537 Homework 1

Gustavo Esparza

7/5/2019

1

A

Here is our initial Covariance matrix:

$$\mathbf{\Sigma} = \begin{bmatrix} 25 & -2 & 4 \\ -2 & 4 & 1 \\ 4 & 1 & 9 \end{bmatrix}$$

In order to find $V^{1/2}$, the standard deviation matrix, we will make a diagonal matrix containing the square root values of our variances from Σ.

$$\mathbf{V^{1/2}} = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

Now, in order to find ρ, we will use the following definition:

$\rho = (V^{1/2})^{-1}\Sigma(V^{1/2})^{-1}$

$$\rho = \begin{bmatrix} 1 & -.2 & .267 \\ -.2 & 1 & .167 \\ .267 & .167 & 1 \end{bmatrix}$$

B

To compute the correlation between X_1 and $\frac{1}{2}x_2 + \frac{1}{2}x_3$, we will use the following equation:

$$Corr(x_1, .5X_2 + .5X_3) = \frac{Cov(x_1, .5X_2 + .5X_3)}{\sqrt{Var(x_1)}\sqrt{Var(.5x_2 + .5x_3)}}$$

Then, we have $Cov(x_1, .5X_2 + .5X_3) = \frac{1}{2}Cov(X_1, X_2) + \frac{1}{2}Cov(X_1, X_3) = .5\sigma_{12} + .5\sigma_{13} = .5(-2) + .5(4) = 1$

For our variances, we have $\sqrt{Var(x_1)}\sqrt{Var(.5x_2 + .5x_3)} =$

$\sqrt{Var(x_1)}\sqrt{.25(Var(X_2) + Var(X_3) + 2Cov(X_2, X_3))} =$

$\sqrt{25}\sqrt{(.25 \times 4 + .25 \times 9 + .5)} = 5 \times \sqrt{3.75}$

Finally, we have
$Corr(x_1, .5x_2 + .5x_3) = .103$

2

If $\vec{x}$ is multivariate normal with $\vec{\mu} = (-1, 1)$ and $\sigma_{11} = \sigma_{22} = 1$, $cov(x_1, x_2) = 0$, we can see that our bivariate normal distribution is independent (since the correlation between X_1 and $X_2 = 0$) and can be expressed as the product of two univariate normal distributions. Therefore any cummulative value over our space can be defined by the product of two seperate cummulative values over x_1 and x_2, respectively. Using *pnorm* in *R*, we have the following result:

Thus, $F_{\vec{x}}(0, 0) = \int_{-\infty}^{0} N(1, 1) \int_{-\infty}^{0} N(-1, 1) \approx .1586 \times .8413 = .1335$

3

We have $f(\vec{x}) = (x_1 + x_2^2 + x_3)^2$ for $\vec{x} = (x_1, x_2, x_3, x_4)$

Thus, we have the following gradient of f with respect to $\vec{x}$:

$$G = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \frac{\partial f}{\partial x_3} \\ \frac{\partial f}{\partial x_4} \end{pmatrix} = \begin{pmatrix} 2(x_1 + x_2^2 + x_3) \\ 4x_2(x_1 + x_2^2 + x_3) \\ 2(x_1 + x_2^2 + x_3) \\ 0 \end{pmatrix}$$

Taking the second partial derivatives, we have the following result

$$H = \begin{pmatrix} \frac{\partial f^2}{\partial x_1 \partial x_1} & \frac{\partial f^2}{\partial x_1 \partial x_2} & \frac{\partial f^2}{\partial x_1 \partial x_3} & \frac{\partial f^2}{\partial x_1 \partial x_4} \\ \frac{\partial f^2}{\partial x_2 \partial x_1} & \frac{\partial f^2}{\partial x_2 \partial x_2} & \frac{\partial f^2}{\partial x_2 \partial x_3} & \frac{\partial f^2}{\partial x_2 \partial x_4} \\ \frac{\partial f^2}{\partial x_3 \partial x_1} & \frac{\partial f^2}{\partial x_3 \partial x_2} & \frac{\partial f^2}{\partial x_3 \partial x_3} & \frac{\partial f^2}{\partial x_3 \partial x_4} \\ \frac{\partial f^2}{\partial x_4 \partial x_1} & \frac{\partial f^2}{\partial x_4 \partial x_2} & \frac{\partial f^2}{\partial x_4 \partial x_3} & \frac{\partial f^2}{\partial x_4 \partial x_4} \end{pmatrix} = \begin{pmatrix} 2 & 4x_2 & 2 & 0 \\ 4x_2 & 4x_1 + 12x_2^2 + 4x_3 & 4x_2 & 0 \\ 2 & 4x_2 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

4

A

Here is our original Matrix:

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

For singular value decomposition, we have the following result:

$A = \Gamma\Lambda\Delta'$, where Γ is a matrix consisting of the eigenvectors of $A * A^T$, Λ is a diagonal matrix consisting of the eigenvalues of $A * A^t$ and Δ is a matrix consisting of the eigenvectors of $A^T * A$.

