jshamp assignment2 PS1 and PS2

Jeff Shamp

2/5/2020

Problem Set 1

Problem 1 - Prove $A^TA \neq AA^T$

Proof by induction:

Assume: $A^TA = AA^T$

Then for rows i in A and columns j in A^T, $A_{i1} \cdot A^T_{1j} = A^T_{i1} \cdot A_{1j}$. This is true if and only if $A_{i1} = A^T_{1j}$, such that $A = A^T$. Therefore, the assumption is false outside of the rare case of identical matricies. Thus, $A^TA \neq AA^T$.

Example:

Consider the following; $A = \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \end{bmatrix}$ and $A^T = \begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \end{bmatrix}$

Here if we take the inner product of the first columns of A and the first row of A^T we get $A_{i1} = 14$. Similarly, we take the first inner product of A^T and A we get $A^T_{1j} = 3$, such at $A_{i1} \neq A^T_{1j}$ and $A^TA \neq AA^T$.

Problem 2 -Under what circumstances could Problem 1 be true?

As mentioned in the proof, this is true for any matrix A that is elementwise identical to its transpose A^T. That is, $A = A^T$.

Problem Set 2

Matrix Factorization

Write a function that factorizes any square matrix less than dimension 5x5. No permuting rows necessary.

```
matrix.factorize<- function(input_mat){
  mat_L<-diag(dim(input_mat)[1])
  row_idx<-1
  for(j in 1:(dim(input_mat)[2]-1)){
    for(i in 1:(dim(input_mat)[1]-row_idx)){
      mat_L[i+row_idx,j]<-(input_mat[i+row_idx,j] /
                             input_mat[j,j])
      input_mat[i+row_idx,]<-((-1*mat_L[i+row_idx,j]) *
                                input_mat[row_idx,]) +
                                input_mat[i+row_idx,]
    }
  row_idx<-row_idx+1
  }
  return(list(input_mat, mat_L))
}
```

Test the known example from the video posted in "Weekly Materials".

```
a<-matrix(c(2,4,-4,1,-4,3,-6,-9,5), byrow = T, nrow=3)
soln<-matrix.factorize(a)
soln[[2]]%*%soln[[1]]
```

```
##      [,1] [,2] [,3]
## [1,]    2    4   -4
## [2,]    1   -4    3
## [3,]   -6   -9    5
```

Returns the input matrix under multiplication such that, $A = LU$.

Let's try one more from the video.

```
b<-matrix(c(1,4,-3,-2,8,5,3,4,7), byrow = T, nrow=3)
soln<-matrix.factorize(b)
soln[[2]]%*%soln[[1]]
```

```
##      [,1] [,2] [,3]
## [1,]    1    4   -3
## [2,]   -2    8    5
## [3,]    3    4    7
```

Solid.