MATH534 HW1 PT1

William Gipson

2024-01-27

```r
library(rgl)
```

Problem 1:

Let

$$f(x_1, x_2) = \cos(x_1 x_2).$$

Then, its second order Taylor approximation is

$$h(x_1, x_2) = 1 - \frac{\pi}{8} x_1^2.$$

i.

```r
f <- function(x1, x2){cos(x1*x2)}
```

ii.

```r
h <- function(x1, x2){1 - (pi**2/8)*x1**2}
```

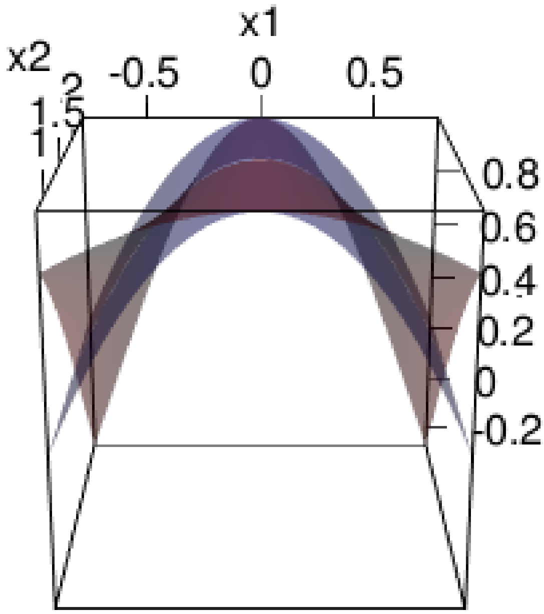
iii.

```r
x1 <- seq(-pi/4, pi/4, (pi/4 + pi/4)/29)
x2 <- seq(pi/4, 3*pi/4, (pi/4 + pi/4)/29)

f <- function(x1, x2){cos(x1*x2)}
f <- outer(x1, x2, FUN = f)
persp3d(x1, x2, f, col = "red",shade = 0.1, alpha = 0.5)

h <- function(x1, x2){1 - (pi**2/8)*x1**2}
h <- outer(x1, x2, FUN = h)
persp3d(x1, x2, h, col = "blue", add = TRUE, alpha = 0.5)
rglwidget(controllers = )
```

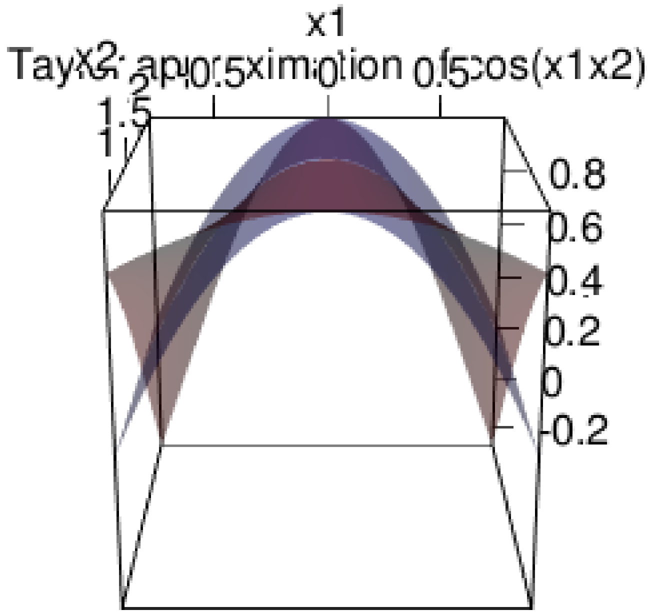
```
## Warning in snapshot3d(scene = x, width = width, height = height): webshot =
## TRUE requires the webshot2 package and Chrome browser; using rgl.snapshot()
## instead
```



iv.

```
persp3d(x1, x2, h, col = "blue", add = TRUE, alpha = 0.5)
title3d("Taylor approximation of cos(x1x2)")
rglwidget(controllers = )
```

```
## Warning in snapshot3d(scene = x, width = width, height = height): webshot =
## TRUE requires the webshot2 package and Chrome browser; using rgl.snapshot()
## instead
```