Thus, using built in R functions, we have the following results:

$$\Gamma = \begin{pmatrix} -4.472136e-01 & -5.551115e-17 \\ -1.110223e-16 & -1 \\ -8.944272e-01 & 0 \end{pmatrix}$$

$$\Lambda = \begin{pmatrix} 3.162278 & 0.000000 \\ 0.000000 & 2.828427 \end{pmatrix}$$

$$\Delta = \begin{pmatrix} -0.7071068 & -0.7071068 \\ -0.7071068 & 0.7071068 \end{pmatrix}$$

Note: Δ will be transposed for the actual diagonalization process. We can note that the multiplication of $\Gamma\Lambda\Delta'$ given above do indeed produce the original matrix A, as desired.

B

Here is our original matrix:

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

For diagonalization, we will use spectral decomposition. For this process, we have the following result:
$A = \Gamma\Lambda\Gamma'$, where Γ is a matrix consisting of the eigenvectors of A and Λ is a diagonal matrix consisting of the eigenvalues of A.
Thus, using built in R functions, we have the following results:

$A = \Gamma\Lambda\Gamma'$,where

$$\Gamma = \begin{pmatrix} 0.7121118 & -0.27071711 & 0.6477724 \\ -0.2440204 & -0.96058049 & -0.1331884 \\ 0.6582939 & -0.06322468 & -0.7501012 \end{pmatrix}$$

$$\Lambda = \begin{pmatrix} 5.6977 & 0 & 0 \\ 0 & 2.934181 & 0 \\ 0 & 0 & -2.631881 \end{pmatrix}$$

$$\Gamma' = \begin{pmatrix} 0.7121118 & -0.2440204 & 0.65829392 \\ -0.2707171 & -0.9605805 & -0.06322468 \\ 0.6477724 & -0.1331884 & -0.75010116 \end{pmatrix}$$

We can note that the multiplication of $\Gamma\Lambda\Gamma'$ given above do indeed produce the original matrix A, as desired.

5

A

Here are the r^2 values for our three linear models:

```
##  r-squared for  y ~ x1 + x2 =  0.4159225
##  r-squared for y ~ x1 =  0.2535442
##  r-squared for y ~ x2 =  0.3441131
```

B

Here is the covariance matrix for X_1 and X_2 from our dataset:

```
##           [,1]      [,2]
## [1,] 0.9475373 0.5281126
## [2,] 0.5281126 1.4499960
```

Here are the eigenvalues computed from our covariance matrix:

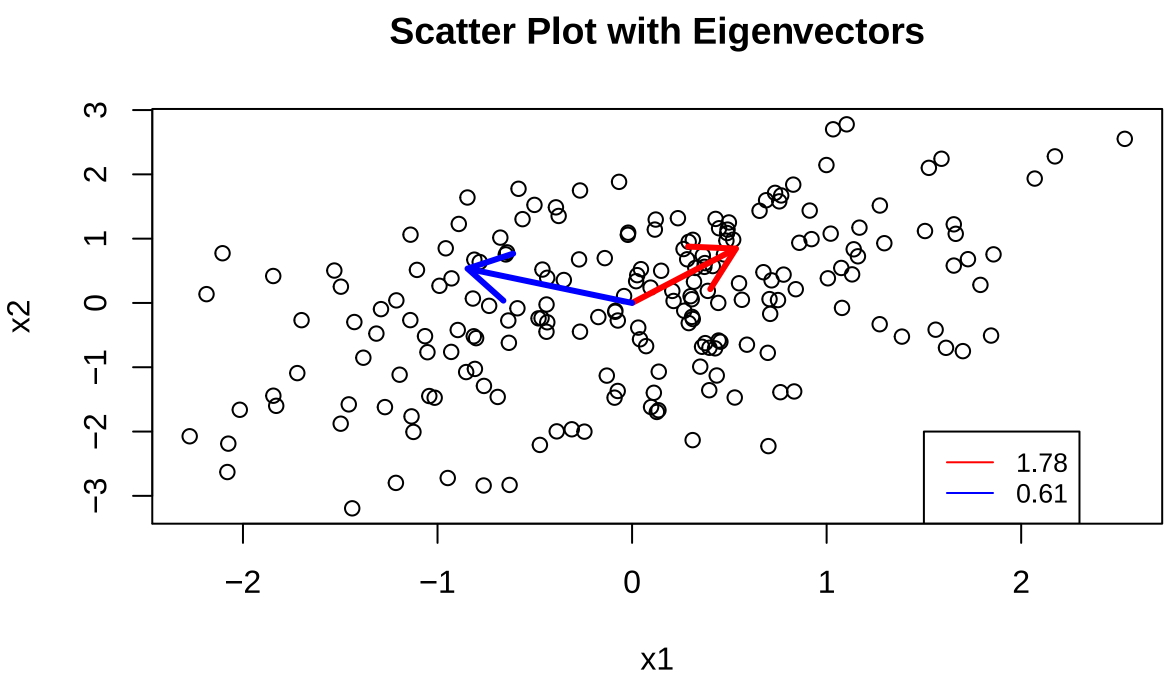
```
## Eigenvalues from covariance matrix:  1.783591   0.6139426
```

