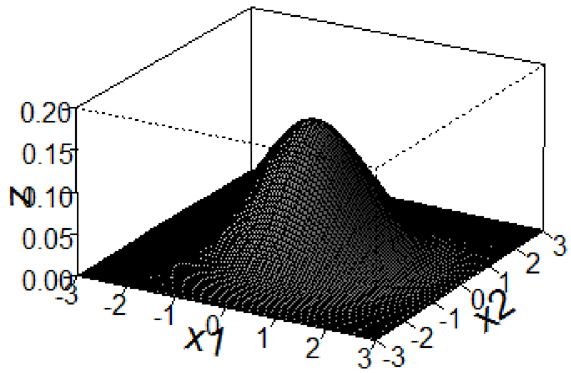
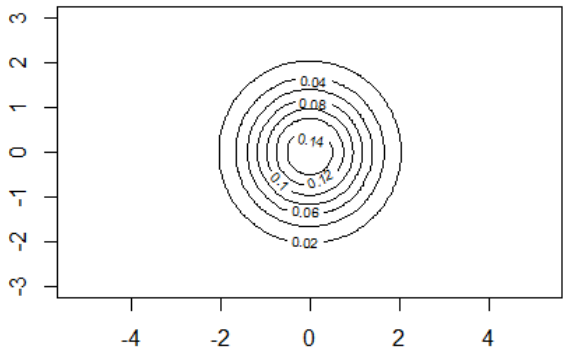
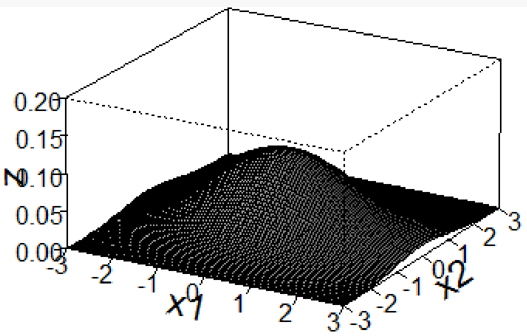
```
### Exercise 1

remove(list = ls())

A = matrix(c(13,-4,2,-4,13,-2,2,-2,10),3,3)

# a) Correlation matrix C

C = cov2cor(A)
C

##            [,1]       [,2]       [,3]
## [1,]  1.0000000 -0.3076923  0.1754116
## [2,] -0.3076923  1.0000000 -0.1754116
## [3,]  0.1754116 -0.1754116  1.0000000

# b)

CE = eigen(C)
CE

## eigen() decomposition
## $values
## [1] 1.4457487 0.8619436 0.6923077
##
## $vectors
##            [,1]       [,2]          [,3]
## [1,]  0.6178686 -0.3438582  7.071068e-01
## [2,] -0.6178686  0.3438582  7.071068e-01
## [3,]  0.4862889  0.8737981 -1.110223e-16

### Die Eigenwerte und die Eigenvektoren der Korrelationsmatrix sind nicht
die gleichen, wie die der Kovarianzmatrix A

## Exercise 2

remove(list = ls())

x1 = seq(-3,3, le = 100)
x2 = seq(-3,3, le = 100)

# i)

m = c(0,0)
m1 = 0
m2 = 0

S11 = 1
S22 = 1

COR12 = 0

f= function(v1,v2)
{
  (1/(2*pi))*exp((-1/2)*(((v1-m1)^2)+((v2-m2)^2)))
}
```

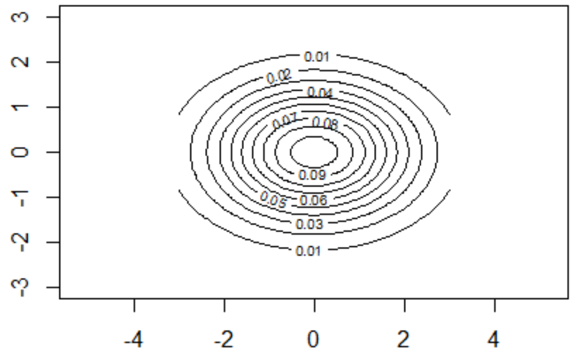
```
z = outer(x1,x2,f)
persp(x1,x2,z,main="",cex.lab=1.5,theta=30,phi=20,r=50,d=0.1,expand=0.5,
