HW1 - Esther Xuanpei Ouyang

STAT 154 Lab 101

Esther Xuanpei Ouyang

1/26/2018

Problem 1

```
X = matrix(c(2,3,-1,4), nrow = 2, ncol = 2)
X
```

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

```
Y = matrix(c(2,0,1,1,-2,3), nrow = 2, ncol = 3)
Y
```

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

```
Z = matrix(c(1,1,-1,1,0,2), nrow = 3, ncol = 2)
Z
```

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

```
W = matrix(c(1,0,8,3), nrow = 2, ncol = 2)
W
```

```
##      [,1] [,2]
## [1,]    1    8
## [2,]    0    3
```

```
I = matrix(c(1,0,0,1), nrow = 2, ncol = 2)
I
```

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

Problem 2

(a)

```
# X + Y
# Cannot be performed because the dimension of X and Y are not the same.
```

(b)

```
X + W
```

```
##      [,1] [,2]
## [1,]    3    7
## [2,]    3    7
```

(c)

```
X - I
```

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

(d)

```
X%*%Y
```

```
##      [,1] [,2] [,3]
## [1,]    4    1   -7
## [2,]    6    7    6
```

(e)

```
X%*%I
```

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

(f)

$X + (Y + Z)$ cannot be performed because the dimension of the three matrices X, Y, Z are not the same.

(g)

$Y(I + W)$ cannot be performed because the dimension of $(I + W)$ is 2 by 2 and the dimension of Y is 2 by 3, and the col number of Y does not match with the row number of $(I + W)$.

Problem 3

(a) Every orthogonal matrix is nonsingular.

True.

By the definition of orthogonal matrix, for orthogonal matrix Q, the product of Q^T and Q, $Q^TQ = I$. Also, by the definition of nonsingular matrix A, a matrix is nonsingular if there exists a matrix B such that $AB = I$. Therefore, every orthogonal matrix Q is a nonsingular matrix since its transpose Q^T always exists and $Q^TQ = I$.

(b) Every nonsingular matrix is orthogonal.

False.

By the definition of orthogonal matrix, for for orthogonal matrix Q, $Q^TQ = QQ^T = I$. A counter-example is that the nonsingular matrix A is not orthogonal Therefore, not all nonsingular matrix is orthogonal.

(c) Every matrix of full rank is square.

False

By the definition of rank, a matrix m by n matrix A is a full rank rank if its rank equals the largest possible for a matrix of the same dimensions, which is the lesser of the number of rows and columns, i.e., $rank(A) = min(m, n)$. Some m by n rectangular matrix can have $rank = min(m, n)$ as long as it has all linearly indepedent vectors as its rows if $m < n$ or columns if $m > n$. A counter-example is that the matrix $X = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ has $rank = 2$, which satisify that $rank(X) = min(2, 3) = 2$. Therefore, not all matrix of full rank is square.

(d) Every square matrix is of full rank.

False

Not all square matrix is of full rank. A counter-example is that the matrix $A = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix}$ is a 2 by 2 square matrix but not of full rank since the row vectors of A are linearly dependent.

(e) Every nonsingular matrix is of full rank.

True

By the invertible matrix theorem, a matrix A is a nonsinguar matrix is equivalent to A is a full rank matrix. Therefore, every nonsingular matrix is of full rank.

Problem 4

Want to Proof: $(XYZ)^T = Z^TY^TX^T$

We would like to prove that $(AB)^T = B^TA^T$,

$$((AB)^T)_{ij} = (AB)_{ji} = \sum_k A_{jk}B_{ki} = \sum_k (A^T)_{kj}(B^T)_{ik} = \sum_k (B^T)_ik(A^T)_{kj} = (B^TA^T)_{ij} \ (1)$$

Therefore, $(AB)^T = B^TA^T$

Let $A = XY, B = Z$,

$$(XYZ)^T = (AB)^T = B^TA^T \quad \text{by (1)} = (Z)^T(XY)^T \quad \text{(plug in A,B)} = Z^TY^TX^T \quad \text{by (1)}$$

Problem 5

Consider the eigenvalue decomposition of a n by n symmetric matrix A, Prove that two eigenvectors v_i and v_j associated with two distinct eigenvalues λ_i and λ_j of A are mutually orthogonal; that is , $v_i^Tv_j = 0$.