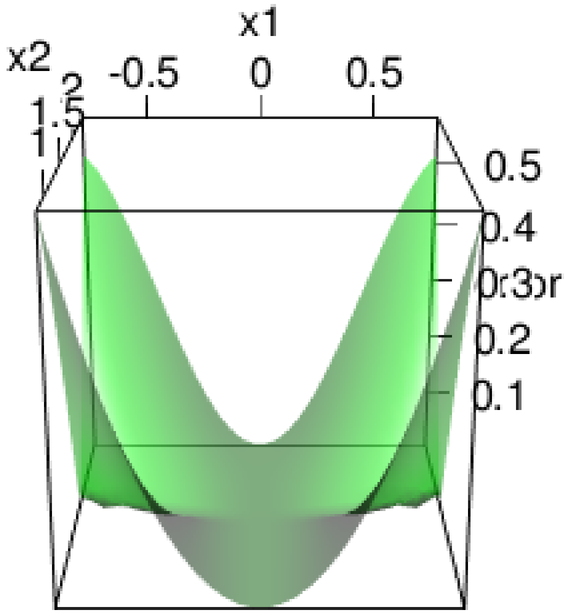


v.

```
error <- abs(f - h)

persp3d(x1, x2, error, col = "green", alpha = 0.5)
rglwidget(controllers = )
```

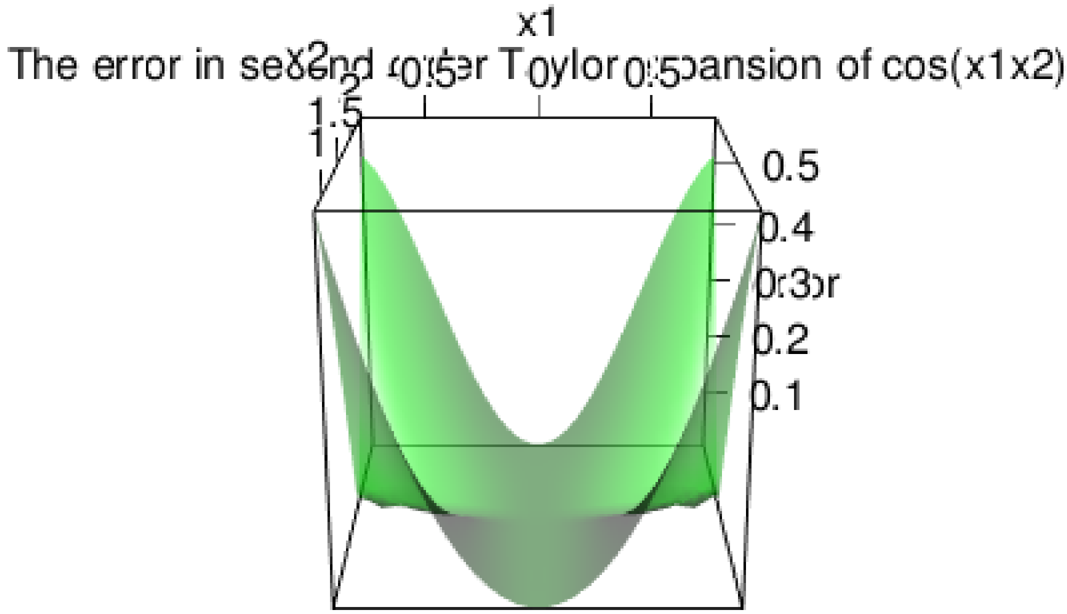
```
## Warning in snapshot3d(scene = x, width = width, height = height): webshot =
## TRUE requires the webshot2 package and Chrome browser; using rgl.snapshot()
## instead
```



vi.

