Homework 1 - Part 1 - Math 534

Elijah Amirianfar

2024-01-26

Problem 1

Part i)

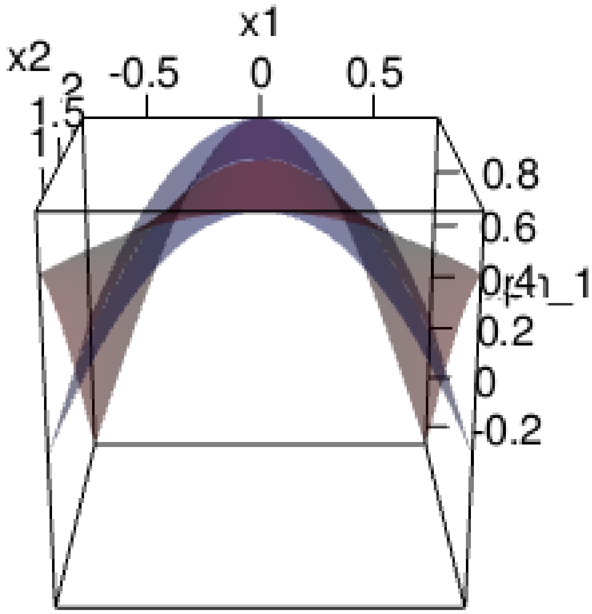
```
f_x <- function(x1, x2){cos(x1*x2)}
```

Part ii) Taylor Approximation

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

Part iii-iv)

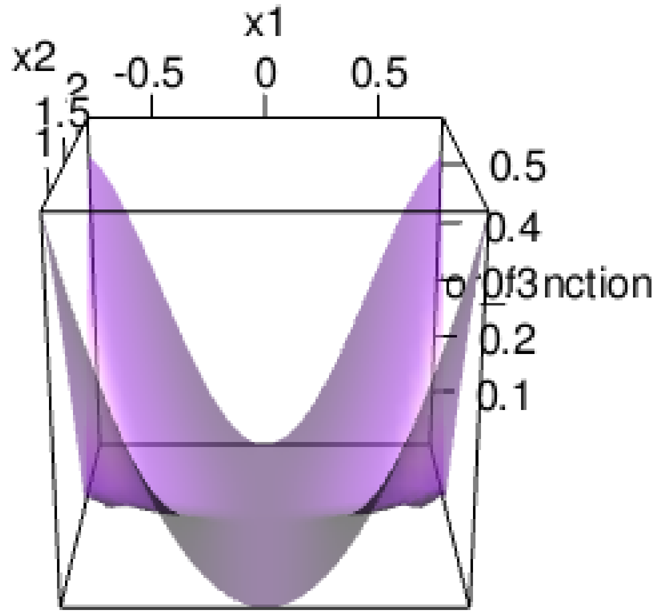
```
x1 = seq(-pi/4,pi/4, length=30)
x2 = seq(pi/4, 3*pi/4, length=30)
graph_1 <- outer(x1,x2, FUN = f_x)
graph_2 <- outer(x1,x2, FUN = h_x)
persp3d(x1, x2, graph_1, col = "red",shade = 0.1, alpha = 0.5, sub
        = "Taylor Approximation of cos(x1x2)")
persp3d(x1, x2, graph_2, col = "blue",shade = 0.1, alpha = 0.5, theta = 30, phi = 30, expand = 0.5, add
rglwidget(controllers = )
```



Taylor Approximation of cos(x1x2)

Part v-vi)

