MatricesWorksheet.R

rachelfeingold

Sat Feb 4 15:27:21 2017

```
# Problem 1
# a. Check if A^3 = 0
A <- matrix(data =c(1,5,-2,1,2,-1,3,6,-3), nrow = 3, ncol = 3)
A
```

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

```
A%*%A%*%A
```

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

```
# b. Replace the third column of A by the sum of the second and third columns.
A[,3] <- A[,2]+A[,3]
A
```

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

```
# Problem 2. Create Matrix B and calculate B^TB
B <- matrix(data=c(10,-10,10), byrow = TRUE, nrow = 15, ncol = 3)
Prod <- t(B)%*%B
Prod
```

```
##       [,1]  [,2]  [,3]
## [1,]  1500 -1500  1500
## [2,] -1500  1500 -1500
## [3,]  1500 -1500  1500
```

```
# Problem 3. Create Matrix matE with all entries equal to 0
matE <- matrix(0, nrow = 6, ncol = 6)
row(matE)
```

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

```
col(matE)
```

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

```
(abs(row(matE)-col(matE))==1)*1
```

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

```
# Problem 4. Use function outer to create a patterned matrix
s <-  0:4
outer(s,s,"+")
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    0    1    2    3    4
## [2,]    1    2    3    4    5
## [3,]    2    3    4    5    6
## [4,]    3    4    5    6    7
## [5,]    4    5    6    7    8
```

```
# Problem 5. Create large patterned matrices
#a.
s <-  0:4
outer(s,s,"+")%%5
```

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

```
#b.
a <- 0:9
outer(a,a,"+")%%10
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
##  [1,]    0    1    2    3    4    5    6    7    8     9
##  [2,]    1    2    3    4    5    6    7    8    9     0
##  [3,]    2    3    4    5    6    7    8    9    0     1
##  [4,]    3    4    5    6    7    8    9    0    1     2
##  [5,]    4    5    6    7    8    9    0    1    2     3
##  [6,]    5    6    7    8    9    0    1    2    3     4
##  [7,]    6    7    8    9    0    1    2    3    4     5
##  [8,]    7    8    9    0    1    2    3    4    5     6
##  [9,]    8    9    0    1    2    3    4    5    6     7
## [10,]    9    0    1    2    3    4    5    6    7     8
```

```
#c.
outer(0:8,0:8,"-")%%9
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##  [1,]    0    8    7    6    5    4    3    2    1
##  [2,]    1    0    8    7    6    5    4    3    2
##  [3,]    2    1    0    8    7    6    5    4    3
##  [4,]    3    2    1    0    8    7    6    5    4
##  [5,]    4    3    2    1    0    8    7    6    5
##  [6,]    5    4    3    2    1    0    8    7    6
##  [7,]    6    5    4    3    2    1    0    8    7
##  [8,]    7    6    5    4    3    2    1    0    8
##  [9,]    8    7    6    5    4    3    2    1    0
```

```
# Problem 6. Solve system of linear equations
y <- matrix(c(7,-1,-3,5,17),c(5,1), byrow = TRUE)
A <- matrix(0,nrow = 5,ncol = 5)
A <- abs(row(A)-col(A))+1
x <- solve(A,y)
x
```