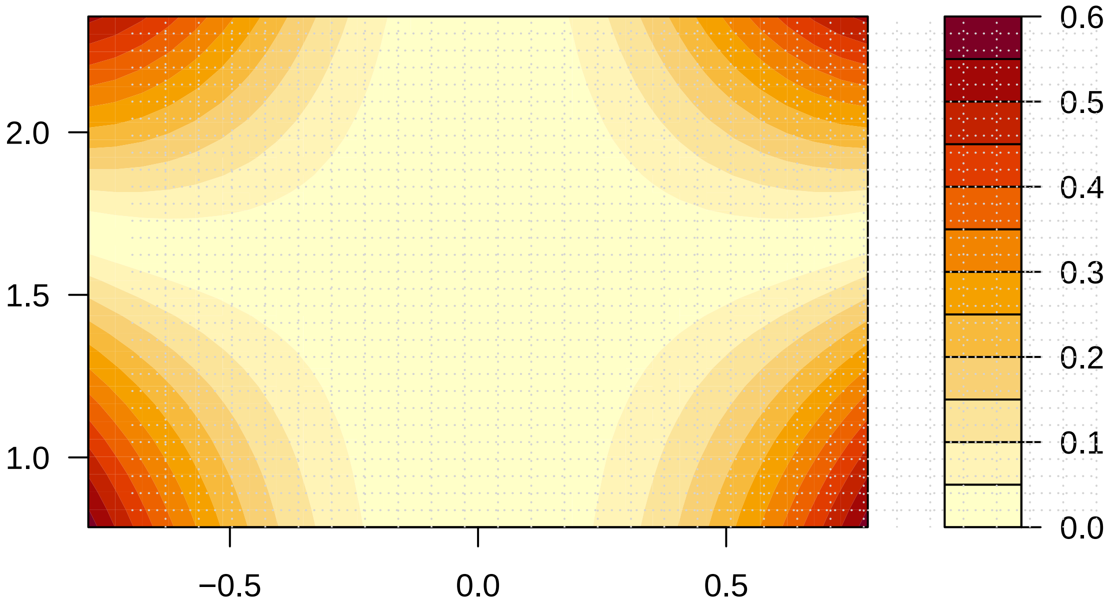
```r
persp3d(x1, x2, error, col = "green", alpha = 0.5)
title3d("The error in second order Taylor expansion of cos(x1x2)")
rglwidget(controllers = )
```

```
## Warning in snapshot3d(scene = x, width = width, height = height): webshot =
## TRUE requires the webshot2 package and Chrome browser; using rgl.snapshot()
## instead
```



vii.

```r
filled.contour(x1, x2, error)
grid(nx = 30, ny = 30)
```



From the plot of the error function, we observe that the magnitude of the error is larger at the edges and and goes to zero at the center. This makes sense in the context of the plot of both f and h where they meet in the center and mathematically since

$$f(0, \frac{\pi}{2}) = 1 = 1 - \frac{\pi^2}{8} * 0^2 = h(0, \frac{\pi}{2}).$$

Problem 2:

Given a p x 1 vector $\boldsymbol{\mu}$ and a p x p positive definite matrix Σ, the pdf for a p-variate normal density at a point $\boldsymbol{x} = (x_1, x_2, \ldots, x_p)^T$ is given by

$f(\boldsymbol{x}) = (2\pi)^{-\frac{p}{2}}|\Sigma|^{-\frac{1}{2}}exp[-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})].$

Now consider the bivariate normal random variable where

$\boldsymbol{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \boldsymbol{\mu} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \Sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{pmatrix}.$

a. Consider the point $\boldsymbol{x_0} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$. Then the second order Taylor expansion for $f(\boldsymbol{x})$ for the bivariate normal density around the point $\boldsymbol{x_0}$ is

$f(\boldsymbol{x}) = \frac{1}{2\pi\sqrt{\sigma_{11}\sigma_{22}-\sigma_{12}\sigma_{21}}} + \frac{-\sigma_{22}(x_1-\mu_1)^2-\sigma_{11}(x_2-\mu_2)^2+(x_1-\mu_1)(x_2-\mu_2)(\sigma_{21}+\sigma_{12})}{4\pi(\sigma_{11}\sigma_{22}-\sigma_{12}\sigma_{21})^{3/2}}.$

b.