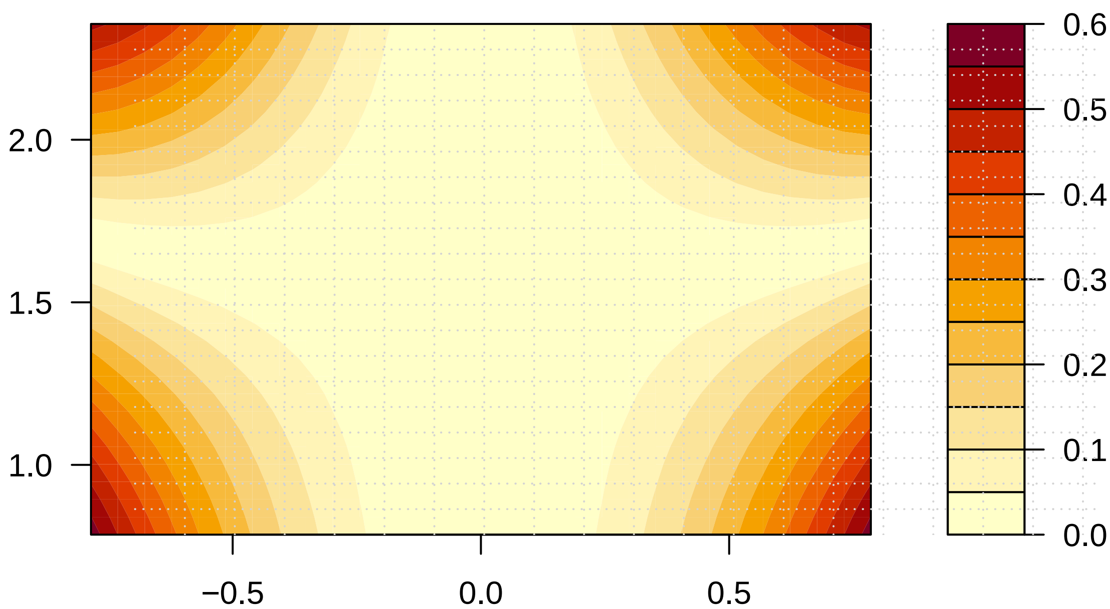
```
error_function <- abs(graph_1 - graph_2)
persp3d(x1,x2,error_function,col="purple",shade=0.1, alpha=0.5, sub =
          "The error in second order Taylor expansion of cos(x1x2)")
rglwidget(controllers = )
```



The error in second order Taylor expansion of cos(x1x2)

Part vii)

```r
filled.contour(x1, x2, error_function)
grid(nx = 20, ny = 20)
```



Here, we can see that the error is minimal when $x_1 = [-0.25, 0.25]$. Once we leave that interval, the error grows larger. Plus, when we look at error related to x_2, we see that error is minimal around 1.7.

Problem 2

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

$$f(\mathbf{x}) = (2\pi)^{-\frac{p}{2}} |\Sigma|^{-1/2} exp\left[-\frac{1}{2}(\mathbf{x}-\mu)^T \Sigma^{-1}(\mathbf{x}-\mu)\right]$$

Now consider the bivariate normal random variable where

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

Part a) Write the 2nd order Taylor expansion for f(x), for the bivariate normal density around the point

$$\mathbf{x_0} = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$$

Here, let $p = 2$. Based on what we saw in lecture,

$$f(\mathbf{x}) \cong f(\mathbf{x_0}) + (\mathbf{x}-\mathbf{x_0})^T \nabla f(\mathbf{x_0}) + \frac{1}{2}(\mathbf{x}-\mathbf{x_0})^T H(\mathbf{x_0})(\mathbf{x}-\mathbf{x_0}) + R$$

where R is the following formula: $R = O(||\mathbf{x}-\mathbf{x_0}||^3)$

Thus, $f(\mathbf{x_0})$ is the following:

$$\frac{1}{2\pi\sqrt{|\Sigma|}} exp\left(-\frac{1}{2}(\mu_1-\mu_1,\ \mu_2-\mu_2)\Sigma^{-1}\left(\begin{array}{c}\mu_1-\mu_1\\ \mu_2-\mu_2\end{array}\right)\right) = \frac{1}{2\pi\sqrt{|\Sigma|}}exp(0) = \frac{1}{2\pi\sqrt{\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12}}}$$

When we compute $\nabla f(\underline{x})$ and $\nabla^2 f(\underline{x})$, we get the following:

$$\nabla f(\underline{x}) = \left(\begin{array}{c}\frac{\partial f(\underline{x})}{\partial x_1}\\ \frac{\partial f(\underline{x})}{\partial x_2}\end{array}\right) = \left(\begin{array}{c}0\\ 0\end{array}\right)$$

$$\nabla^2 f(\underline{x}) = \left(\begin{array}{cc}\frac{\partial^2 f(\underline{x})}{\partial x_1\partial x_1} & \frac{\partial^2 f(\underline{x})}{\partial x_1\partial x_2}\\ \frac{\partial^2 f(\underline{x})}{\partial x_2\partial x_1} & \frac{\partial^2 f(\underline{x})}{\partial x_2\partial x_2}\end{array}\right) = \left(\begin{array}{cc}\frac{-\sigma_{22}}{2\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}} & \frac{\sigma_{21}+\sigma_{12}}{4\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}}\\ \frac{\sigma_{21}+\sigma_{12}}{4\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}} & \frac{-\sigma_{11}}{2\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}}\end{array}\right)$$