Here are the eigenvectors computed from our covariance matrix:

```
##           [,1]       [,2]
## [1,] 0.5340500 -0.8454529
## [2,] 0.8454529  0.5340500
```

C

Here is a plot of X_1 and X_2, along with the eigenvectors found for their respective covariance matrix:



D

By muliplying our X variables by our eigenvectors, we have created two new columns, C_1 and C_2. Using these C variables, we will make linear models and compute their respective r^2 values. Here are the results:

```
##  r-squared for y ~ c1 + c2 =  0.4159225
##  r-squared for y ~ c1 =  0.4136737
##  r-squared for y ~ c2 =  0.002248745
```

E

For the models consisting of only one predictor, $y \sim C_1$ has the highest r^2 value. This can be attributed to the eigenvectors shifting the influence of x_1 and x_2.

For the models consisting of two predictors, both $y \sim x_1 + x_2$ and $y \sim c_1 + c_2$ have the same r^2 value. This makes sense because we have not added any information to the model, only shifted the importance of the variables x_1 and x_2.

Appendix

1

A

```
Sigma = matrix(c(25,-2,4,-2,4,1,4,1,9),3,3)
pnorm(0, mean = 1, sd = 0)
```

```
V = diag(sqrt(diag(Sigma)))
```

```
Vinv = inv(V)
```

```
rho = Vinv%*%Sigma%*%Vinv
```

2

```
fx1 = pnorm(0,1,1)
fx2 = pnorm(0,-1,1)
fx1*fx2
```

4

A

Here is our original Matrix:

```
svd = svd(A,2,2)

Gamma0 = svd$u
Delta0 = svd$v
Lambda0 = diag(svd$d)
Gamma0 %*% Lambda0 %*% t(Delta0)
```

```
A_AT = A %*% AT
AT_A = AT %*% A
```

```
Gamma = eigen(A_AT)
#Non-zero Eigenvalues
eigendiag = diag(c(10,8))
eigendiag = sqrt(eigendiag)
#Non-zero Eigenvectors
eigenmatrix_gamma =Gamma$vectors[,1:2]
```

```
delta = eigen(AT_A,F)
eigenmatrix_delta = delta$vectors
```

```
eigenmatrix_gamma %*% eigendiag %*% t(eigenmatrix_delta)
```

B

Here is our original matrix:

```r
A2 = matrix(c(2,0,4,0,3,-1,4,-1,1),3,3)
A2
```

```r
svd(A2)
```

```r
G = eigen(A2)

eigenvalues = G$values
eigenvectors = G$vectors
Lambda = diag(eigenvalues)
```

```r
eigenvectors %*% Lambda %*% t(eigenvectors)
```

5

```r
data = read.csv("hw1.csv")
y =  data$y
x1 = data$x1
x2 = data$x2
data = data[,2:4]
dataMatrix = as.matrix(data)
```

A

Here are the r^2 values for our three linear models:

```r
i = lm(y~x1+x2)
ii=lm(y~x1)
iii = lm(y~x2)

one =summary(i)
two = summary(ii)
three = summary(iii)
```

```r
cat(" r-squared for  y ~ x1 + x2 = ",one$r.squared,
    "\n r-squared for y ~ x1 = ",two$r.squared,
    "\n r-squared for y ~ x2 = ",three$r.squared,"")
```

B

Here is the covariance matrix for X_1 and X_2 from our dataset:

```r
cov.matrix = matrix(c(var(x1),cov(x1,x2),cov(x1,x2),var(x2)),2,2)
cov.matrix
```

```r
eg = eigen(cov.matrix)
vectors = eg$vectors
values = eg$values
```

Here are the eigenvalues computed from our covariance matrix:

```r
cat("Eigenvalues from covariance matrix: ",values[1]," ",values[2],"")
```

Here are the eigenvectors computed from our covariance matrix:

```r
E = vectors
E
```

C

```r
ev1 =E[,1]
ev2 = E[,2]
```

Here is a plot of X_1 and X_2, along with the eigenvectors found for their respective covariance matrix:

```r
plot(x1,x2,main="Scatter Plot with Eigenvectors")
arrows(0,0,ev1[1],ev1[2],col="red",lwd=3)
arrows(0,0,ev2[1],ev2[2],col="blue",lwd=3)
legend(1.5, -2, legend=c("1.78"  , "0.61"),
       col=c("red", "blue"), lty=1, cex=0.8)
```

D

```r
X = cbind(x1,x2)
XE = X%*%E
```

```r
c1 = XE[,1]
c2 = XE[,2]
```

```r
iv = lm (y~c1+c2)

v = lm(y~c1)

vi = lm(y~c2)

four =summary(iv)
five = summary(v)
six = summary(vi)
```