

Data 605 - HW3

Jeff Shamp

2/11/2020

Problem Set 1

- 1) Rank of given matrix A . For a square matrix the rank will be equal to the number of rows so long as the Upper Triangular (I'll call U) form of the matrix has a non-zero determinant. If $\det(U) = 0$, then we know the rank is less than the dimension and will need to repeat this process for $N-1$ dimension submatrix until a non-zero determinant is found. For some cases, visual inspection might be better but those are rare cases and do not scale well.

I'm going to bring in a function from last weeks HW to calculate the Upper Triangular matrix.

```
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))
}
```

```
A<- matrix(c(1,2,3,4,-1,0,1,3,0,1,-2,1,5,4,-2,-3), byrow=T, nrow=4)
soln<-matrix.factorize(A)
det(soln[[1]])
```

```
## [1] -9
```

Thus the rank of matrix A is 4, since it has a non=zero determinant. I'm going to try another example where I know the answer, just to be sure this works. A 4x4 checkerboard matrix of 1s and -1s should have a rank of 1.

```
C<-matrix(c(1,-1,1,-1,-1,1,-1,1,1,-1,1,-1,-1,1,-1,1), byrow = T, nrow = 4)
soln<- matrix.factorize(C)
det(soln[[1]])
```

```
## [1] 0
```

This is the expected answer for this matrix. Let's take one step further and find the rank of square matrix. I'll return a vector of all the determinants as well for redundancy and transparency.

```
find.matrix.rank<- function(mat){  
  mat_dim<-dim(mat)[1]-1  
  rank<-dim(mat)[1]  
  mat_det_vec<-rep(NA,mat_dim)  
  for (i in 1:mat_dim){  
    mat_det_vec[i]<-det(matrix.factorize(mat)[[1]])  
    if (mat_det_vec[i]!=0){  
      return(list(rank, mat_det_vec))  
    } else {  
      rank<-rank-1  
      mat<-mat[1:dim(mat)[1]-1, 1:dim(mat)[2]-1]  
    }  
  }  
  return(list(rank, mat_det_vec))  
}  
find.matrix.rank(C)
```

```
## [[1]]  
## [1] 1  
##  
## [[2]]  
## [1] 0 0 0
```

As expected. Let's re-try problem 1.1 with the same matrix A.

```
find.matrix.rank(A)
```

```
## [[1]]  
## [1] 4  
##  
## [[2]]  
## [1] -9 NA NA
```

Excellent.

- 2) The minimum rank for any non-zero matrix is rank =1. The maximum is no greater than the lesser of the dimensions. In this case, n .
- 3) Rank of a given matrix can be found with out handy function from 1.1

```
B<-matrix(c(1,2,1,3,6,3,2,4,2), byrow=T, nrow=3)  
find.matrix.rank(B)
```

```
## [[1]]  
## [1] 1  
##  
## [[2]]  
## [1] 0 0
```

This makes sense as this represents three parallel lines, which is a dependent system.

Problem 2 - Eigenvectors and values

We are to calculate the eigenvector and values for a 3x3 matrix in Upper Triangular form, which is helpful. I'll do most of this work by-hand but with some machine computation. First set up the $\det(A - \lambda I)$ equation.

$$\det\left(\begin{bmatrix} 1 & 2 & 3 \\ 0 & 4 & 5 \\ 0 & 0 & 6 \end{bmatrix} - \begin{bmatrix} \lambda & 0 & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & \lambda \end{bmatrix}\right) = 0$$

Which expands to the following:

$$(1 - \lambda)\det\left(\begin{bmatrix} 4 - \lambda & 5 \\ 0 & 6 - \lambda \end{bmatrix}\right) - (2)\det\left(\begin{bmatrix} 0 & 5 \\ 0 & 6 - \lambda \end{bmatrix}\right) + (3)\det\left(\begin{bmatrix} 0 & 4 - \lambda \\ 0 & 0 \end{bmatrix}\right) = 0$$

Thankfully, the last two terms of this expansion reduce to zero. Thus we are left with the following:

$$(1 - \lambda)(24 - 10\lambda + \lambda^2) = 0$$

This expands to the following,

$$24 - 34\lambda + 11\lambda^2 - \lambda^3 = 0$$

Which can be factored via the fundamental theorem of algebra (luckily $\lambda = 1$ is a root!). We see that this reduces to,

$-\lambda^2 + 10\lambda - 24 = 0$. Which factors our last two remaining eigenvalues and our set of eigenvalues are as follows:

$$\lambda_1 = 1, \lambda_2 = 4, \text{ and } \lambda_3 = 6.$$

If we return the first eigenvalue in the original matrix A we apply $\lambda * I - A$ as the following,

$$\begin{bmatrix} 0 & 2 & 3 \\ 0 & 3 & 5 \\ 0 & 0 & 5 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

The second eigenvalue can be used to obtain the eigenvector in the same way subtract $\lambda = 4$ from the diagonal of the original matrix A and the same is true for the the last eigenvalue.

$$\begin{bmatrix} -3 & 2 & 3 \\ 0 & 0 & 5 \\ 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = 0$$

R can compute the values of the eigenvectors and reduce them to unit values by calling the `eigen(A)$vectors` command.

```
A<-matrix(c(1,2,3,0,4,5,0,0,6), byrow = T, nrow = 3)
eigen(A)$vectors
```

```
##           [,1]      [,2] [,3]
## [1,] 0.5108407 0.5547002    1
## [2,] 0.7981886 0.8320503    0
## [3,] 0.3192754 0.0000000    0
```

This is the span of vectors for the Eigenspace that are characteristic to matrix A .