is605_Assignment4

Prashant B. Bhuyan

February 22, 2015

PS1

Step 1. Create Matrix A

```
A <- matrix(c(1,2,3,-1,0,4), nrow = 2, ncol = 3, byrow = TRUE)
A
```

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

Step 2. Create Transpose of MatrixA

```
ATrans <- t(A)
ATrans
```

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

Step 3. Compute AATrans and store in variable X

```
X <- A%*%ATrans
X
```

```
##      [,1] [,2]
## [1,]   14   11
## [2,]   11   17
```

Step 4. Compute ATransA and storein variable Y

```
Y <- ATrans%*%A
Y
```

```
##      [,1] [,2] [,3]
## [1,]    2    2   -1
## [2,]    2    4    6
## [3,]   -1    6   25
```

Step 5. Compute SVD of A and store in a variable ASVD

```
ASVD <- svd(A, nu = nrow(A), nv = ncol(A))
ASVD
```

```
## $d
## [1] 5.158 2.097
##
## $u
##         [,1]    [,2]
## [1,] -0.6576 -0.7534
## [2,] -0.7534  0.6576
##
## $v
##          [,1]    [,2]    [,3]
## [1,]  0.01857 -0.6728 -0.7396
## [2,] -0.25500 -0.7185  0.6472
## [3,] -0.96676  0.1766 -0.1849
```

Verfiy that the two sets of singular vectors are the eigenvectors of X and Y.

Here are the eigenvalues and eigenvectors of X:

```r
eigenX <- eigen(X)
eigenX
```

```
## $values
## [1] 26.602  4.398
##
## $vectors
##        [,1]    [,2]
## [1,] 0.6576 -0.7534
## [2,] 0.7534  0.6576
```

Here are the eigenvalues and eigenvectors of Y:

```r
eigenY <- eigen(Y)
eigenY
```

```
## $values
## [1] 2.660e+01 4.398e+00 1.059e-16
##
## $vectors
##          [,1]    [,2]    [,3]
## [1,] -0.01857 -0.6728  0.7396
## [2,]  0.25500 -0.7185 -0.6472
## [3,]  0.96676  0.1766  0.1849
```

The eigenvectors of X and Y above are indeed the same as the two sets of singular vectors.

Step 6. The squares of the non-zero singular values of A are equivalent to the eigenvalues of X and Y. The square roots of the eigenvalues of X are:

```r
eigenXVals <- c(26.601802, 4.398198)
sqrt(eigenXVals)
```

```
## [1] 5.158 2.097
```

```r
eigenYVals <- c(26.60180, 4.398198, 0)
sqrt(eigenYVals)
```

```
## [1] 5.158 2.097 0.000
```

We can see from above that the square roots of the eigenvalues of X and Y are the same as the non-zero singular values of A. Thus, the squares of the non-zero singular values of A are the same as the eigenvalues of X and Y.

PS2

First, we create a function that constructs the submatrix submat. This function submat takes values for the row and column that needs to be eliminated from the passed parent matrix.

```r
submat <- function(A, i, j){
  return(det(A[-i, -j]))
  }
```

Next, we create a function that computes the cofactors of the parent matrix passing in the row and column variables that need to be eliminated from the sub matrix (via the submat function call).

```r
cofactor <- function(A, i, j){
  return((-1)^(i+j)*submat(A, i, j))
  }
```

Finally, we create the myinverse function. The function takes values for a full rank square matrix A as well as values for i, j which are row and column values that are needed to compute the submatrix and cofactors of the parent matrix.

```r
myinverse <- function(A, i, j){
  c <- cofactor(A, i, j)
  ctrans <- function(A){
  n <- nrow(A)
  B <- matrix(NA, n, n)
  for(i in 1:n)
    for(j in 1:n)
      B[j,i] <- cofactor(A, i, j)
  return(B)
  }
  return(ctrans(A)/det(A))
  }
```

Test Case:

testMat is a full rank square matrix. B is the inverse of testMat using cofactor, submatrix, cofactor transpose and determinant functions.

```r
testMat <- matrix(c(1, 2, 3, 4, 5, 6, 13, 19, 88), nrow = 3, ncol = 3, byrow = TRUE)
testMat
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    3
## [2,]    4    5    6
## [3,]   13   19   88
```

```r
B <- myinverse(testMat, 1, 1)

B
```

```
##          [,1]     [,2]     [,3]
## [1,] -1.7249  0.62963  0.01587
## [2,]  1.4497 -0.25926 -0.03175
## [3,] -0.0582 -0.03704  0.01587
```

And multiplying testMat by the inverse of testMat B we get the identity matrix where the diagnols are approximately 1 and the non-diagnols are approximately 0.

```r
testMat%*%B
```

```
##           [,1]       [,2]       [,3]
## [1,] 1.000e+00 -4.580e-16 -2.776e-17
## [2,] 6.550e-15  1.000e+00 -6.939e-17
## [3,] 2.043e-14 -4.441e-15  1.000e+00
```