      ltheta=90,lphi=180,shade=0.75,ticktype="detailed",nticks=5,
      xlim=c(-3,3),ylim=c(-3,3),zlim=c(0,0.2))
contour(x1,x2,z,asp = 1)
```



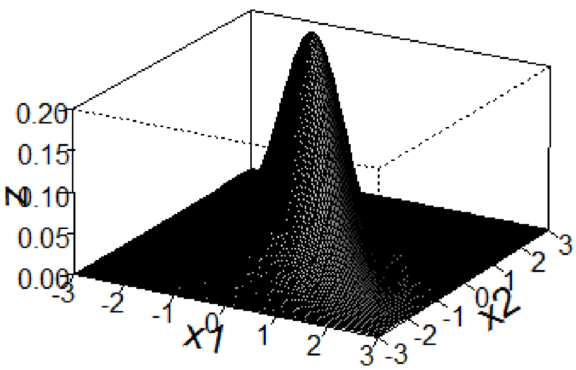


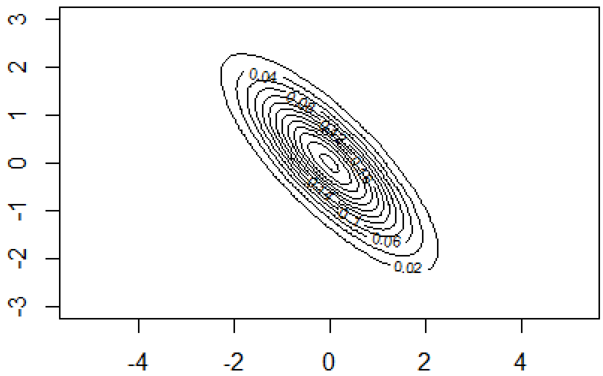

```
# ii)

m = c(0,0)
m1 = 0
m2 = 0

S11 = 1.5^2
S22 = 1

COR12 = 0

f= function(v1,v2)
{
  (1/((2*pi)*sqrt(S11*S22)*sqrt(1-COR12^2)))*exp((-1/(2*(1-COR12^2)))*(((v1
-m1)^2/S11)
  -2*COR12*((v1-m1)/sqrt(S11))*((v2-m2)/sqrt(S22))+((v2-m2)^2)/S22))
}

z = outer(x1,x2,f)
persp(x1,x2,z,main="",cex.lab=1.5,theta=30,phi=20,r=50,d=0.1,expand=0.5,
      ltheta=90,lphi=180,shade=0.75,ticktype="detailed",nticks=5,
      xlim=c(-3,3),ylim=c(-3,3),zlim=c(0,0.2))
contour(x1,x2,z,asp = 1)
```





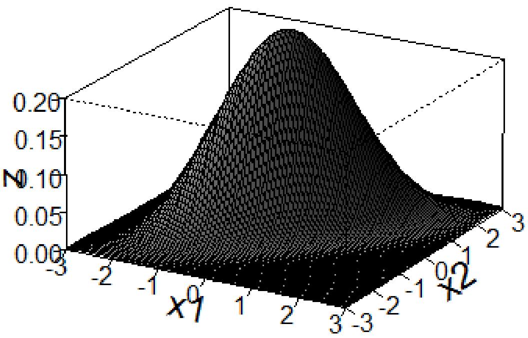
```r
# iii)

m = c(0,0)
m1 = 0
m2 = 0

S11 = 1
S22 = 1

COR12 = -0.8

f= function(v1,v2)
{
  (1/((2*pi)*sqrt(S11*S22)*sqrt(1-COR12^2)))*exp((-1/(2*(1-COR12^2)))*(((v1
-m1)^2/S11)
  -2*COR12*((v1-m1)/sqrt(S11))*((v2-m2)/sqrt(S22))+((v2-m2)^2)/S22))
}
z = outer(x1,x2,f)

persp(x1,x2,z,main="",cex.lab=1.5,theta=30,phi=20,r=50,d=0.1,expand=0.5,
      ltheta=90,lphi=180,shade=0.75,ticktype="detailed",nticks=5,
      xlim=c(-3,3),ylim=c(-3,3),zlim=c(0,0.2))