By the definition of eigenvalues and eigenvectors, $A\vec{q} = \lambda\vec{q}$. Since A is a symmetric matrix, we can do eigendecomposition on matrix A. Here, I denote the orthonormal basis of eigenvectors as $q_1, \ldots, q_n$ and their corresponding eigenvalues $\lambda_1, \ldots, \lambda_n$. By eigendecomposition,

$$A = U\Lambda U^T$$

where U is a orthogonal matrix with $q_1, \ldots, q_n$ as its columns and Λ is a diagonal matrix $diag(\lambda_1, \ldots, \lambda_n)$. Since by definition $Aq_i = \lambda_i q_i$ for every i, for every column `q_i` of U, i.e., eigenvectors of U which corresponding to a distinct eigenvalues λ_i in Λ. Also, U is a orthonormal matrix, its columns are all orthogonal to each other, i.e., $q_i^T q_j = 0$ for $i \neq j$.

Problem 6

Problem 6.1

```
inner_product = function (v, u) {
  return (v%*%u)
}
v = c(1, 3, 5)
u = c(1, 2, 3)
inner_product(v, u)
```

```
##      [,1]
## [1,]   22
```

Problem 6.2

```
projection = function (v, u) {
  numer = inner_product(u, v)
  denom = inner_product(u, u)
  return ((numer/denom)*u)
}
v = c(1,3,5)
u = c(1,2,3)
projection(v, u)
```

```
## [1] 1.571429 3.142857 4.714286
```

Problem 7

```
vnorm = function(x) {
  return(sqrt(t(x)%*%x))
}

x = c(1, 2, 3)
y = c(3, 0, 2)
z = c(3, 1, 1)
# Start by setting u1 = x, and report the set of vectors uk and the orthonormalized vectors uk, for k =
u1 = x
u1
```

```
## [1] 1 2 3
```

```
u2 = y - projection(y, u1)
u2
```

```
## [1]  2.35714286 -1.28571429  0.07142857
```

```
u3 = z - projection(z, u1) - projection(z, u2)
u3
```

```
## [1]  0.5148515  0.9009901 -0.7722772
```

```
e1 = u1 / vnorm(u1)
e1
```

```
## [1] 0.2672612 0.5345225 0.8017837
```

```
e2 = u2 / vnorm(u2)
e2
```

```
## [1]  0.87758509 -0.47868278  0.02659349
```

```
e3 = u3 / vnorm(u3)
e3
```

```
## [1]  0.3980149  0.6965260 -0.5970223
```

```
# check
vnorm(e1)
```

```
##      [,1]
## [1,]    1
```

```
vnorm(e2)
```

```
##      [,1]
## [1,]    1
```

```
vnorm(e3)
```

```
##      [,1]
## [1,]    1
```

Problem 8

```
# function for computing L_p norm of a vector
#
# x - the input vector, p - the value for p
lp_norm = function(x, p = 1) {
  if (p == "max") {
    return(max(abs(x)))
  } else {
    tot_sum = 0
    for ( val in x ){
      tot_sum = tot_sum + abs(val)^p
    }
    return(tot_sum^(1/p))
  }
}
```

```r
y = matrix(-5:4,10)
y
```

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

```r
lp_norm(y) # default p = 1
```

```
## [1] 25
```

```r
lp_norm(y, p = 2)
```

```
## [1] 9.219544
```

```r
lp_norm(y, p = "max") # L-max norm
```

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

Problem 9

(a)

```r
zero = rep(0, 10)
lp_norm(zero, 1)
```

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

(b)

```r
ones = rep(1, 5)
lp_norm(ones, 2)
```

```
## [1] 2.236068
```

(c)

```r
u = rep(0.4472136, 5)
lp_norm(u, 2)
```

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

(d)

```r
u = 1:500
lp_norm(u, 100)
```

```
## [1] 508.5663
```

(e)

```r
u = 1:500
lp_norm(u, "max")
```

```
## [1] 500
```

Problem 10

Consider the eigendecomposition of a square matrix A.