Now, we can write out the full 2nd order Taylor Expansion for f(x).

$$f(\mathbf{x}) = \frac{1}{2\pi\sqrt{\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12}}} + \frac{1}{2}(x_1-\mu_1,\ x_2-\mu_2)\left(\begin{array}{cc}\frac{-\sigma_{22}}{2\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}} & \frac{\sigma_{21}+\sigma_{12}}{4\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}}\\ \frac{\sigma_{21}+\sigma_{12}}{4\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}} & \frac{-\sigma_{11}}{2\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}}\end{array}\right)\left(\begin{array}{c}x_1-\mu_1\\ x_2-\mu_2\end{array}\right)$$

$$= \frac{1}{2\pi\sqrt{\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12}}} + \frac{1}{2}\left[\ \frac{-(x_1-\mu_1)^2\sigma_{22}}{2\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}} + \frac{(x_1-\mu_1)(x_2-\mu_2)(\sigma_{21}+\sigma_{12})}{2\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}} + \frac{-\sigma_{11}(x_2-\mu_2)^2}{2\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}}\ \right]$$

$$= \frac{1}{2\pi\sqrt{\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12}}} + \frac{1}{2}\left[\ \frac{-(x_1-\mu_1)^2\sigma_{22}+(x_1-\mu_1)(x_2-\mu_2)(\sigma_{21}+\sigma_{12})-\sigma_{11}(x_2-\mu_2)^2}{2\pi(\sigma_{11}\sigma_{22}-\sigma_{21}\sigma_{12})^{3/2}}\ \right]$$

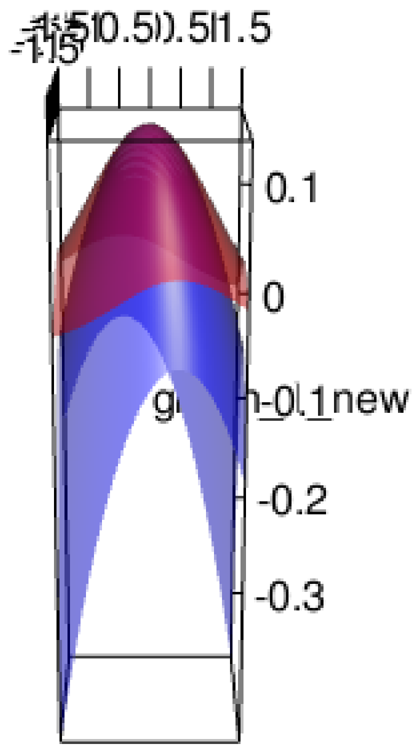
Part b)

i)

```r
new_f_x_sigma_1 <- function(x_1, x_2){((2*pi*sqrt(1 - 0.3^2))^{-1})*
    exp(-0.5*((x_1)^2+(0.6*x_1*x_2)+(x_2)^2)/0.91)}
new_f_x_sigma_1_approx <- function(x_1, x_2){(1/(2*pi*sqrt(1-0.3^2)))+
    0.5*(-x_1^2 + x_1*x_2*-0.6 - x_2^2)/(2*pi*(1-0.3^2)^(3/2))}

x1_new = seq(-1.5,1.5, length=30)
x2_new = seq(-1.5,1.5, length=30)
graph_1_new <- outer(x1_new,x2_new, FUN = new_f_x_sigma_1)
graph_2_new <- outer(x1_new,x2_new, FUN = new_f_x_sigma_1_approx)
persp3d(x1_new, x2_new, graph_1_new, col = "red",shade = 0.1, alpha = 0.5, sub
```

```
        = "Taylor Approximation of f(x1x2), first set of parameters")
persp3d(x1_new, x2_new, graph_2_new, col = "blue",shade = 0.1, alpha = 0.5, add=TRUE)
rglwidget(controllers = )
```