contour(x1,x2,z,asp = 1)
```

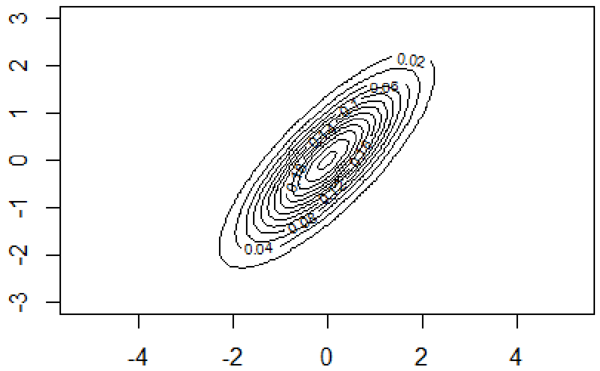





```r
# iv)

m = c(0,0)
m1 = 0
m2 = 0

S11 = 1
S22 = 1

COR12 = 0.8

f= function(v1,v2)
{
  (1/((2*pi)*sqrt(S11*S22)*sqrt(1-COR12^2)))*exp((-1/(2*(1-COR12^2)))*(((v1
-m1)^2/S11)
  -2*COR12*((v1-m1)/sqrt(S11))*((v2-m2)/sqrt(S22))+((v2-m2)^2)/S22))
}
```

```
z = outer(x1,x2,f)

persp(x1,x2,z,main="",cex.lab=1.5,theta=30,phi=20,r=50,d=0.1,expand=0.5,
      ltheta=90,lphi=180,shade=0.75,ticktype="detailed",nticks=5,
      xlim=c(-3,3),ylim=c(-3,3),zlim=c(0,0.2))
contour(x1,x2,z,asp = 1)
```



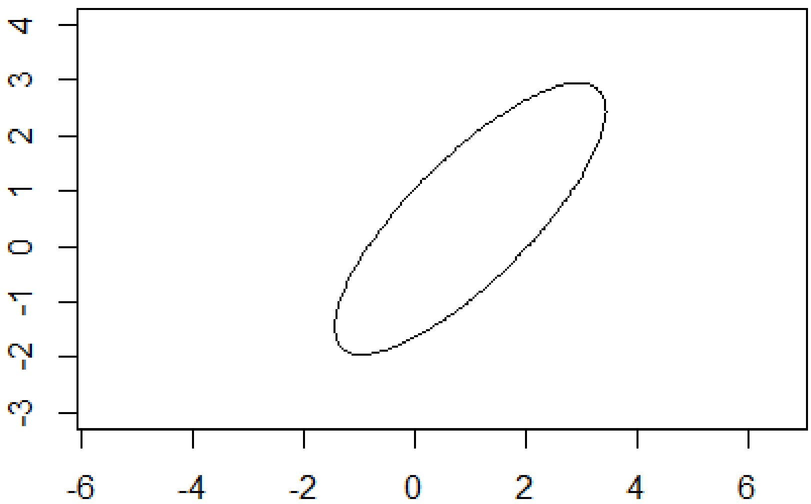


```
### Exercise 3

remove(list = ls())

m = c(1,0,-1,2)
m

## [1]  1  0 -1  2

s = c(1,0.2,0.4,-0.5,0.2,2,0.8,0,0.4,0.8,2,0,-0.5,0,0,1)
S = matrix(s,4,4)
S

##      [,1] [,2] [,3] [,4]
## [1,]  1.0  0.2  0.4 -0.5
## [2,]  0.2  2.0  0.8  0.0
## [3,]  0.4  0.8  2.0  0.0
## [4,] -0.5  0.0  0.0  1.0

COR = cov2cor(S)
round(COR, digits = 4)

##         [,1]   [,2]   [,3] [,4]
## [1,]  1.0000 0.1414 0.2828 -0.5
## [2,]  0.1414 1.0000 0.4000  0.0
## [3,]  0.2828 0.4000 1.0000  0.0
## [4,] -0.5000 0.0000 0.0000  1.0

### X3 und X4 sind unkorreliert, Die beiden paare X1 X2, X2 X3 sind korreli
ert.