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

```
# Problem 7. Create Matrix of random intergers from from 1,2,...,10.Matrix is 6 x 10.
set.seed(75)
aMat <- matrix( sample(10, size=60, replace=T), nr=6)
aMat
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,]    3    6    7    7    2    4    3    7    1     4
## [2,]    1    9    8    7    2    6   10    9    5     2
## [3,]    7   10    8    4   10    5    4    8    4     4
## [4,]    4    3    1    1    3    3    9    7    4     2
## [5,]    1    8    1    9    9    8    1    3    7     7
## [6,]    2    6    7    5    6   10    4    6   10     1
```

```
# a. Find numbers of entities in each row which are greater than 4.
cond <- (aMat>4) *1
rowSums(cond)
```

```
## [1] 4 7 6 2 6 7
```

```
# b. Which rows contain exactly two occurrences of the number seven?
cond <- (aMat==7) *1
x <- rowSums(cond)
which(x==2)
```

```
## [1] 5
```

```
# c. Find those pairs of columns whose total (over both columns) is greater than 75.
ColSumsaMat <- colSums(aMat)
ColSumsaMat
```

```
##  [1] 18 42 32 33 32 36 31 40 31 20
```

```
t <- outer(ColSumsaMat,ColSumsaMat,'+') #forms matrix of two column sums
t1 <- t>75 # filter for pairs whose sum >75 (Logicals)
```

```
t1 <- t1*1 # filtered as Binaries
t1 <- upper.tri(t1,diag=TRUE)*1*t1 #eliminate the repeated answers in the lower triangle
t1 <- upper.tri(t1,diag=TRUE)&t1  #eliminates repeated answers in logical matrix using logical AND oper
t1
```

```
##        [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10]
##  [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [2,] FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE
##  [3,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [4,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [5,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [6,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
##  [7,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [8,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
##  [9,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [10,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
which(t1,arr.ind=TRUE, useNames = TRUE)
```

```
##      row col
## [1,]   2   2
## [2,]   2   6
## [3,]   2   8
## [4,]   6   8
## [5,]   8   8
```

```
# Problem 8. Calculate

options(digits = 2)
# a. Need to seperate the sums: sum[(i=1:20) i^4 * sum[(i=1:5) (i+3)^-1]] #calculate the inner sum firs
s1 <- sum(1/4:8)
s1 * sum((1:20)^4) #multple prvious sum by the outer sum
```

```
## [1] 639215
```

```
# b. Need to make a  matrix by forming 2 vectors  and using outer(). Make the denominator first.
outer(1:20,1:5,"*") # matrix of ij products
```

```
##       [,1] [,2] [,3] [,4] [,5]
##  [1,]    1    2    3    4    5
##  [2,]    2    4    6    8   10
##  [3,]    3    6    9   12   15
##  [4,]    4    8   12   16   20
##  [5,]    5   10   15   20   25
##  [6,]    6   12   18   24   30
##  [7,]    7   14   21   28   35
##  [8,]    8   16   24   32   40
##  [9,]    9   18   27   36   45
## [10,]   10   20   30   40   50
## [11,]   11   22   33   44   55
## [12,]   12   24   36   48   60
## [13,]   13   26   39   52   65
## [14,]   14   28   42   56   70
## [15,]   15   30   45   60   75
## [16,]   16   32   48   64   80
## [17,]   17   34   51   68   85
## [18,]   18   36   54   72   90
## [19,]   19   38   57   76   95
## [20,]   20   40   60   80  100
```

```
denom <- outer(1:20,1:5,"*") +3
denom
```

```
##       [,1] [,2] [,3] [,4] [,5]
##  [1,]    4    5    6    7    8
##  [2,]    5    7    9   11   13
##  [3,]    6    9   12   15   18
##  [4,]    7   11   15   19   23
##  [5,]    8   13   18   23   28
##  [6,]    9   15   21   27   33
##  [7,]   10   17   24   31   38
##  [8,]   11   19   27   35   43
##  [9,]   12   21   30   39   48
## [10,]   13   23   33   43   53
## [11,]   14   25   36   47   58
## [12,]   15   27   39   51   63
## [13,]   16   29   42   55   68
## [14,]   17   31   45   59   73
## [15,]   18   33   48   63   78
## [16,]   19   35   51   67   83
## [17,]   20   37   54   71   88
## [18,]   21   39   57   75   93
## [19,]   22   41   60   79   98
## [20,]   23   43   63   83  103
```

```
numer <- (1:20)^4
numer
```

```
##  [1]      1     16     81    256    625   1296   2401   4096   6561  10000
## [11]  14641  20736  28561  38416  50625  65536  83521 104976 130321 160000
```

```
sum(numer/denom)
```

```
## [1] 89912
```

```
# c. Similiar to b, need a 10x10 denom matrix and a length=10 numerator
denom <- outer(1:10, 1:10,"*") +3
numer <- (1:10)^4
t <- numer/denom
t
```

```
##         [,1] [,2]   [,3]   [,4]   [,5]   [,6]   [,7]    [,8]    [,9]
##  [1,]   0.25   0.2   0.17   0.14   0.12   0.11   0.10   0.091   0.083
##  [2,]   3.20   2.3   1.78   1.45   1.23   1.07   0.94   0.842   0.762
##  [3,]  13.50   9.0   6.75   5.40   4.50   3.86   3.38   3.000   2.700
##  [4,]  36.57  23.3  17.07  13.47  11.13   9.48   8.26   7.314   6.564
##  [5,]  78.12  48.1  34.72  27.17  22.32  18.94  16.45  14.535  13.021
##  [6,] 144.00  86.4  61.71  48.00  39.27  33.23  28.80  25.412  22.737
##  [7,] 240.10 141.2 100.04  77.45  63.18  53.36  46.17  40.695  36.379
##  [8,] 372.36 215.6 151.70 117.03  95.26  80.31  69.42  61.134  54.613
##  [9,] 546.75 312.4 218.70 168.23 136.69 115.11  99.41  87.480  78.107
## [10,] 769.23 434.8 303.03 232.56 188.68 158.73 136.99 120.482 107.527
##        [,10]
##  [1,]  0.077
##  [2,]  0.696
##  [3,]  2.455
##  [4,]  5.953
##  [5,] 11.792
##  [6,] 20.571
##  [7,] 32.890
##  [8,] 49.349
##  [9,] 70.548
## [10,] 97.087
```

```
# sum either upper or lower triangle, try out both, use lower
sum(lower.tri(t,dia=TRUE)*t)
```

```
## [1] 6945
```