Since A is a symmetric matrix, we can do eigendecomposition on matrix A. Here, I denote the orthonormal basis of eigenvectors as $q_1, \ldots, q_n$ and their corresponding eigenvalues $\lambda_1, \ldots, \lambda_n$. By eigendecomposition,

$$A = U\Lambda U^T$$

where U is a orthogonal matrix with $q_1, \ldots, q_n$ as its columns and Λ is a diagonal matrix $diag(\lambda_1, \ldots, \lambda_n)$. Since by definition $Aq_i = \lambda_i q_i$ for every i, for every column `q_i` of U, i.e., eigenvectors of U which corresponding to a eigenvalues λ_i in Λ.

$U = \begin{bmatrix} | & & | \\ q_1 & \ldots & q_n \\ | & & | \end{bmatrix}$, where q_i is the i column vector of U, $\Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{bmatrix}$.

a. Prove that the matrix bA, where b is an arbitrary scalar, has $b\lambda$ as an eigenvalue, with v as the associated eigenvector.

By the definition of eigenvalues and eigenvectors,

$$A\vec{q_i} = \lambda_i \vec{q_i}$$

Then, multiple both sides by scalar b,

$$bA\vec{q_i} = k\lambda_i\vec{q_i} \Rightarrow (bA)\vec{q_i} = (k\lambda_i)\vec{q_i}$$

Again, by definition of eigenvalues and eigenvectors, for matrix bA, $\vec{q}$ and $k\lambda$ are the corresponding eigenvectors and eigenvalues.

We can also use eigendecomposition of A to prove.

$$A = U\Lambda U^T$$

$$bA = b(U\Lambda U^T) = (U)(b\Lambda)(U^T)$$

where $b\Lambda = \begin{bmatrix} b\lambda_1 & & 0 \\ & \ddots & \\ 0 & & b\lambda_n \end{bmatrix}$.

Therefore, by eigendecomposition, the matrix bA has $\vec{q}$ as an eigenvector and $b\lambda$ as an eigenvalue.

b. Prove that the matrix $A + cI$, where c is an arbitrary scalar, has $(\lambda + c)$ as an eigenvalue, with v as the associated eigenvector.

By the definition of eigenvalues and eigenvectors,

$$A\vec{q_i} = \lambda_i \vec{q_i}$$

Here, let $A' = A + cI$ and plug in,

$$(A + cI)\vec{q_i} = A\vec{q_i} + cI\vec{q_i} = \lambda_i\vec{q_i} + c\vec{q_i} \text{ (since } A\vec{q_i} = \lambda_i\vec{q_i} \text{)} = (\lambda_i + c)\vec{q_i} \text{ (since } c, \lambda_i \text{ are both scalar value)}$$

Therefore,

$$A'\vec{q_i} = (\lambda_i + c)\vec{q_i}$$

Again, by the definition of eigenvalues and eigenvectors, the matrix $A' = A + cI$ has $vecq_i$ as an eigenvector and $(\lambda_i + c)$ as an eigenvalues.

Problem 11

(a)

Select the first five columns of state.x77 and convert them as a matrix; this will be the data matrix X. Let n be the number of rows of X, and p the number of columns of X.

```
head(state.x77, 5)
```

```
##            Population Income Illiteracy Life Exp Murder HS Grad Frost
## Alabama          3615   3624        2.1    69.05   15.1    41.3    20
## Alaska            365   6315        1.5    69.31   11.3    66.7   152
## Arizona          2212   4530        1.8    70.55    7.8    58.1    15
## Arkansas         2110   3378        1.9    70.66   10.1    39.9    65
## California      21198   5114        1.1    71.71   10.3    62.6    20
##              Area
## Alabama     50708
## Alaska     566432
## Arizona    113417
## Arkansas    51945
## California 156361
```

```
X = matrix(head(state.x77, 5), nrow = 5)
X
```

```
##       [,1] [,2] [,3]  [,4] [,5] [,6] [,7]   [,8]
## [1,]  3615 3624  2.1 69.05 15.1 41.3   20  50708
## [2,]   365 6315  1.5 69.31 11.3 66.7  152 566432
## [3,]  2212 4530  1.8 70.55  7.8 58.1   15 113417
## [4,]  2110 3378  1.9 70.66 10.1 39.9   65  51945
## [5,] 21198 5114  1.1 71.71 10.3 62.6   20 156361
```

```
# number of rows of X
n = dim(X)[1]
n
```

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

```
# number of columns of X
p = dim(X)[2]
p
```

```
## [1] 8
```

(b)

