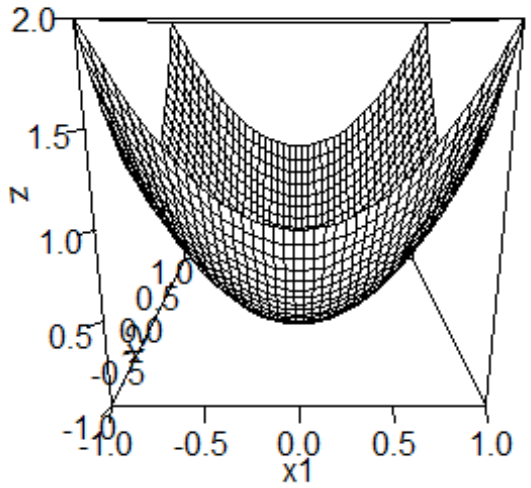
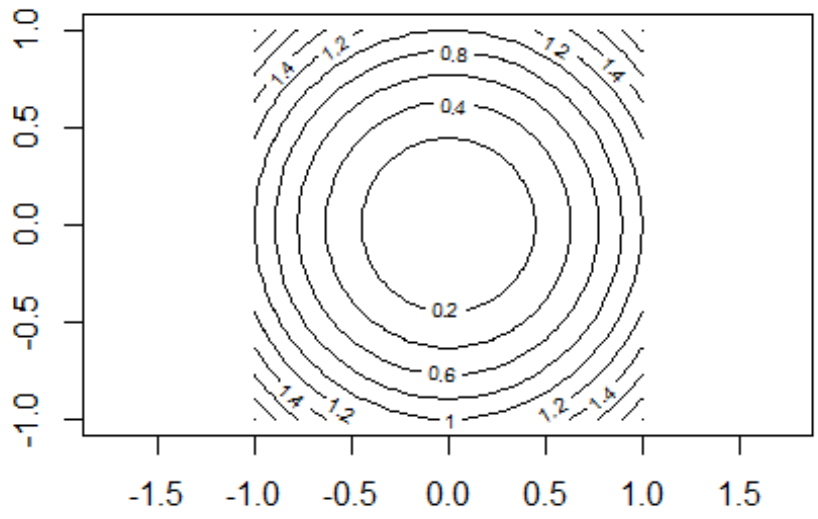
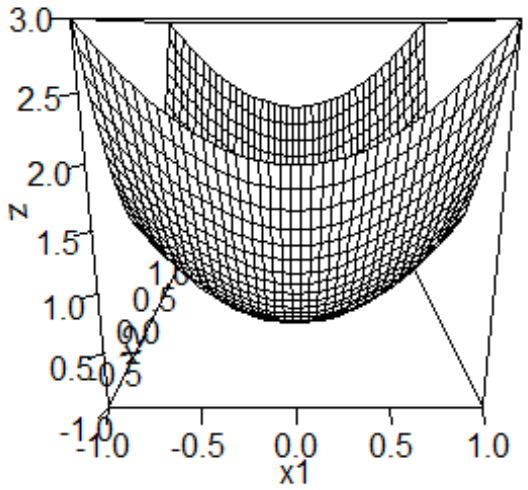
```
###Exercise 3

remove(list = ls())

x1 = seq(-1,1,le=40)
x2=x1

### i)

A1=matrix(c(1,0,0,1),2,2)
f= function(v1,v2)
{
  A1[1,1]*v1^2+A1[2,2]*v2^2+2*A1[1,2]*v1*v2
}
resouter=outer(x1,x2,f)
persp(x1,x2,asp = 1,resouter,ticktype="detailed", zlab = "z")
```



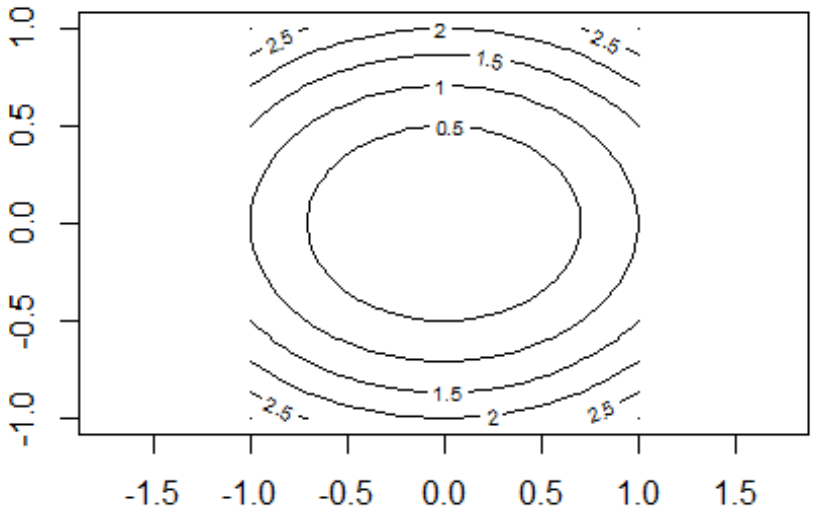

```
contour(x1,x2,resouter,asp = 1)
```



