DATA605: Fundamentals of Computational Mathematics

Assignment 2

Donald Butler

02/04/2022

Problem Set 1

1. Show that $A^TA \neq AA^T$ in general.

Let A be an m $\times$ n matrix such that $m \neq n$. A^T has dimensions n $\times$ m. AA^T has dimensions m $\times$ m and A^TA has dimensions n $\times$ n. Since $m \neq n$, the dimensions of AA^T and A^TA are not the same, therefore $A^TA \neq AA^T$.

2. For a special type of square matrix A, we get $A^TA = AA^T$. Under what conditions could this be true?

This would be true for all symmetric matrices because $A = A^T$ would imply that $A^TA = AA^T$.

Problem Set 2

Write an R function to factorize a square matrix A into LU or LDU, whichever you prefer.

```
LU <- function(A, debug=FALSE) {
  if (nrow(A) != ncol(A)) {
    return ("A is not square")
  }

  n <- ncol(A)
  U <- A
  L <- diag(n)

  for (col in c(1:(n-1))) {
    if (U[col,col] == 0) {
      # needs a row swap to get a non-zero element in position U[col,col]
      if (debug) {
        print(sprintf("needs row swap in col %s U=",col))
        print(U)
      }
      for (row in c((col+1):n)) {
        if (U[row,col] != 0) {
          E <- diag(n)
          E[col,col] <- 0
```

```
          E[row,row] <- 0
          E[col,row] <- 1
          E[row,col] <- 1
          U <- E %*% U
          L <- L %*% solve(E)
          if (debug) {
            print(sprintf("row swap complete with row %s U=",row))
            print(U)
          }
          break # break from the row swap loop
        }
      }
    }

    if (U[col,col] == 0) {
      # All elements in col are 0
      if (debug) {
        print(sprintf("entire col %s is 0 U=",col))
        print(U)
      }
    } else {
      for (row in c((col+1):n)) {
        if (U[row,col] == 0) {
          if (debug) {
            print(sprintf("eliminating row %s col %s is not needed",row,col))
            print("U=")
            print(U)
          }
        } else {
          E <- diag(n)
          E[row,col] <- -1 * U[row,col] / U[col,col]
          if (debug) {
            print(sprintf("elminating row %s col %s E=",row,col))
            print(E)
          }
          U <- E %*% U
          L <- L %*% solve(E)
          if (debug) {
            print("U=")
            print(U)
          }
        }
      }
    }
  }
  return (list(L=L,U=U))
}
```

LU Testing 1

Using the 3×3 example from the Weekly Materials.

```
A <- matrix(c(1,2,3,1,1,1,2,0,1),nrow=3)
r = LU(A)
print(r[1])
```

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

```
print(r[2])
```

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

```
r[1]$L %*% r[2]$U == A
```

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

LU Testing 2

Test a matrix with a zero in a pivot to verify the row swap functionality.

```
A <- matrix(c(0,2,3,1,1,1,2,0,1),nrow=3)
r = LU(A)
print(r[1])
```

```
## $L
##      [,1] [,2] [,3]
## [1,]  0.0  1.0    0
## [2,]  1.0  0.0    0
## [3,]  1.5 -0.5    1
```

```
print(r[2])
```

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

```
r[1]$L %*% r[2]$U == A
```

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

Testing my function with this matrix produced a valid upper-triangular matrix but L was not a lower-triangular matrix. I suspect my pivoting technique is not correct even though I am getting an LU=A result.

LU Testing 3

Testing with a higher order matrix

```
A <- matrix(c(1,2,3,4,2,4,3,1,3,2,1,4,4,2,1,3),nrow=4)
r = LU(A)
print(r[1])
```

```
## $L
##      [,1]     [,2]      [,3] [,4]
## [1,]    1 0.000000  0.000000    0
## [2,]    2 0.000000  1.000000    0
## [3,]    3 1.000000  0.000000    0
## [4,]    4 2.333333 -2.666667    1
```

```
print(r[2])
```

```
## $U
##      [,1] [,2] [,3]       [,4]
## [1,]    1    2    3   4.000000
## [2,]    0   -3   -8 -11.000000
## [3,]    0    0   -4  -6.000000
## [4,]    0    0    0  -3.333333
```

```
r[1]$L %*% r[2]$U == A
```

```
##      [,1] [,2] [,3] [,4]
## [1,] TRUE TRUE TRUE TRUE
## [2,] TRUE TRUE TRUE TRUE
## [3,] TRUE TRUE TRUE TRUE
## [4,] TRUE TRUE TRUE TRUE
```