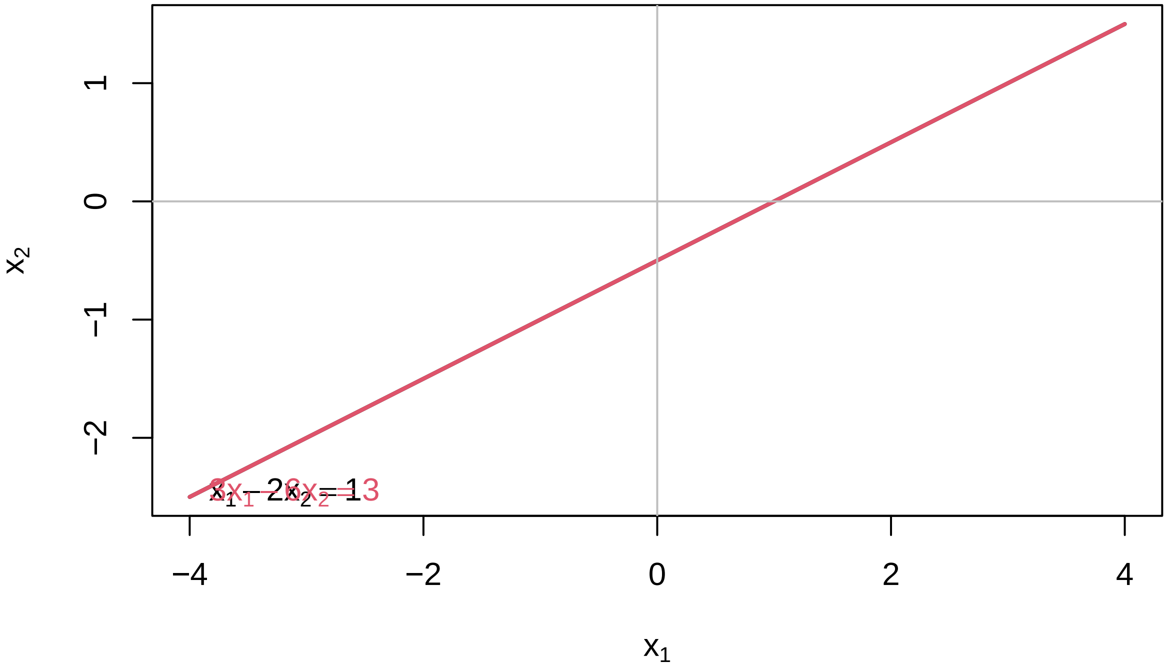
There exists unique solution.

Problem 1 (ii)

```
a2 = matrix(c(1,-2,3,-6),2,2,byrow=T)
b2 = c(1,3)
```

```r
plotEqn(a2,b2)
```

```
##   x[1] - 2*x[2]  =  1
## 3*x[1] - 6*x[2]  =  3
```




```r
Solve(a2,b2,verbose = T)
```

```
##
## Initial matrix:
##      [,1] [,2] [,3]
## [1,]    1   -2    1
## [2,]    3   -6    3
##
## row: 1
##
##  exchange rows 1 and 2
##      [,1] [,2] [,3]
## [1,]    3   -6    3
## [2,]    1   -2    1
##
##  multiply row 1 by 0.3333333
##      [,1] [,2] [,3]
## [1,]    1   -2    1
## [2,]    1   -2    1
##
##  subtract row 1 from row 2
##      [,1] [,2] [,3]
## [1,]    1   -2    1
## [2,]    0    0    0
##
## row: 2
## x1 - 2*x2  =  1
##         0  =  0
```