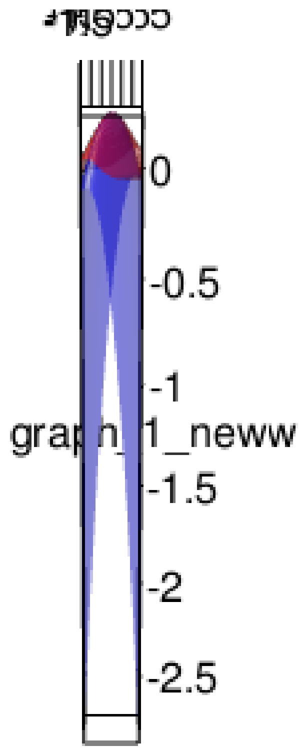


Taylor Approximation of f(x1x2), first set of parameters

ii)

```
new_f_x_sigma_2 <- function(x_1, x_2){(2*pi*sqrt(1 - 0.8^2))^{-1}*
    exp(-0.5*((x_1)^2-(1.6*x_1*x_2)+(x_2)^2)/0.36)}
new_f_x_sigma_2_approx <- function(x_1, x_2){(1/(2*pi*sqrt(1-0.8^2)))+
    0.5*(-x_1^2 + x_1*x_2*1.6 - x_2^2)/(2*pi*(1-0.8^2)^(3/2))}

x1_neww = seq(-1.5,1.5, length=30)
x2_neww = seq(-1.5,1.5, length=30)
graph_1_neww <- outer(x1_neww,x2_neww, FUN = new_f_x_sigma_2)
graph_2_neww <- outer(x1_neww,x2_neww, FUN = new_f_x_sigma_2_approx)
persp3d(x1_neww, x2_neww, graph_1_neww, col = "red",shade = 0.1, alpha = 0.5, sub
        = "Taylor Approximation of f(x1x2), second set of parameters")
persp3d(x1_neww, x2_neww, graph_2_neww, col = "blue",shade = 0.1, alpha = 0.5, add=TRUE)
rglwidget(controllers = )
```

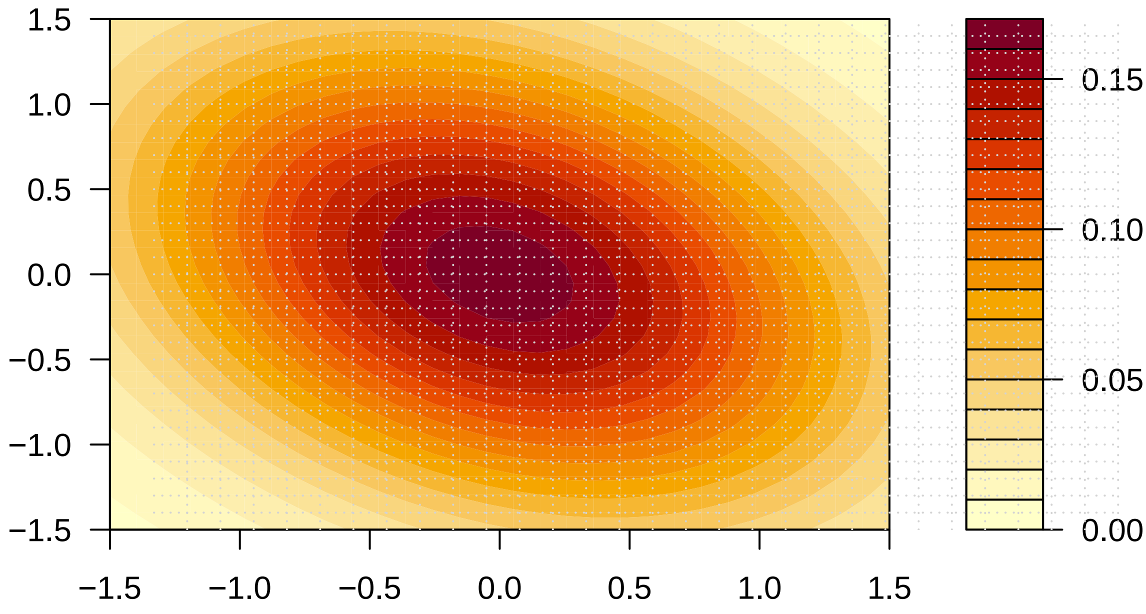


Taylor Approximation of f(x1x2), second set of parameters

Part c) Graph the constant value contours for f(x) for the cases (i) and (ii) in part b. What is the shape of the constant value contours? What is the center of the constant value contours?

