Data605-Week1-HomeWork2-kamath

Vinayak Kamath

09/06/2020

Problem set 1

(1) Show that $A^TA \neq AA^T$ in general. (Proof and demonstration.)

Figure 1: .

```
#defining the sample matrix:
A <- matrix(c(3, 2, 2, -1, 5, 0, 4, 1, 6), 3, 3)
A
```

```
##      [,1] [,2] [,3]
## [1,]    3   -1    4
## [2,]    2    5    1
## [3,]    2    0    6
```

```
A1 <- t(A)
A1
```

```
##      [,1] [,2] [,3]
## [1,]    3    2    2
## [2,]   -1    5    0
## [3,]    4    1    6
```

```
#Checking by comparing the Matrix side by side.
A1 %*% A
```

```
##      [,1] [,2] [,3]
## [1,]   17    7   26
## [2,]    7   26    1
## [3,]   26    1   53
```

```
A %*% A1
```

```
##      [,1] [,2] [,3]
## [1,]   26    5   30
## [2,]    5   30   10
## [3,]   30   10   40
```

```r
#Checking with NOt equal to operator '!='
A1 %*% A != A %*% A1
```

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

==> From above two methods we can say that A1 %% A and A %% A1, are not equal in this case

(2) For a special type of square matrix A, we get $A^TA = AA^T$. Under what conditions could this be true? (Hint: The Identity matrix I is an example of such a matrix).

Figure 2: .

```
#defining the square  matrix:
A <- matrix(c(2, 0,2, 0, 2, 0, 2, 0, 2), 3, 3)
A
```

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

```
A1 <- t(A)
A1
```

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

```
#Checking if A == A1
A == A1
```

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

```
#Checking by comparing the Matrix side by side.
A1 %*% A
```

```
##      [,1] [,2] [,3]
## [1,]    8    0    8
## [2,]    0    4    0
## [3,]    8    0    8
```

```
A %*% A1
```

```
##      [,1] [,2] [,3]
## [1,]    8    0    8
## [2,]    0    4    0
## [3,]    8    0    8
```

```
#Checking with equal to operator '=='
A1 %*% A == A %*% A1
```

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

==> When A and A1 are same then we can get A1 %% A and A %% A1 as equal

Problem set 2

Matrix factorization is a very important problem. There are supercomputers built just to do matrix factorizations. Every second you are on an airplane, matrices are being factorized. Radars that track flights use a technique called Kalman filtering. At the heart of Kalman Filtering is a Matrix Factorization operation. Kalman Filters are solving linear systems of equations when they track your flight using radars.

Write an R function to factorize a square matrix A into LU or LDU, whichever you prefer. Please submit your response in an R Markdown document using our class naming convention, E.g. LFulton_Assignment2_PS2.png

Figure 3: .

```
#defining the square  matrix:
A <- matrix(c(2, 6, -2, -1, 5, 0, 4, 1, 6), 3, 3)
A
```

```
##      [,1] [,2] [,3]
## [1,]    2   -1    4
## [2,]    6    5    1
## [3,]   -2    0    6
```

```
#Getting cell 2, 1 as 0
A21 <- matrix(c(1, -(6/2), 0, 0, 1, 0, 0, 0, 1), 3, 3)
A21  %*%  A
```

```
##      [,1] [,2] [,3]
## [1,]    2   -1    4
## [2,]    0    8  -11
## [3,]   -2    0    6
```

```
#Getting cell 3, 1 as 0
A31 <- matrix(c(1, 0, -(-2/2), 0, 1, 0, 0, 0, 1), 3, 3)
A31  %*% A21  %*%  A
```

```
##      [,1] [,2] [,3]
## [1,]    2   -1    4
## [2,]    0    8  -11
## [3,]    0   -1   10
```

```
#Getting cell 3, 2 as 0
A32 <- matrix(c(1, 0, 0, 0, 1, -(-1/8), 0, 0, 1), 3, 3)
A32 %*% A31  %*% A21  %*%  A
```

```
##      [,1] [,2]    [,3]
## [1,]    2   -1   4.000
## [2,]    0    8 -11.000
## [3,]    0    0   8.625
```

```r
#Upper Triangular matrix U
U  <- A32 %*% A31  %*% A21  %*%  A
U
```

```
##      [,1] [,2]    [,3]
## [1,]    2   -1   4.000
## [2,]    0    8 -11.000
## [3,]    0    0   8.625
```

```r
#Lower Triangular matrix L
L <- solve(A21) %*%  solve(A31)  %*% solve(A32)
L
```

```
##      [,1]   [,2] [,3]
## [1,]    1  0.000    0
## [2,]    3  1.000    0
## [3,]   -1 -0.125    1
```

```r
#Checking for factorize for square matrix A into LU
A == L %*% U
```

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