Comment

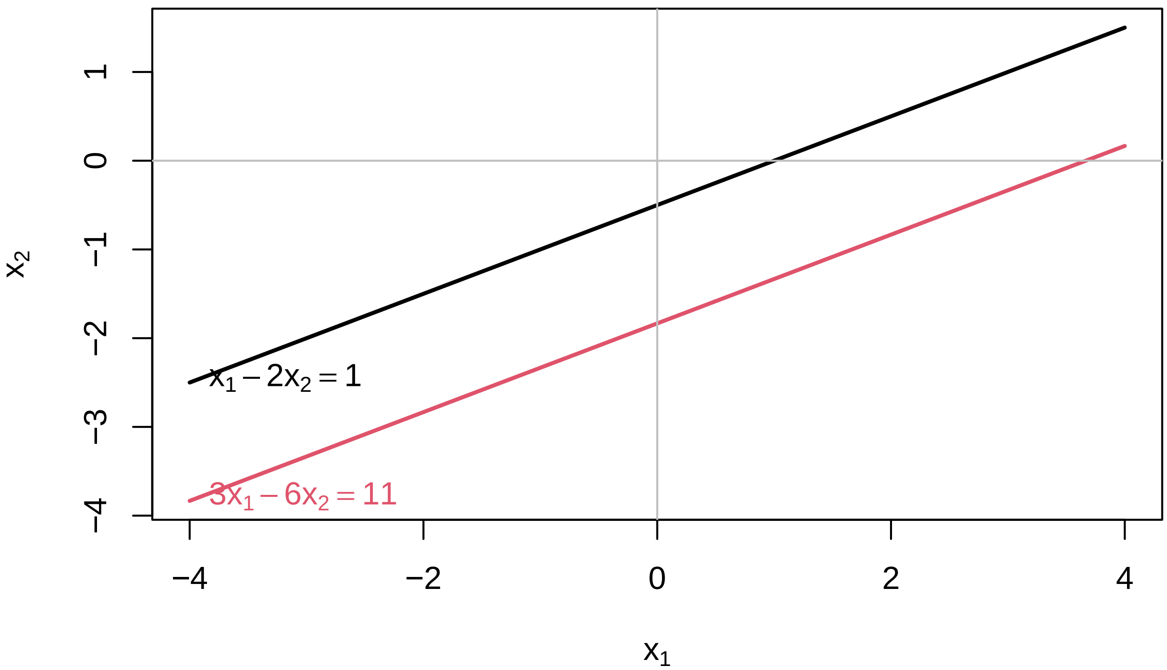
Two lines are overlaying on each other, hence all the points on them can solve the set of equations, Hence, there does not exist any unique solution.

Problem 1 (iii)

```
a3 = matrix(c(1,-2,3,-6),2,2,byrow=T)
b3 = c(1,11)

plotEqn(a3,b3)
```

```
##   x[1] - 2*x[2]  =   1
## 3*x[1] - 6*x[2]  =  11
```




```
Solve(a3,b3,verbose = T)
```

```
##
## Initial matrix:
##      [,1] [,2] [,3]
## [1,]    1   -2    1
## [2,]    3   -6   11
##
## row: 1
##
##  exchange rows 1 and 2
##      [,1] [,2] [,3]
## [1,]    3   -6   11
## [2,]    1   -2    1
##
##  multiply row 1 by 0.3333333
##      [,1] [,2]     [,3]
## [1,]    1   -2 3.666667
## [2,]    1   -2 1.000000
```

```
##
##  subtract row 1 from row 2
##      [,1] [,2]      [,3]
## [1,]    1   -2  3.666667
## [2,]    0    0 -2.666667
##
## row: 2
## x1 - 2*x2  =   3.66666667
##         0  =  -2.66666667
```

