IS 605 Assignment 2

Oluwakemi Omotunde

February 12, 2017

2. 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 You don't have to worry about permuting rows of A and you can assume that A is less than 5x5, if you need to hard-code any variables in your code. If you doing the entire assignment in R, then please submit only one markdown document for both the problems.

- I have decided to write a function to factorize a square matrix A into LU. Before writing out our fuction, I would like to go through the steps required when performing LU factorization.
  - Our first step is to row reduce A to get U which is the echelon form of A. U is also an upper triangular matrix.
  - Second, we calculate L, a lower triangular matrix.

```
#A is a 2x2 matrix
lufact <- function(A){
multiplier1 <- -A[2,1]/A[1,1]
L <- matrix(c(1,0,multiplier1,1), nrow = 2, byrow = T)
U <- L%*%A

print(A)
print(L)
print(U)
}
#I could not find a way to print all 3 variables in one statement
```

I ran the code above and it worked so I'd like to try it on a 2x2 matrix and compare my answer with the built in function.

```
require(Matrix)
```

```
## Loading required package: Matrix
```

```
A <- matrix(c(4, 10, 8, 1),  nrow = 2, byrow = T)
lufact(A)
```

```
##      [,1] [,2]
## [1,]    4   10
## [2,]    8    1
##      [,1] [,2]
## [1,]    1    0
## [2,]   -2    1
##      [,1] [,2]
## [1,]    4   10
## [2,]    0  -19
```

```
lu(A)
```

```
## 'MatrixFactorization' of Formal class 'denseLU' [package "Matrix"] with 4 slots
```

1

```
##   ..@ x       : num [1:4] 8 0.5 1 9.5
##   ..@ perm    : int [1:2] 2 2
##   ..@ Dimnames:List of 2
##   .. ..$ : NULL
##   .. ..$ : NULL
##   ..@ Dim     : int [1:2] 2 2
```

#I thought I would be able to check my work with this build in fuction but I could out understand the r