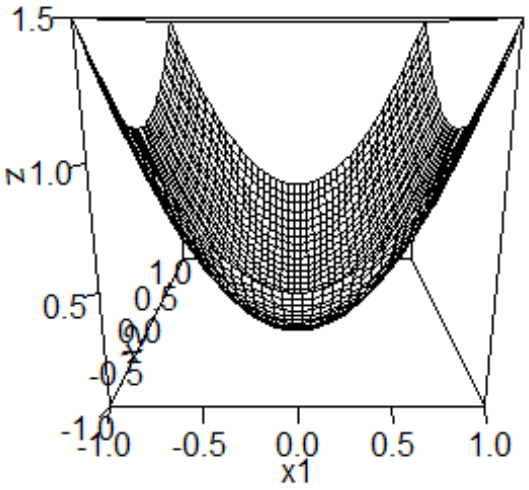
```
### ii)

A2=matrix(c(1,0,0,2),2,2)
f= function(v1,v2)
{
  A2[1,1]*v1^2+A2[2,2]*v2^2+2*A2[1,2]*v1*v2
}
resouter=outer(x1,x2,f)
persp(x1,x2,asp = 1,resouter,ticktype="detailed", zlab = "z")
```



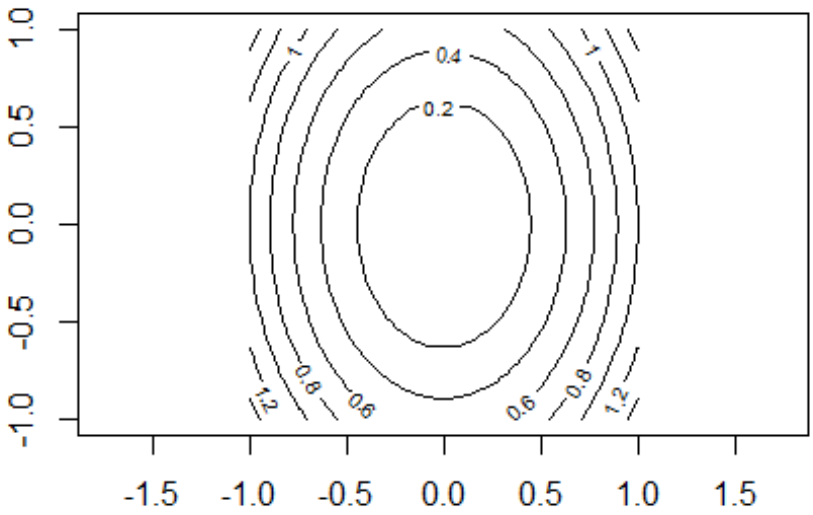

```
contour(x1,x2,resouter,asp=1)
```



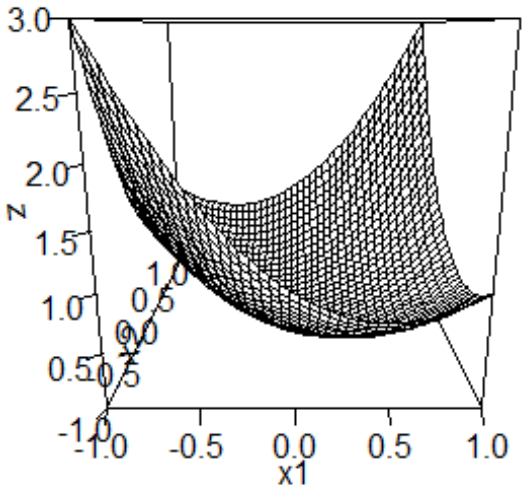
```
###  iii)

A3=matrix(c(1,0,0,0.5),2,2)
f= function(v1,v2)
{
  A3[1,1]*v1^2+A3[2,2]*v2^2+2*A3[1,2]*v1*v2
}
resouter=outer(x1,x2,f)
persp(x1,x2,asp = 1,resouter,ticktype="detailed", zlab = "z")
```