Comment

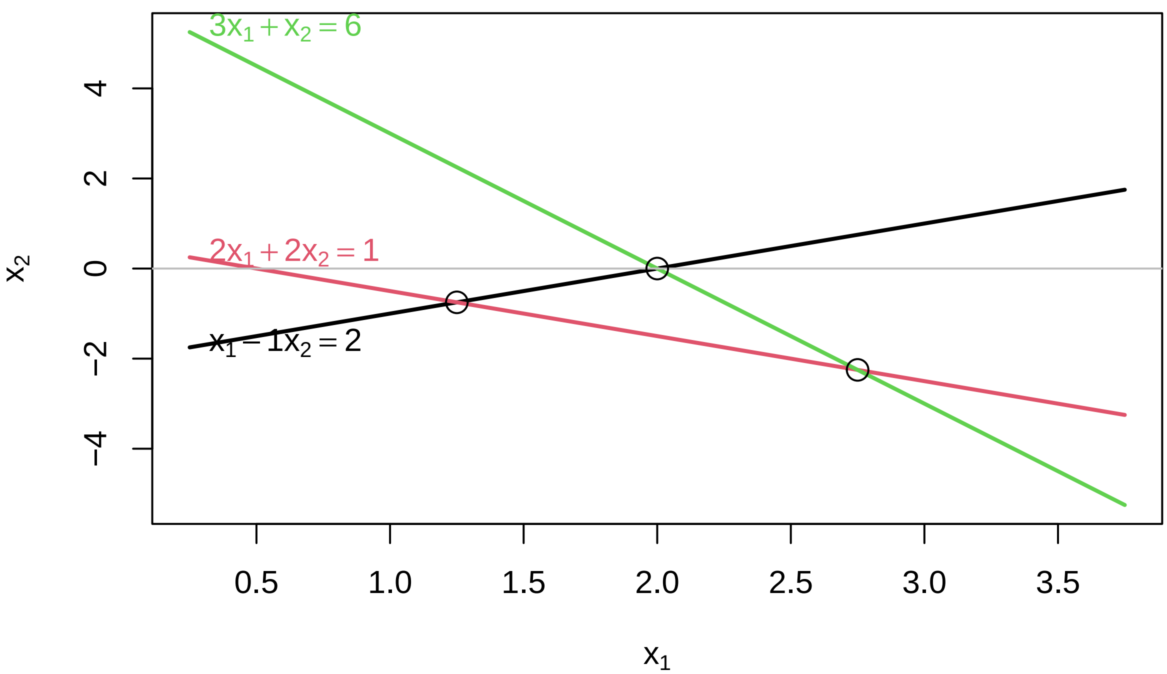
These two lines are parallel to each other, i.e., they will never intersect each other at any time. Hence, there does not exist any solution at all.

Problem 1 (iv)

```
a4 = matrix(c(1,-1,2,2,3,1),3,2,byrow=T)
b4 = c(2,1,6)

plotEqn(a4,b4)
```

```
##   x[1] - 1*x[2]  =  2
## 2*x[1] + 2*x[2]  =  1
## 3*x[1]   + x[2]  =  6
```




```
Solve(a4,b4,verbose = T)
```

```
##
## Initial matrix:
##      [,1] [,2] [,3]
## [1,]    1   -1    2
## [2,]    2    2    1
## [3,]    3    1    6
```

```
##
## row: 1
##
##  exchange rows 1 and 3
##      [,1] [,2] [,3]
## [1,]    3    1    6
## [2,]    2    2    1
## [3,]    1   -1    2
##
##  multiply row 1 by 0.3333333
##      [,1]       [,2] [,3]
## [1,]    1  0.3333333    2
## [2,]    2  2.0000000    1
## [3,]    1 -1.0000000    2
##
##  multiply row 1 by 2 and subtract from row 2
##      [,1]       [,2] [,3]
## [1,]    1  0.3333333    2
## [2,]    0  1.3333333   -3
## [3,]    1 -1.0000000    2
##
##  subtract row 1 from row 3
##      [,1]       [,2] [,3]
## [1,]    1  0.3333333    2
## [2,]    0  1.3333333   -3
## [3,]    0 -1.3333333    0
##
## row: 2
##
##  multiply row 2 by 0.75
##      [,1]       [,2]  [,3]
## [1,]    1  0.3333333  2.00
## [2,]    0  1.0000000 -2.25
## [3,]    0 -1.3333333  0.00
##
##  multiply row 2 by 0.3333333 and subtract from row 1
##      [,1]      [,2]  [,3]
## [1,]    1  0.000000  2.75
## [2,]    0  1.000000 -2.25
## [3,]    0 -1.333333  0.00
##
##  multiply row 2 by 1.333333 and add to row 3
##      [,1] [,2]  [,3]
## [1,]    1    0  2.75
## [2,]    0    1 -2.25
## [3,]    0    0 -3.00
## x1    =   2.75
##   x2  =  -2.25
##    0  =     -3
```