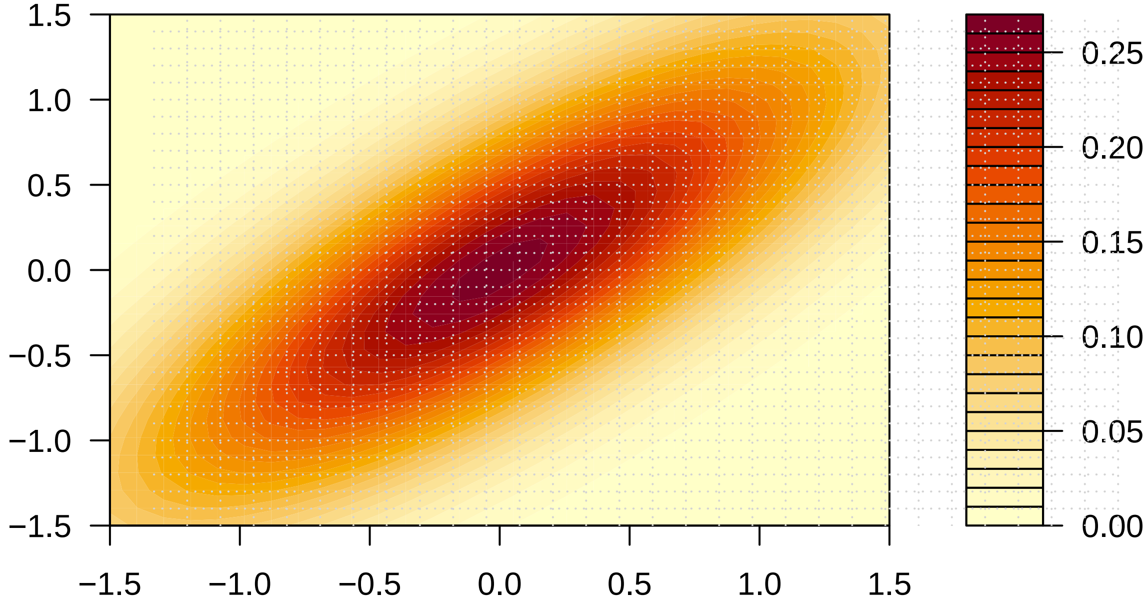
Case 1

```
filled.contour(x1_new, x2_new, graph_1_new)
grid(nx = 30, ny = 30)
```



Case 2

```
filled.contour(x1_neww, x2_neww, graph_1_neww)
grid(nx = 30, ny = 30)
```



In both cases, the shape of the constant value contours are ellipses in a diagonal angle. The centers for both contours are (0,0).

Part d) Compute the eigenvalues and eigenvectors for each of the covariance matrices Σ given in part (b). Superimpose the eigenvectors on each of their corresponding constant value contours that you drew in part (c) and explain how the eigenvectors and eigenvalues are related to the constant value contours.

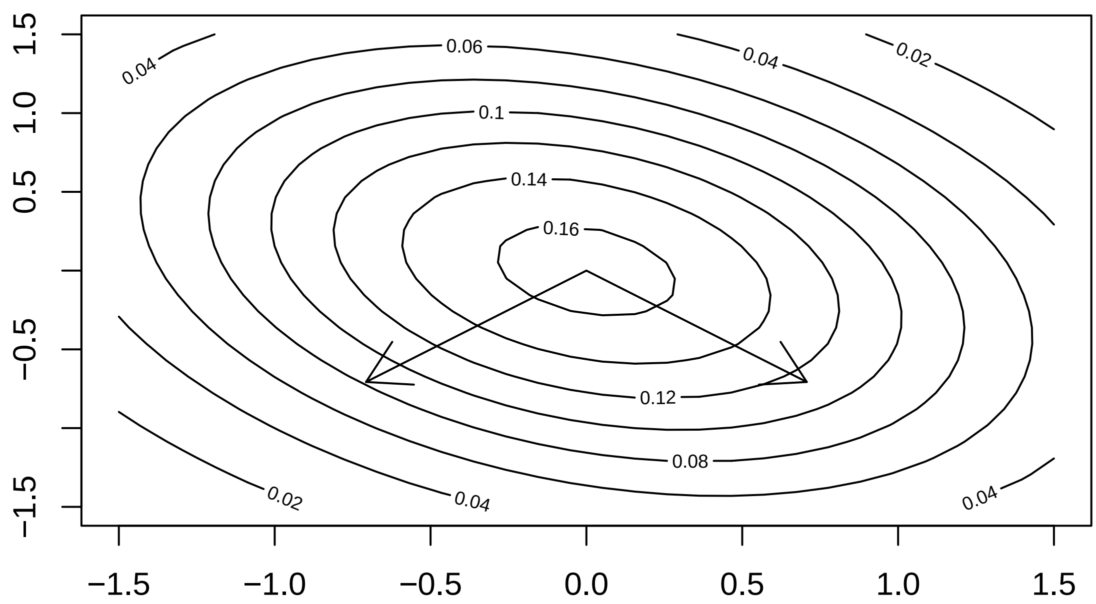
For case (i), the eigenvalues and eigenvectors are

```r
sigma_1 = matrix(c(1,-0.3,-0.3,1),2,2)
eigen(sigma_1)
```

```
## eigen() decomposition
## $values
## [1] 1.3 0.7
##
## $vectors
##            [,1]       [,2]
## [1,] -0.7071068 -0.7071068
## [2,]  0.7071068 -0.7071068
```