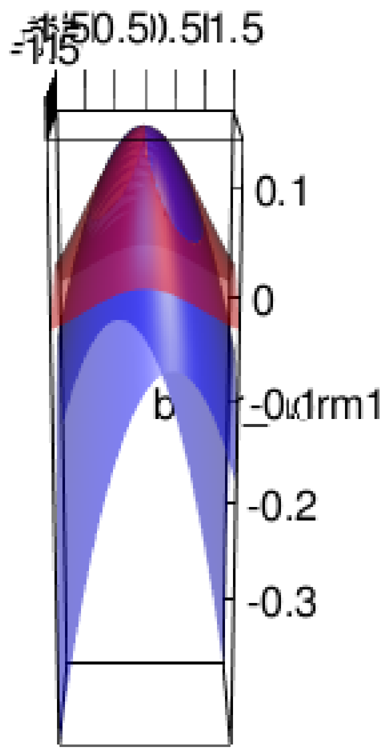
i.

```
x1 <- seq(-1.5, 1.5, 0.1)
x2 <- seq(-1.5, 1.5, 0.1)

bivar_norm1 <- function(x1,
                        x2){
  1/(2*pi*sqrt(1*1 - (-0.3)*(-0.3))) *
       exp(-(1/(2*(1*1 -
                 (-0.3)*(-0.3))))*((x1 - 0)**2*1 -
                                   (x1 - 0)*(x2 - 0)*((-0.3) -
                                                      (-0.3)) +
                                   (x2 - 0)**2*1))
}
bivar_norm1 <- outer(x1, x2, FUN = bivar_norm1)
persp3d(x1, x2, bivar_norm1, col = "red", shade = 0.1, alpha = 0.5)

bivar_norm_2nd_taylor_expansion1 <- function(x1,
                        x2){
   1/(2*pi*sqrt(1*1 - (-0.3)*(-0.3))) +
    (((-1*x1**2 - 1*x2**2 + x1*x2*(-0.3 + -0.3)))/(4*pi*(1*1 - (-0.3)*(-0.3))**(1.5)))
}
bivar_norm_2nd_taylor_expansion1 <- outer(x1,
                                          x2,
                                          FUN = bivar_norm_2nd_taylor_expansion1)
persp3d(x1, x2, bivar_norm_2nd_taylor_expansion1, col = "blue", add = TRUE, alpha = 0.5)
rglwidget(controllers = )
```

```
## Warning in snapshot3d(scene = x, width = width, height = height): webshot =
## TRUE requires the webshot2 package and Chrome browser; using rgl.snapshot()
## instead
```



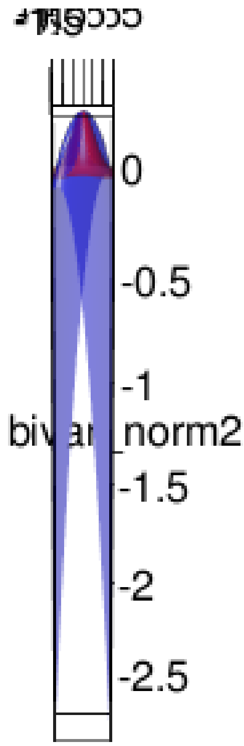
ii.

```r
x1 <- seq(-1.5, 1.5, 0.1)
x2 <- seq(-1.5, 1.5, 0.1)

bivar_norm2 <- function(x1,
                        x2){
  1/(2*pi*sqrt(1*1 - (0.8)*(0.8))) *
       exp(-(1/(2*(1*1 -
                     (0.8)*(0.8))))*((x1 - 0)**2*1 -
                                       (x1 - 0)*(x2 - 0)*((0.8) - (0.8)) +
                                       (x2 - 0)**2*1))
}
bivar_norm2 <- outer(x1, x2, FUN = bivar_norm2)
persp3d(x1, x2, bivar_norm2, col = "red", shade = 0.1, alpha = 0.5)

bivar_norm_2nd_taylor_expansion2 <- function(x1,
                        x2){
   1/(2*pi*sqrt(1*1 - (0.8)*(0.8))) +
    (((-1*x1**2 - 1*x2**2 + x1*x2*(0.8 + 0.8)))/(4*pi*(1*1 - (0.8)*(0.8))**(1.5)))
}
bivar_norm_2nd_taylor_expansion2 <- outer(x1,
                                          x2,
                                          FUN = bivar_norm_2nd_taylor_expansion2)
persp3d(x1, x2, bivar_norm_2nd_taylor_expansion2, col = "blue", add = TRUE, alpha = 0.5)
rglwidget(controllers = )
```