Comment

This set of equations is not solvable. In this problem, set of any two equations taken at a time among the set of three equations, is solvable with unique solution point, but when all three equations are taken at a time, the set is failing to have any unique solution. This kind of set is called inconsistant.

Problem 1 (v)

```
a5 = matrix(c(13,-4,2,-4,11,-2,2,-2,8),3,3,byrow = T)
b5 = c(1,2,6)

plotEqn3d(a5,b5)
```

Comment

There exists a unique solution for the given set of linear equations.

Problem 1 (vi)

```
a6 = matrix(c(1,2,3,2,5,2,6,-3,1),3,3,byrow=T)
b6 = c(6,4,2)

plotEqn3d(a6,b6)
```

Comment

There exists a unique solution for the given set of linear equations.

Problem 2

Let, producer A distributes x_1 and x_2 many bottles from the production site to distributor C & D respectively. Again, producer B distributes x_3 and x_4 many bottles to distributor C & D respectively.

$\therefore$ According to the problem

$$x_1 + x_2 = 475 \quad ...(i)$$
$$x_3 + x_4 = 489 \quad ...(ii)$$
$$x_1 + x_3 = 542 \quad ...(iii)$$
$$x_2 + x_4 = 422 \quad ...(iv)$$

We now solve the following equation as follows:

```
a2 = matrix(c(1,1,0,0,0,0,1,1,1,0,1,0,0,1,0,1),4,4,byrow = T)
a2
```

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

```
b2 = c(475,489,542,422)


det(a2)
```

```
## [1] 0
```

Since, $det(A) = 0$ the given system of linear equations are not solvable.

Problem 3

Let, the person keeps $\$100x_1$ in the bank with annual interest rate 4%, say bank A, and the rest $\$100x_2$ in the other bank with 3% annual rate of interest, say bank B.

$\therefore 100x_1 + 100x_2 = 15000$
$\Rightarrow x_1 + x_2 = 150 \quad ...(i)$

Now, from bank A, he earns an annual interest of $\$4x_1$, and from bank B, he earns that of $\$3x_2$.

