Homework 2

Jeremy Maslanko

2024-02-22

Question 1

First, we define the matrix:

```
B <- matrix(c(1, 7, 8, 1, 3, 2, 6, 3, 1, 1, 4, 0, 2, 2, 1, -1),
            nrow = 4)
B
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    3    1    2
## [2,]    7    2    1    2
## [3,]    8    6    4    1
## [4,]    1    3    0   -1
```

Next, we define the function. This function will print the rank, SVD, determinant, and inverse of a matrix. This will only be done if the matrix is square and not singular.

```
linalg <- function(A) {

  dims <- dim(A)
  det <- det(A)

  if (det == 0) {
    print("Matrix is singular.")
  } else if (dims[1] != dims[2]) {
    print("Matix is not square")
  } else {
    rank <- qr(A)$rank
    svd <- svd(A)
    inv <- solve(A)

    cat("Rank of Matrix:\n")
    print(rank)
    cat("\n")

    cat("SVD of Matrix:\n")
    print(svd)
    cat("\n")

    cat("Det of Matrix:\n")
    print(det)
    cat("\n")

    cat("Inverse of Matrix:\n")
    print(inv)
  }
}
```

Finally, we see the results when we call the function:

```
linalg(B)
```

```
## Rank of Matrix:
## [1] 4
##
## SVD of Matrix:
## $d
## [1] 13.421740  3.688447  2.248213  1.482498
##
## $u
##            [,1]       [,2]       [,3]       [,4]
## [1,] -0.2210812  0.4114000  0.8724057  0.1441576
## [2,] -0.5323582 -0.6841250  0.1072497  0.4868934
## [3,] -0.8006332  0.2274669 -0.2265673 -0.5058779
## [4,] -0.1634144  0.5576574 -0.4196105  0.6973098
##
## $v
##            [,1]       [,2]        [,3]        [,4]
## [1,] -0.7835101 -0.5422562 -0.27087966  0.13672714
## [2,] -0.5231813  0.7872483  0.09495594  0.31225769
## [3,] -0.2947436  0.1727400  0.03264197 -0.93926651
## [4,] -0.1597481 -0.2374009  0.95736204  0.03973981
##
##
## Det of Matrix:
## [1] 165
##
## Inverse of Matrix:
##             [,1]        [,2]         [,3]        [,4]
## [1,] -0.13939394  0.16363636 -0.006060606  0.04242424
## [2,]  0.16363636 -0.01818182 -0.036363636  0.25454545
## [3,] -0.05454545 -0.32727273  0.345454545 -0.41818182
## [4,]  0.35151515  0.10909091 -0.115151515 -0.19393939
```

Question 3

[a]

Let’s first define the covariance matrix

```
cov <- matrix(c(25, -2, 4, -2, 4, 1, 4, 1, 9),
              nrow = 3)
cov
```

```
##      [,1] [,2] [,3]
## [1,]   25   -2    4
## [2,]   -2    4    1
## [3,]    4    1    9
```

Then, we create the matrix D. This matrix is the square root of the diagonal of the covariance matrix we defined above. Additionally, we fill the diagonals in with zero’s for matrix multiplication.

```
D <- diag(sqrt(diag(cov)))
D
```

```
##      [,1] [,2] [,3]
## [1,]    5    0    0
## [2,]    0    2    0
## [3,]    0    0    3
```

Next, we find the inverse of the matrix D.

```
D_inv <- solve(D)
D_inv
```

```
##      [,1] [,2]      [,3]
## [1,]  0.2  0.0 0.0000000
## [2,]  0.0  0.5 0.0000000
## [3,]  0.0  0.0 0.3333333
```

Finally, we compute the correlation matrix by multiplying the inverse of D by the covariance matrix and then by the inverse once again. The resulting matrix is displayed.

```
cor <- D_inv %*% cov %*% D_inv
cor
```

```
##            [,1]       [,2]      [,3]
## [1,]  1.0000000 -0.2000000 0.2666667
## [2,] -0.2000000  1.0000000 0.1666667
## [3,]  0.2666667  0.1666667 1.0000000
```

[b]

Because V is a diagonal matrix, we can simply take the square root of the diagonal vector. Below, we create V from the covariance matrix and then take the square root.

```
V <- diag(diag(cov))
V_sqrt <- sqrt(V)
V_sqrt
```

```
##      [,1] [,2] [,3]
## [1,]    5    0    0
## [2,]    0    2    0
## [3,]    0    0    3
```

[c]

When we multiply the matrices together as shown, we get the original covariance matrix.

```
V_sqrt %*% cor %*% V_sqrt
```

```
##      [,1] [,2] [,3]
## [1,]   25   -2    4
## [2,]   -2    4    1
## [3,]    4    1    9
```