# b)

#  i)

m1 = m[3]
S1 = S[3,3]

paste("X3 ~ N(",m1,",",S1,")")

## [1] "X3 ~ N( -1 , 2 )"
```

```
### Exercise 4

remove(list = ls())

e = c(1,0,-1,2)
e

## [1]  1  0 -1  2

s = c(1,0.2,0.4,-0.5,0.2,2,0.8,0,0.4,0.8,2,0,-0.5,0,0,1)
S = matrix(s,4,4)
S

##      [,1] [,2] [,3] [,4]
## [1,]  1.0  0.2  0.4 -0.5
## [2,]  0.2  2.0  0.8  0.0
## [3,]  0.4  0.8  2.0  0.0
## [4,] -0.5  0.0  0.0  1.0

x1 = c(0,0,0,0)
x2 = c(1,1,1,1)
x3 = c(1,0,1,0)

# a)
#  i)

denx1 = (1/(((2*pi)^(4/2))*sqrt(det(S))))*exp((-0.5)*(t(x1-e)%*%solve(S)%*%
(x1-e)))
denx1

##              [,1]
## [1,] 4.705344e-05

#  ii)

denx2 = (1/(((2*pi)^(4/2))*sqrt(det(S))))*exp((-0.5)*(t(x2-e)%*%solve(S)%*%
(x2-e)))
denx2

##            [,1]
## [1,] 0.00202044

#  iii)

denx3 = (1/(((2*pi)^(4/2))*sqrt(det(S))))*exp((-0.5)*(t(x3-e)%*%solve(S)%*%
(x3-e)))
denx3

##              [,1]
## [1,] 0.0001671811

# b)

#  i)

(1/(((2*pi)^(4/2))*sqrt(det(S))))*exp((-0.5)*(qchisq(0.95,4)))
```

```
## [1] 0.0001470645

#  ii)

(1/(((2*pi)^(4/2))*sqrt(det(S))))*exp((-0.5)*(qchisq(0.9,4)))

## [1] 0.000345508

#  iii)

(1/(((2*pi)^(4/2))*sqrt(det(S))))*exp((-0.5)*(qchisq(0.8,4)))

## [1] 0.0008459224
```

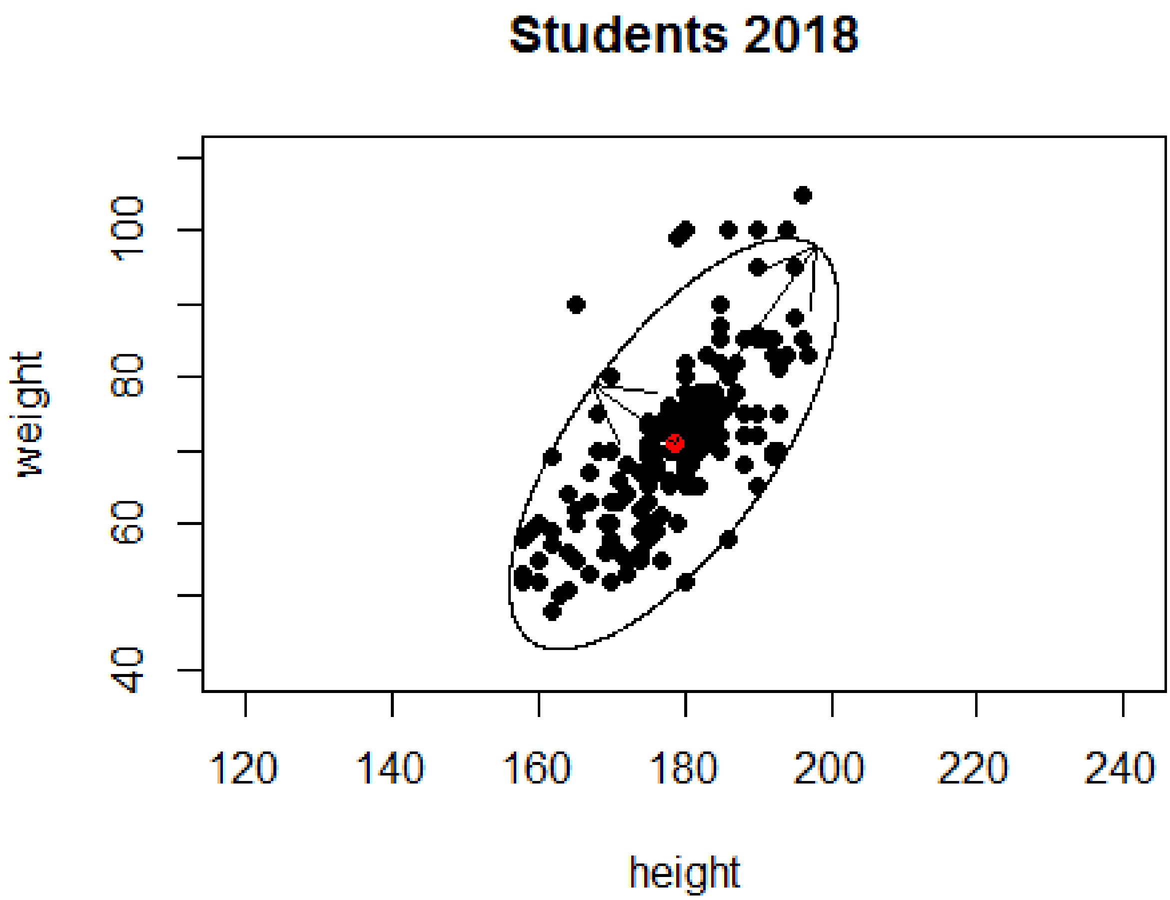
```
### Exercise 5

remove(list = ls())

m = c(1,0.5)
m1 = 1
m2 = 0.5
s = matrix(c(1,0.8,0.8,1),2,2)
S = solve(s)

# a)

x1 = seq(-3,4,le=100)
x2 = x1

f= function(v1,v2)
{
 S[1,1]*(v1-m1)^2+S[2,2]*(v2-m2)^2+2*S[2,1]*(v1-m1)*(v2-m2)
}

c = qchisq(0.95,2)
z = outer(x1,x2,f)
contour(x1,x2, z, levels = c, asp = 1, drawlabels = FALSE)
```



```
paste("Ervery point on this contour exhibit is lying on on the density valu
e", dens1)

## [1] "Ervery point on this contour exhibit is lying on on the density val
ue 0.00803"

# c)

L1 = sqrt(sum(eigen(s)$values[1]^2))
L1

## [1] 1.8

L2 = sqrt(sum(eigen(s)$values[2]^2))