$\therefore 4x_1 + 3x_2 = 550 \quad ...(ii)$

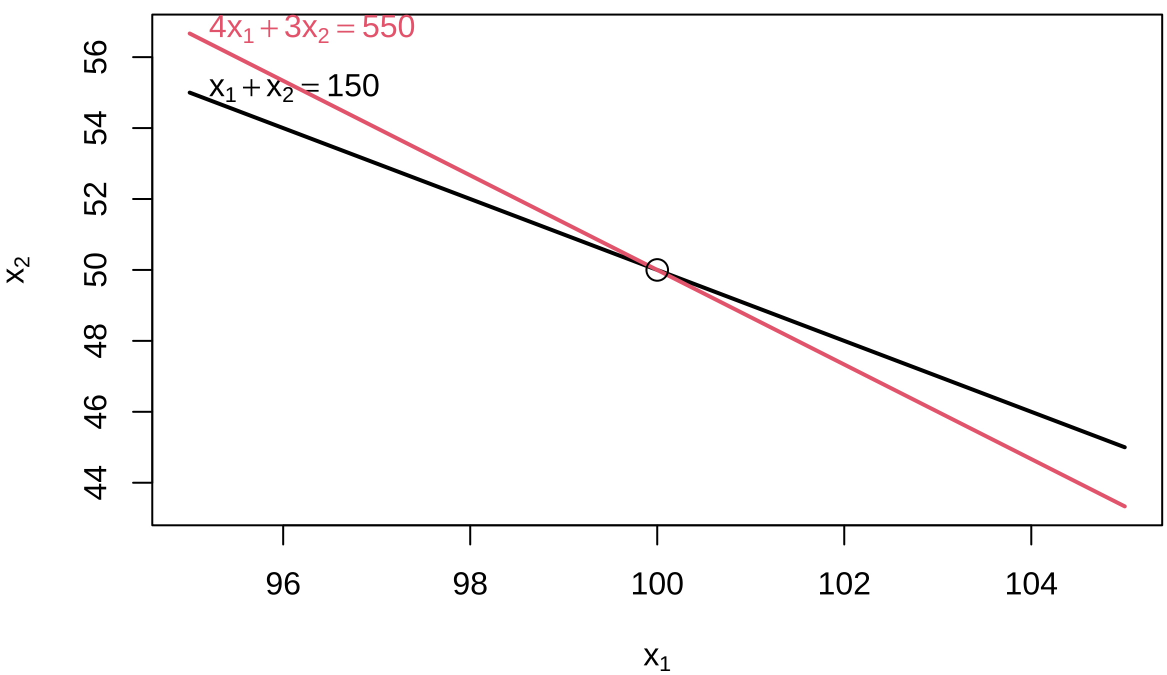
We solve the equation (i) & (ii) as follows,

```
a3 = matrix(c(1,4,1,3),2,2)

b3 = c(150,550)

plotEqn(a3,b3,solution = T)
```

```
##   x[1]   + x[2]  =  150
## 4*x[1] + 3*x[2]  =  550
```




```
t(t(solve(a3,b3)))
```

```
##      [,1]
## [1,]  100
## [2,]   50
```

$\therefore x_1 = 100 \quad \& \quad x_2 = 50$
$\Rightarrow 100x_1 = 10000 \quad \& \quad 100x_2 = 5000$

He has invested \$10,000 in the bank A with 4% annual rate of interest and \$5,000 in bank B with 3% annual rate of interest.

Problem 4

```
Sys.setenv(RGL_USE_NULL=TRUE)
library(matlib)
```

The given system of equation is:

$x_1 + 3x_2 - 2x_3 = -4$
$3x_1 + 7x_2 + x_3 = 4$
$-2x_1 + x_2 + 7x_3 = 7$

```
a = matrix(c(1,3,-2,3,7,1,-2,1,7),3,3, byrow = T)
paste("coefficient matrix A is:")
```

```
## [1] "coefficient matrix A is:"
```

```
a
```

```
##      [,1] [,2] [,3]
## [1,]    1    3   -2
## [2,]    3    7    1
## [3,]   -2    1    7
```

```
b = c(-4,4,7)
paste("vector b is:")
```

```
## [1] "vector b is:"
```

```
b
```

```
## [1] -4  4  7
```