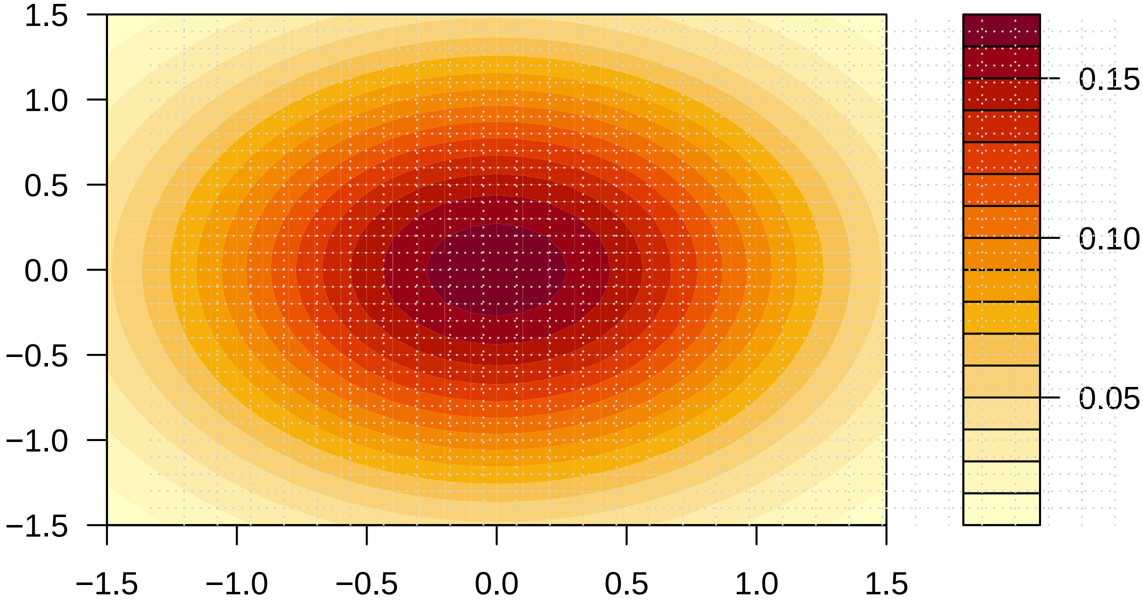
```
## Warning in snapshot3d(scene = x, width = width, height = height): webshot =
## TRUE requires the webshot2 package and Chrome browser; using rgl.snapshot()
## instead
```



c.