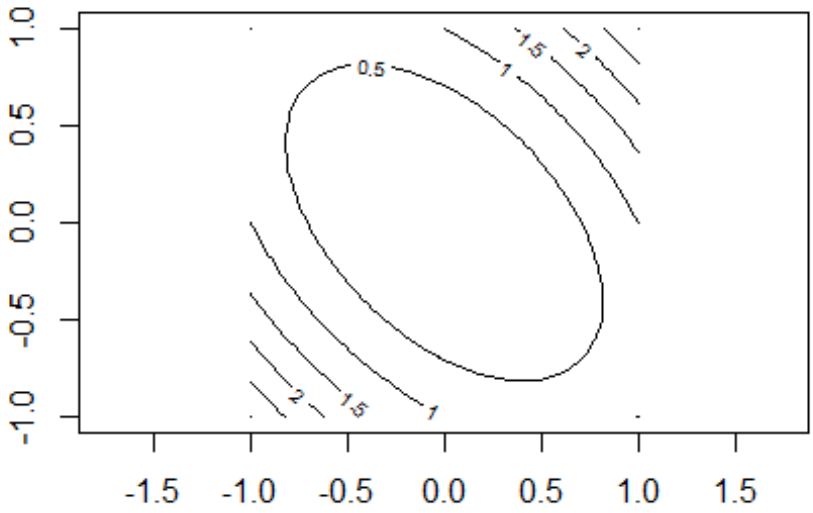

```
contour(x1,x2,resouter,asp=1)
```



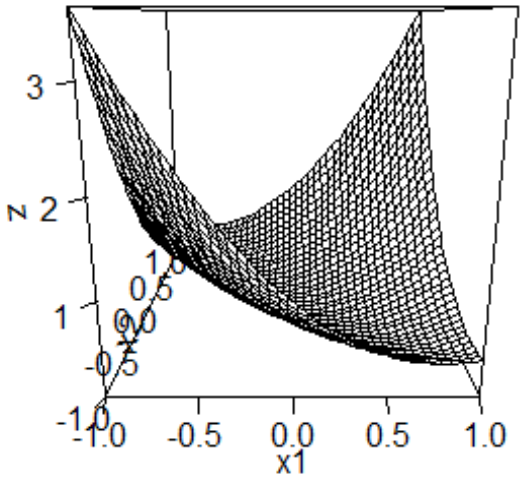
```
###  iv)

A4=matrix(c(1,0.5,0.5,1),2,2)
f= function(v1,v2)
{
  A4[1,1]*v1^2+A4[2,2]*v2^2+2*A4[1,2]*v1*v2
}
resouter=outer(x1,x2,f)
persp(x1,x2,asp = 1,resouter,ticktype="detailed", zlab = "z")
```