Our contour plot is below:

```r
contour(x1_new, x2_new, graph_1_new)
arrows(0,0, eigen(sigma_1)$vectors[1, 1], eigen(sigma_1)$vectors[2, 2])
arrows(0,0, eigen(sigma_1)$vectors[2, 1], eigen(sigma_1)$vectors[1, 2])
```



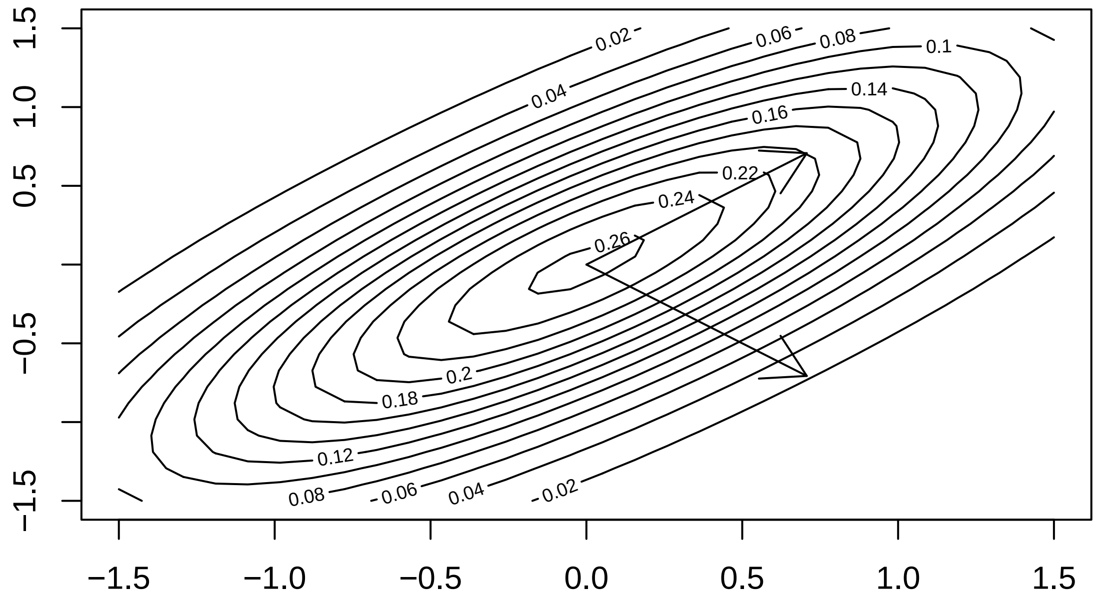
For case (ii), the eigenvalues and eigenvectors are

```r
sigma_2 = matrix(c(1,0.8,0.8,1),2,2)
eigen(sigma_2)
```

```
## eigen() decomposition
## $values
## [1] 1.8 0.2
##
## $vectors
##           [,1]       [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068
```

Our contour plot is below:

```r
contour(x1_neww, x2_neww, graph_1_neww)
arrows(0,0, eigen(sigma_2)$vectors[1, 1], eigen(sigma_2)$vectors[2, 2])
arrows(0,0, eigen(sigma_2)$vectors[2, 1], eigen(sigma_2)$vectors[1, 2])
```


0.02
0.04
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
0.22
0.24
0.26
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
−1.5
−0.5
0.5
1.0
1.5