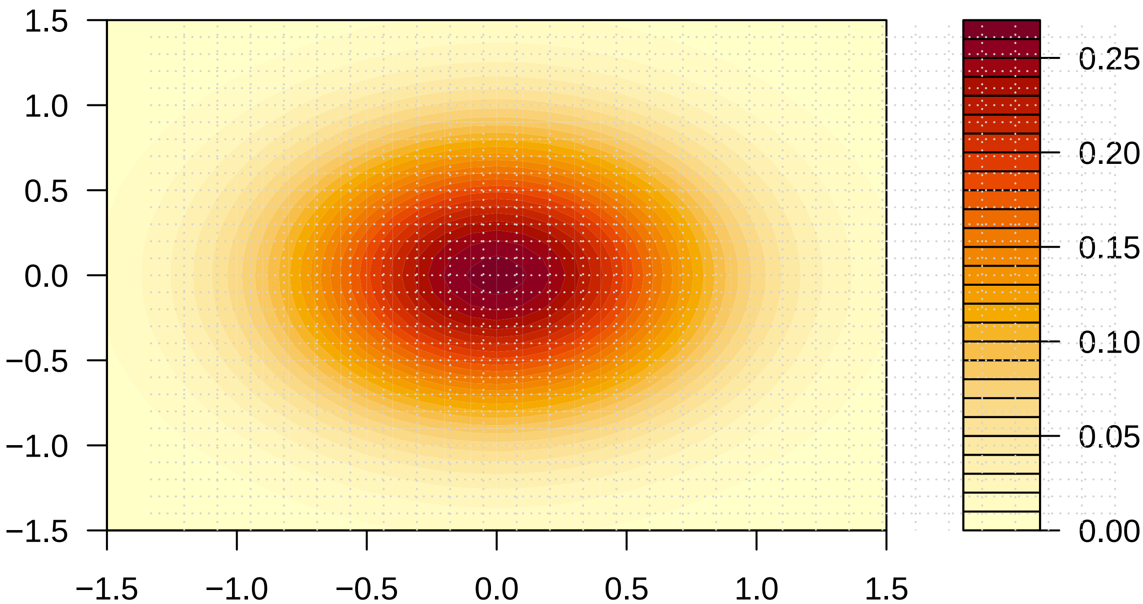
i.

```r
filled.contour(x1, x2, bivar_norm1)
grid(nx = 30, ny = 30)
```



ii.

```r
filled.contour(x1, x2, bivar_norm2)
grid(nx = 30, ny = 30)
```



From the constant value contour graphs in i. and ii., the shapes shows ellipses with the highest values being in the center. For both i. and ii., the center of the constant value contours is at $(x_1, x_2)^T = (0, 0)^T$.

d.

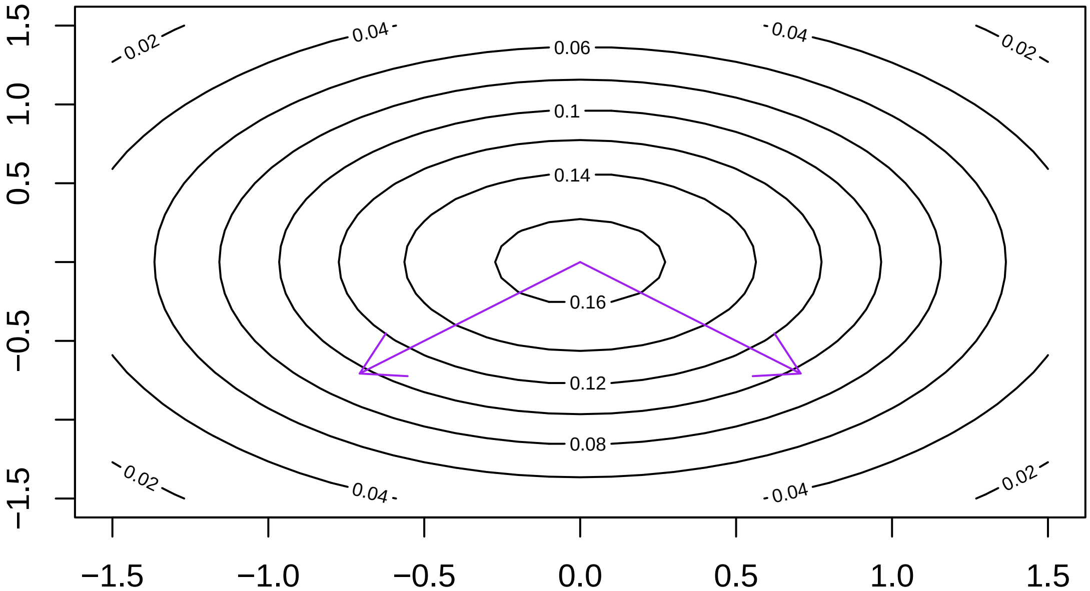
For $\Sigma = \begin{pmatrix} 1 & -0.3 \\ -0.3 & 1 \end{pmatrix}$, the eigenvalues are $\lambda = 1.3, 0.7$. For $\lambda = 1.3$, the eigenvectors are $\{\begin{pmatrix} x \\ -x \end{pmatrix} \ |x \in \mathbb{R}\}$ and for $\lambda = 0.7$, the eigenvectors are $\{\begin{pmatrix} x \\ x \end{pmatrix} \ |x \in \mathbb{R}\}$. For $\Sigma = \begin{pmatrix} 1 & 0.8 \\ 0.8 & 1 \end{pmatrix}$, the eigenvalues are $\lambda = 0.2, 1.8$. For $\lambda = 0.2$, the eigenvectors are $\{\begin{pmatrix} x \\ -x \end{pmatrix} \ |x \in \mathbb{R}\}$ and for $\lambda = 1.8$, the eigenvectors are $\{\begin{pmatrix} x \\ x \end{pmatrix} \ |x \in \mathbb{R}\}$.