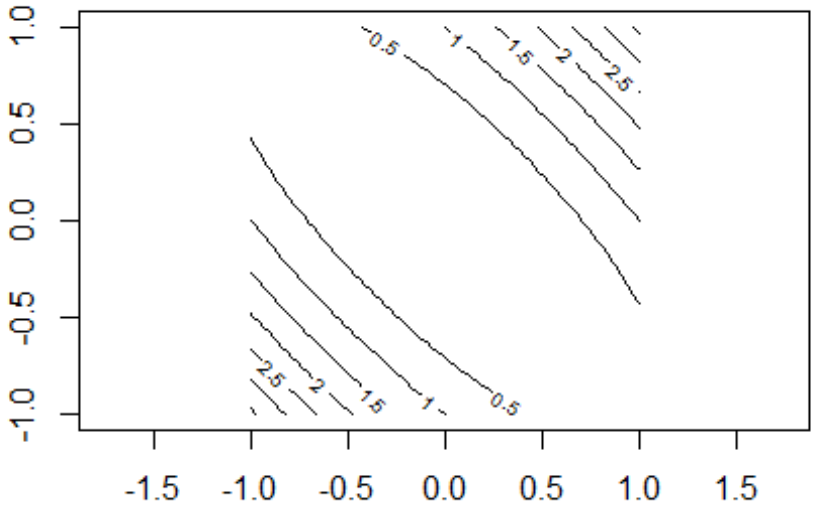

```
contour(x1,x2,resouter,asp=1)
```



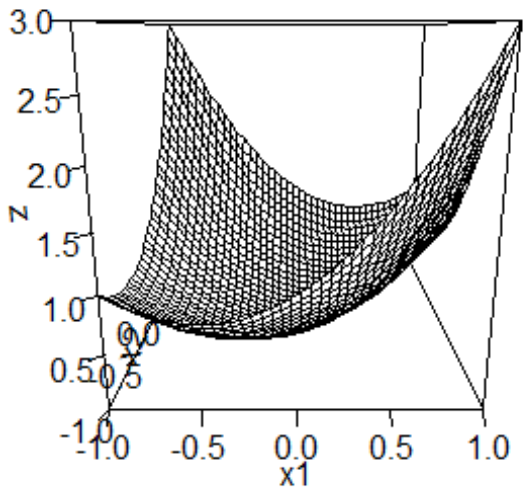
```
###  v)

A5=matrix(c(1,0.8,0.8,1),2,2)
f= function(v1,v2)
{
  A5[1,1]*v1^2+A5[2,2]*v2^2+2*A5[1,2]*v1*v2
}
resouter=outer(x1,x2,f)
persp(x1,x2,asp = 1,resouter,ticktype="detailed", zlab = "z")
```



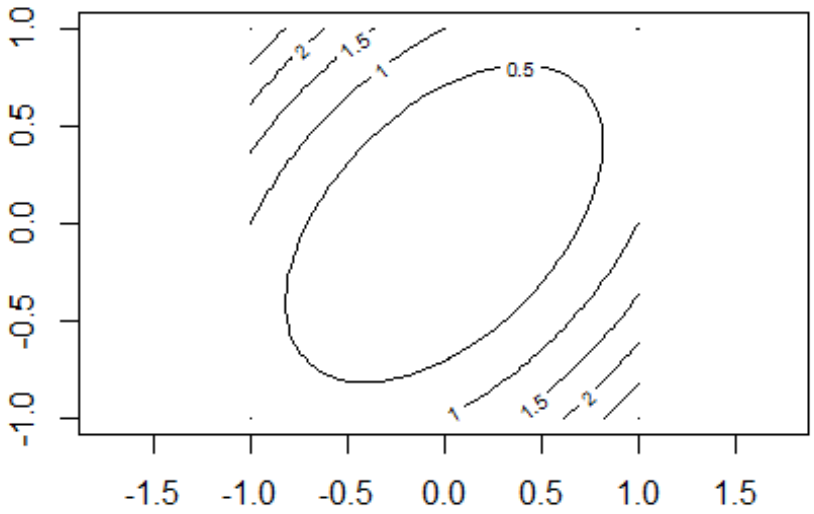

```
contour(x1,x2,resouter,asp=1)
```



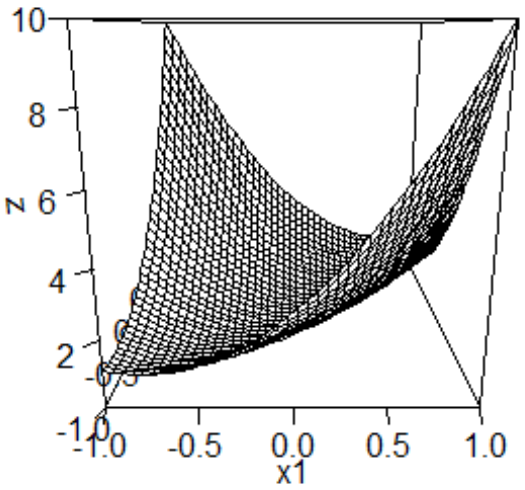
```
###  vi)

A6=matrix(c(1,-0.5,-0.5,1),2,2)
f= function(v1,v2)
{
  A6[1,1]*v1^2+A6[2,2]*v2^2+2*A6[1,2]*v1*v2
}
resouter=outer(x1,x2,f)
persp(x1,x2,asp = 1,resouter,ticktype="detailed", zlab = "z")
```