Solve by Gaussian Elimination

```
gaussianElimination(a,b,verbose = T, tol = (.Machine$double.eps)**0.5)
```

```
##
## Initial matrix:
##      [,1] [,2] [,3] [,4]
## [1,]    1    3   -2   -4
## [2,]    3    7    1    4
## [3,]   -2    1    7    7
##
## row: 1
##
##  exchange rows 1 and 2
##      [,1] [,2] [,3] [,4]
## [1,]    3    7    1    4
## [2,]    1    3   -2   -4
## [3,]   -2    1    7    7
##
##  multiply row 1 by 0.3333333
```

```
##      [,1]     [,2]       [,3]      [,4]
## [1,]    1 2.333333  0.3333333  1.333333
## [2,]    1 3.000000 -2.0000000 -4.000000
## [3,]   -2 1.000000  7.0000000  7.000000
##
##  subtract row 1 from row 2
##      [,1]      [,2]       [,3]      [,4]
## [1,]    1 2.3333333  0.3333333  1.333333
## [2,]    0 0.6666667 -2.3333333 -5.333333
## [3,]   -2 1.0000000  7.0000000  7.000000
##
##  multiply row 1 by 2 and add to row 3
##      [,1]      [,2]       [,3]      [,4]
## [1,]    1 2.3333333  0.3333333  1.333333
## [2,]    0 0.6666667 -2.3333333 -5.333333
## [3,]    0 5.6666667  7.6666667  9.666667
##
## row: 2
##
##  exchange rows 2 and 3
##      [,1]      [,2]       [,3]      [,4]
## [1,]    1 2.3333333  0.3333333  1.333333
## [2,]    0 5.6666667  7.6666667  9.666667
## [3,]    0 0.6666667 -2.3333333 -5.333333
##
##  multiply row 2 by 0.1764706
##      [,1]      [,2]       [,3]      [,4]
## [1,]    1 2.3333333  0.3333333  1.333333
## [2,]    0 1.0000000  1.3529412  1.705882
## [3,]    0 0.6666667 -2.3333333 -5.333333
##
##  multiply row 2 by 2.333333 and subtract from row 1
##      [,1]      [,2]      [,3]      [,4]
## [1,]    1 0.0000000 -2.823529 -2.647059
## [2,]    0 1.0000000  1.352941  1.705882
## [3,]    0 0.6666667 -2.333333 -5.333333
##
##  multiply row 2 by 0.6666667 and subtract from row 3
##      [,1] [,2]      [,3]      [,4]
## [1,]    1    0 -2.823529 -2.647059
## [2,]    0    1  1.352941  1.705882
## [3,]    0    0 -3.235294 -6.470588
##
## row: 3
##
##  multiply row 3 by -0.3090909
##      [,1] [,2]      [,3]      [,4]
## [1,]    1    0 -2.823529 -2.647059
## [2,]    0    1  1.352941  1.705882
## [3,]    0    0  1.000000  2.000000
##
##  multiply row 3 by 2.823529 and add to row 1
##      [,1] [,2]     [,3]     [,4]
## [1,]    1    0 0.000000 3.000000
```

```
## [2,]    0    1 1.352941 1.705882
## [3,]    0    0 1.000000 2.000000
##
##  multiply row 3 by 1.352941 and subtract from row 2
##      [,1] [,2] [,3] [,4]
## [1,]    1    0    0    3
## [2,]    0    1    0   -1
## [3,]    0    0    1    2
```

$\therefore$ The solution is:

$x_1 = 3$
$x_2 = -1$
$x_3 = 2$

Solve by LU Decomposition

```
library(matrixcalc)
```

```
##
## Attaching package: 'matrixcalc'

## The following object is masked from 'package:matlib':
##
##     vec
```

```
lu_a = lu.decomposition(a)
lu_a$L
```

```
##      [,1] [,2] [,3]
## [1,]    1  0.0    0
## [2,]    3  1.0    0
## [3,]   -2 -3.5    1
```

```
lu_a$U
```

```
##      [,1] [,2] [,3]
## [1,]    1    3 -2.0
## [2,]    0   -2  7.0
## [3,]    0    0 27.5
```

$Ax = b$
$LUx = b$
Now, $Ux = y$
$\therefore Ly = b$

We first solve $y = L^{-1}b$

```
y = solve(lu_a$L,b)
y
```

```
## [1] -4 16 55
```

Then we solve, $x = U^{-1}y$

```r
x = solve(lu_a$U,y)
t(t(x))
```

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

∴ The solution is:

$x_1 = 3$
$x_2 = -1$
$x_3 = 2$

Solve by QR Factorization

```r
qra = qr(a, tol = 1e-7, LAPACK = F)
Ra = qr.R(qra)
Qa = qr.Q(qra)
```

The R matrix is:

```r
Ra
```

```
##           [,1]      [,2]      [,3]
## [1,] -3.741657 -5.879747  3.474396
## [2,]  0.000000 -4.942527 -5.751829
## [3,]  0.000000  0.000000 -2.974059
```