i.

```r
sigma1 = matrix(c(1, -0.3, -0.3, 1), nrow = 2, ncol = 2, byrow = TRUE)

contour(x1, x2, bivar_norm1)

arrows(0, 0, eigen(sigma1)$vectors[1, 1], eigen(sigma1)$vectors[2, 2], col = "purple")
arrows(0, 0, eigen(sigma1)$vectors[2, 1], eigen(sigma1)$vectors[1, 2], col = "purple")
```

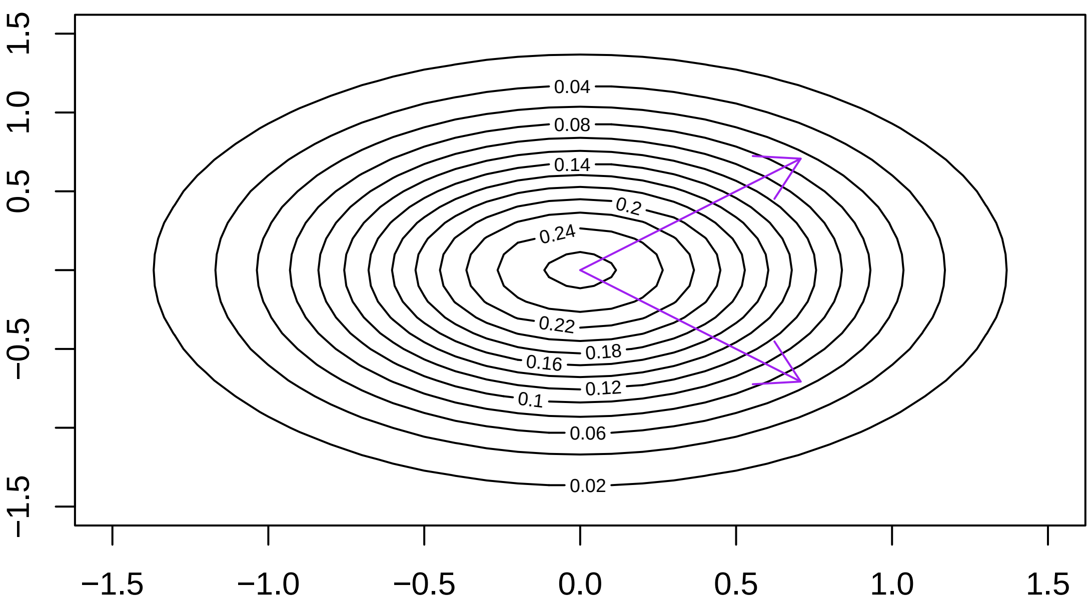


ii.

```r
sigma2 = matrix(c(1, 0.8, 0.8, 1), nrow = 2, ncol = 2, byrow = TRUE)

contour(x1, x2, bivar_norm2)

arrows(0, 0, eigen(sigma2)$vectors[1, 1], eigen(sigma2)$vectors[2, 2], col = "purple")
arrows(0, 0, eigen(sigma2)$vectors[2, 1], eigen(sigma2)$vectors[1, 2], col = "purple")
```



The eigenvectors and eigenvalues of the covariance matrix are related to the constant value contours because they determine the shape and orientation of the ellipse of the graph of the constant value contours.