L2

## [1] 0.2
```

```
### Exercise 7

remove(list = ls())

getwd()

## [1] "C:/Users/B-C-Herbert/Documents/Studium/Mannheim/VWL/2019 - 2020 WS/
Applied Multivariate Statistics/Assignments/Assignment3/Assignment3"

students  = read.table(file = "students2008.txt", header = T, dec =",")
attach(students)

## The following object is masked from package:datasets:
##
##     sleep

hw = data.frame(height,weight)
heigthweight = na.omit(hw)
detach(students)
attach(heigthweight)

# a)

X = cbind(height,weight)

m1 = mean(height)
m2 = mean(weight)
m = c(m1,m2)
s = cov(X)
S = solve(s)

x1 = seq(140,220, le = 1000)
x2 = seq(40,110, le = 1000)

f = function(v1,v2)
```

```
{
  S[1,1]*(v1-m1)^2+S[2,2]*(v2-m2)^2+2*S[2,1]*(v1-m1)*(v2-m2)
}

resouter = outer(x1,x2,f)
contour(x1,x2, resouter, levels = qchisq(0.95,2), asp = 1, drawlabels = FAL
SE, main = "Students 2018", xlab = "height", ylab = "weight")

# b)

points(height,weight, pch =  16)
points(m1,m2,pch=16,col = "red")

c = sqrt(qchisq(0.95,2))
arrows(m1,m2,m1+eigen(S)$vectors[1,1]*c/sqrt(eigen(S)$values[1]),
       m2+eigen(S)$vectors[2,1]*c/sqrt(eigen(S)$values[1]))
arrows(m1,m2,m1-eigen(S)$vectors[1,2]*c/sqrt(eigen(S)$values[2]),
       m2-eigen(S)$vectors[2,2]*c/sqrt(eigen(S)$values[2]))
```



```
e11 = eigen(S)$vectors[1,1]*c/sqrt(eigen(S)$values[1])
e12 = eigen(S)$vectors[2,1]*c/sqrt(eigen(S)$values[1])
e21 = eigen(S)$vectors[1,2]*c/sqrt(eigen(S)$values[2])
e22 = eigen(S)$vectors[2,2]*c/sqrt(eigen(S)$values[2])

L1 = sqrt(e11^2+e12^2)
L1

## [1] 13.60434

L2 = sqrt(e21^2+e22^2)
L2

## [1] 33.26774

# c)

?mahalanobis

## starting httpd help server ...

##  done

dis = mahalanobis(heigthweight,m,solve(S))
dm = dim(heigthweight)[1]
c = sqrt(qchisq(0.95,2))
sum((sqrt(dis))<c)/dm

## [1] 0.9433962
```

94,34% der Punkte liegen in der Kontur.