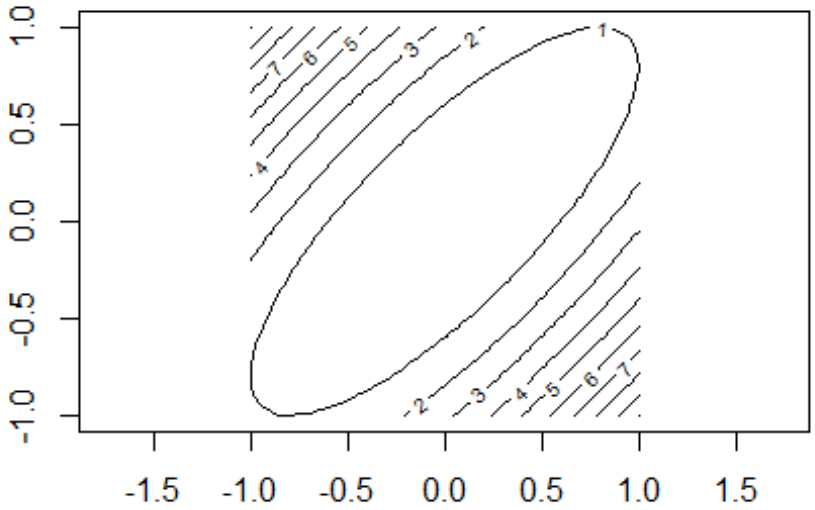

```
contour(x1,x2,resouter,asp=1)
```



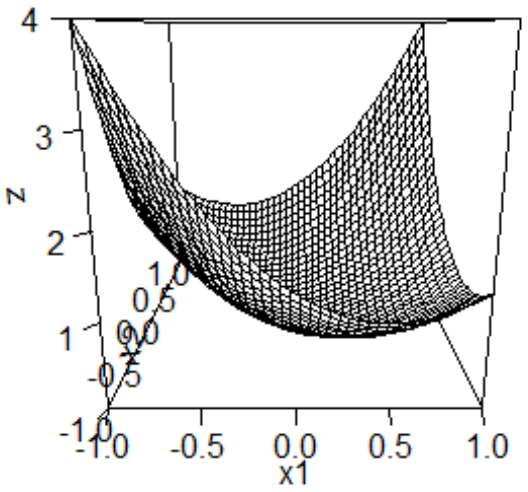
```
###  vii)

A7 = solve(A5)
f= function(v1,v2)
{
  A7[1,1]*v1^2+A7[2,2]*v2^2+2*A7[1,2]*v1*v2
}
resouter=outer(x1,x2,f)
persp(x1,x2,asp = 1,resouter,ticktype="detailed", zlab = "z")
```