The Q matrix is:

```r
Qa
```

```
##            [,1]       [,2]       [,3]
## [1,] -0.2672612 -0.2890367  0.9192547
## [2,] -0.8017837 -0.4624587 -0.3785166
## [3,]  0.5345225 -0.8382063 -0.1081476
```

The R^{-1} matrix is:

```r
Ra_inv = inv(Ra)
Ra_inv
```

```
##            [,1]       [,2]       [,3]
## [1,] -0.2672612  0.3179403 -0.9271200
## [2,]  0.0000000 -0.2023257  0.3912977
## [3,]  0.0000000  0.0000000 -0.3362408
```

The Q^T matrix is:

```r
Qa_tran = t(Qa)
Qa_tran
```

```
##            [,1]       [,2]       [,3]
## [1,] -0.2672612 -0.8017837  0.5345225
## [2,] -0.2890367 -0.4624587 -0.8382063
## [3,]  0.9192547 -0.3785166 -0.1081476
```

The pseudo inverse of the matrix given is:

```
pseudoa = Ra_inv%*%Qa_tran
pseudoa
```

```
##            [,1]        [,2]        [,3]
## [1,] -0.8727273  0.41818182 -0.30909091
## [2,]  0.4181818 -0.05454545  0.12727273
## [3,] -0.3090909  0.12727273  0.03636364
```

The final solution is:

```
sol = pseudoa%*%b
t(t(sol))
```

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

Solve by Cholesky Factorization

```
library(pracma)
```

```
##
## Attaching package: 'pracma'

## The following objects are masked from 'package:matlib':
##
##     angle, inv
```

```
isposdef(a)
```

```
## [1] FALSE
```

Since, A is not a positive definite matrix, hence Cholesky factorization is not applicable.

Problem 5

```
library(matlib)
library(pracma)

set.seed(4865299)
mat = matrix(runif(12,0,1),4,3,byrow = T)
bvec5 = c(1,2,3,4)

qrmat = qr(mat, tol = 1e-7, LAPACK = F)

R = qr.R(qrmat)
Q = qr.Q(qrmat)
```

The R matrix is:

```
R
```

```
##            [,1]       [,2]       [,3]
## [1,] -0.7422313 -0.5040855 -1.3101599
## [2,]  0.0000000  0.6096591  0.4796298
## [3,]  0.0000000  0.0000000 -0.1842760
```

The Q matrix is:

```
Q
```

```
##              [,1]        [,2]       [,3]
## [1,] -0.05600242  0.35320857 -0.1125353
## [2,] -0.32698797  0.07784176 -0.9277743
## [3,] -0.25818526  0.88239511  0.2217669
## [4,] -0.90734942 -0.30093727  0.2781909
```

The R^{-1} matrix is:

```
R_inv = inv(R)
R_inv
```

```
##            [,1]      [,2]      [,3]
## [1,] -1.347289 -1.113981  6.679466
## [2,]  0.000000  1.640261  4.269238
## [3,]  0.000000  0.000000 -5.426642
```

The Q^T matrix is:

```
Q_tran = t(Q)
Q_tran
```

```
##              [,1]        [,2]       [,3]       [,4]
## [1,] -0.05600242 -0.32698797 -0.2581853 -0.9073494
## [2,]  0.35320857  0.07784176  0.8823951 -0.3009373
## [3,] -0.11253531 -0.92777426  0.2217669  0.2781909
```

The pseudo inverse of the matrix given is:

```
pseudo = R_inv%*%Q_tran
pseudo
```

```
##              [,1]      [,2]      [,3]       [,4]
## [1,] -1.06969213 -5.843204  0.846163  3.4158669
## [2,]  0.09891426 -3.833208  2.394134  0.6940473
## [3,]  0.61068890  5.034699 -1.203450 -1.5096424
```