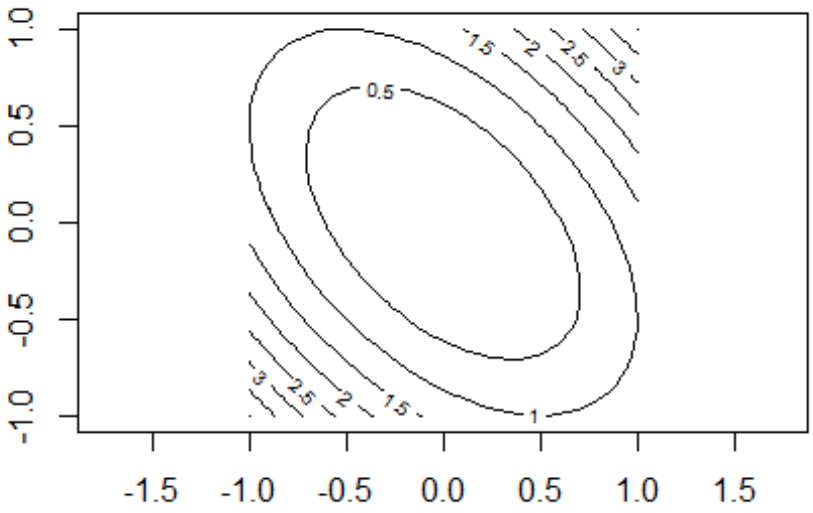

```
contour(x1,x2,resouter,asp=1)
```



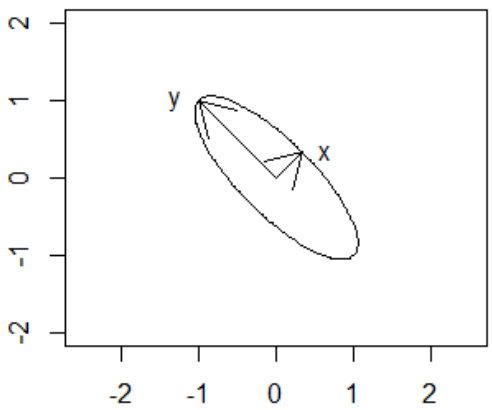
```
###  viii)

A8 = solve(A6)
f= function(v1,v2)
{
  A8[1,1]*v1^2+A8[2,2]*v2^2+2*A8[1,2]*v1*v2
}
resouter=outer(x1,x2,f)
persp(x1,x2,asp = 1,resouter,ticktype="detailed", zlab = "z")
```




```
contour(x1,x2,resouter,asp=1)
```



```
### Exercise 5

remove(list = ls())

x1 = seq(-2,2,le=100)
x2 = x1
A = matrix(c(5,4,4,5),2,2)

f= function(v1,v2)
{
  A[1,1]*v1^2+A[2,2]*v2^2+2*A[1,2]*v1*v2
}

resouter=outer(x1,x2,f)
contour(x1 ,x2 ,resouter,levels = 2, asp = 1, drawlabels = FALSE, xlim=c(-2
,2), ylim = c(-2,2))

E=eigen(A)
E

## eigen() decomposition
## $values
## [1] 9 1
##
## $vectors
##           [,1]       [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068

###Diese Werte stimmen mit jenen überein, welche mit Hand berechnet wurden.

e10 = E$vectors[,1]
e11 = (sqrt(2)/sqrt(9))*e10
arrows(0,0,e11[1],e11[2],col = "black")
text(e11[1]+0.3,e11[2],"x")

e20 = E$vectors[,2]
e21 = (sqrt(2)/sqrt(1))*e20
arrows(0,0,e21[1],e21[2],col = "black")
text(e21[1]-0.3,e21[2],"y")
```



```
le11 = sqrt(sum(e11^2))
paste("Die Länge des Eigenvektors x beträgt:", round(le11, digits = 6))
```

```
## [1] "Die Länge des Eigenvektors x beträgt: 0.471405"
```

```
le21 = sqrt(sum(e21^2))
paste("Die Länge des Eigenvektors y beträgt:", round(le21, digits = 6))
```

```
## [1] "Die Länge des Eigenvektors y beträgt: 1.414214"
```

```
###Exercise 6

remove(list = ls())

### a)
A = matrix(c(13,-4,2,-4,13,-2,2,-2,10),3,3)
EA = eigen(A)
lamEA = EA$values

###Eigenvalue matrix
m_lamEA = matrix(c(18,0,0,0,9,0,0,0,9),3,3)
m_lamEA
```

```
##      [,1] [,2] [,3]
## [1,]   18    0    0
## [2,]    0    9    0
## [3,]    0    0    9
```

```
###Eigenvector matrix
eiVA = EA$vectors
eiVA
```

```
##            [,1]       [,2]      [,3]
## [1,]  0.6666667 -0.7453560 0.0000000
## [2,] -0.6666667 -0.5962848 0.4472136
## [3,]  0.3333333  0.2981424 0.8944272
```

```
###sectral decompostion
eiVA%*%m_lamEA%*%t(eiVA)
```

```
##      [,1] [,2] [,3]
## [1,]   13   -4    2
## [2,]   -4   13   -2
## [3,]    2   -2   10
```

```
###Wie zu erwarten entspreicht das Ergebnis der sectral decomposition der M
atrix.
```

```
### b)

eiEA1 = EA$vectors[,1]
eiEA2 = EA$vectors[,2]
eiEA3 = EA$vectors[,3]

###  i)
lamEA[1]*eiEA1%*%t(eiEA1)

##      [,1] [,2] [,3]
## [1,]    8   -8    4
## [2,]   -8    8   -4
## [3,]    4   -4    2

###  ii)
lamEA[1]*eiEA1%*%t(eiEA1)+lamEA[2]*eiEA2%*%t(eiEA2)

##      [,1] [,2] [,3]
## [1,]   13 -4.0  2.0
## [2,]   -4 11.2 -5.6
## [3,]    2 -5.6  2.8

###  iii)
lamEA[1]*eiEA1%*%t(eiEA1)+lamEA[2]*eiEA2%*%t(eiEA2)+lamEA[3]*eiEA3%*%t(eiEA
3)

##      [,1] [,2] [,3]
## [1,]   13   -4    2
## [2,]   -4   13   -2
## [3,]    2   -2   10

### c)
sqm_lamEA = sqrt(matrix(c(18,0,0,0,9,0,0,0,9),3,3))
sqm_A = eiVA%*%sqm_lamEA%*%t(eiVA)
sqm_A

##            [,1]       [,2]       [,3]
## [1,]  3.5522847 -0.5522847  0.2761424
## [2,] -0.5522847  3.5522847 -0.2761424
## [3,]  0.2761424 -0.2761424  3.1380712
```