The final solution is:

```
sol = pseudo%*%bvec5
t(t(sol))
```

```
##          [,1]
## [1,] 3.445857
## [2,] 2.391089
## [3,] 1.031169
```

```
qr.solve(mat, bvec5)
```

```
## [1] 3.445857 2.391089 1.031169
```

```
solve(qr(mat, LAPACK = TRUE), bvec5)
```

```
## [1] 3.445857 2.391089 1.031169
```

Problem 6

```r
library(matrixcalc)
library(clusterGeneration)
```

```
## Loading required package: MASS
```

```r
cho_mat = matrix(runif(36,0,1),6,6)
cho_mat[lower.tri(cho_mat)] = t(cho_mat)[lower.tri(cho_mat)]
cho_mat
```

```
##           [,1]      [,2]      [,3]        [,4]      [,5]      [,6]
## [1,] 0.2355056 0.6842727 0.9780618 0.212791326 0.7616378 0.3828227
## [2,] 0.6842727 0.6018899 0.4316654 0.485680262 0.6125051 0.2610462
## [3,] 0.9780618 0.4316654 0.6239027 0.315389807 0.7553102 0.9963399
## [4,] 0.2127913 0.4856803 0.3153898 0.006591378 0.1512169 0.8980854
## [5,] 0.7616378 0.6125051 0.7553102 0.151216928 0.6946293 0.7219685
## [6,] 0.3828227 0.2610462 0.9963399 0.898085385 0.7219685 0.6558913
```

```r
b6 = 1:6

isposdef(cho_mat)
```

```
## [1] FALSE
```

The matrix is not positive definite. So we approach for LU decomposition.

```r
lu_mat = lu.decomposition(cho_mat)
lu_mat$L
```

```
##           [,1]       [,2]       [,3]       [,4]      [,5] [,6]
## [1,] 1.0000000 0.00000000  0.0000000  0.0000000   0.00000    0
## [2,] 2.9055470 1.00000000  0.0000000  0.0000000   0.00000    0
## [3,] 4.1530292 1.73854505  1.0000000  0.0000000   0.00000    0
## [4,] 0.9035509 0.09564688 -0.4491531  1.0000000   0.00000    0
## [5,] 3.2340532 1.15449266  0.4981848  0.6639043   1.00000    0
## [6,] 1.6255351 0.61405552  1.1785707 -3.1772901 -18.51377    1
```

```r
lu_mat$U
```

```
##               [,1]       [,2]       [,3]       [,4]        [,5]       [,6]
## [1,]  2.355056e-01  0.6842727  0.9780618  0.2127913  0.76163776  0.3828227
## [2,]  0.000000e+00 -1.3862966 -2.4101390 -0.1325949 -1.60046922 -0.8512631
## [3,] -1.110223e-16  0.0000000  0.7521189 -0.3378165  0.37469422  0.8864253
## [4,] -4.986601e-17  0.0000000  0.0000000 -0.3247254 -0.21558661  1.0317469
## [5,]  8.841588e-17  0.0000000  0.0000000  0.0000000  0.03564418 -0.6599083
## [6,]  1.609320e-15  0.0000000  0.0000000  0.0000000  0.00000000 -9.4276251
```

$Ax = b$
$LUx = b$
Now, $Ux = y$
$\therefore Ly = b$

We first solve $y = L^{-1}b$

```r
y = solve(lu_mat$L,b6)
y
```

```
## [1]  1.0000000 -0.9055470  0.4213051  3.3722923  0.3626271 21.8623293
```

Then we solve, $x = U^{-1}y$

```
x = solve(lu_mat$U,y)
t(t(x))
```

```
##              [,1]
## [1,]  14.769801
## [2,]   2.296243
## [3,]  21.408168
## [4,]   3.995929
## [5,] -32.759256
## [6,]  -2.318965
```

```
t(t(solve(cho_mat,b6)))
```

```
##              [,1]
## [1,]  14.769801
## [2,]   2.296243
## [3,]  21.408168
## [4,]   3.995929
## [5,] -32.759256
## [6,]  -2.318965
```