Create a diagonal matrix $D = \frac{1}{n}I$ where I is the nxn identity matrix. Display the output of $sum(diag(D))$.

```r
D = diag(1, n)/n
D
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]  0.2  0.0  0.0  0.0  0.0
## [2,]  0.0  0.2  0.0  0.0  0.0
## [3,]  0.0  0.0  0.2  0.0  0.0
## [4,]  0.0  0.0  0.0  0.2  0.0
## [5,]  0.0  0.0  0.0  0.0  0.2
```

```r
sum(diag(D))
```

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

(c)

Compute the vector of column means $g = X^T D1$ where 1 is a vector of 1's of length n. Display (i.e. print) g.

```r
ones = rep(1, n)
ones
```

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

```r
g = t(X)%*%D%*%ones
g
```

```
##             [,1]
## [1,]   5900.000
## [2,]   4592.200
## [3,]      1.680
## [4,]     70.256
## [5,]     10.920
## [6,]     53.720
## [7,]     54.400
## [8,] 187772.600
```

(d)

Calculate the mean-centered matrix $X_c = X - 1g^T$. Display the output of $colMeans(Xc)$.

```r
Xc = X - ones%*%t(g)
Xc
```

```
##       [,1]    [,2]  [,3]   [,4]  [,5]   [,6]  [,7]      [,8]
## [1,] -2285  -968.2  0.42 -1.206  4.18 -12.42 -34.4 -137064.6
## [2,] -5535  1722.8 -0.18 -0.946  0.38  12.98  97.6  378659.4
## [3,] -3688   -62.2  0.12  0.294 -3.12   4.38 -39.4  -74355.6
## [4,] -3790 -1214.2  0.22  0.404 -0.82 -13.82  10.6 -135827.6
## [5,] 15298   521.8 -0.58  1.454 -0.62   8.88 -34.4  -31411.6
```

```r
colMeans(Xc)
```

```
## [1]  0.000000e+00  1.818989e-13  8.881784e-17 -2.842171e-15  1.776357e-16
## [6] -5.684342e-15 -5.684342e-15 -3.492460e-11
```

(e)

Compute the (population) variance-covariance matrix $V = X^T DX - gg^T$. Display the output of V.

```r
V = t(X)%*%D%*%X - g%*%t(g)
V
```

```
##                [,1]          [,2]         [,3]          [,4]         [,5]
## [1,]  5.957034e+07  1.098069e+06  -2022.52000   5523.936000  -1305.00000
## [2,]  1.098069e+06  1.131175e+06   -258.79600    -42.449200   -505.24400
## [3,] -2.022520e+03 -2.587960e+02      0.12160     -0.211080      0.29840
## [4,]  5.523936e+03 -4.244920e+01     -0.21108      0.942624     -1.51012
## [5,] -1.305000e+03 -5.052440e+02      0.29840     -1.510120      5.68160
## [6,]  2.572120e+04  1.110568e+04     -3.04360      2.263080    -10.96440
## [7,] -1.765460e+05  3.461632e+04     -2.89200    -21.632400      5.77200
## [8,] -2.948424e+08  1.876433e+08 -29262.36800 -63063.849600 -13239.23200
##                [,6]         [,7]          [,8]
## [1,]   25721.20000 -176546.0000 -2.948424e+08
## [2,]   11105.67600   34616.3200  1.876433e+08
## [3,]      -3.04360      -2.8920 -2.926237e+04
## [4,]       2.26308     -21.6324 -6.306385e+04
## [5,]     -10.96440       5.7720 -1.323923e+04
## [6,]     122.35360     213.9120  1.577973e+06
## [7,]     213.91200    2711.4400  8.848515e+06
## [8,] 1577973.24800 8848515.3600  3.742685e+10
```

(f)

Display only the elements in the diagonal of $D_{1/S}$.

```r
diag(V)
```

```
## [1] 5.957034e+07 1.131175e+06 1.216000e-01 9.426240e-01 5.681600e+00
## [6] 1.223536e+02 2.711440e+03 3.742685e+10
```

```r
D_1S = 1/diag(V)
D_1S
```

```
## [1] 1.678688e-08 8.840362e-07 8.223684e+00 1.060868e+00 1.760068e-01
## [6] 8.173033e-03 3.688077e-04 2.671879e-11
```

(g)

Display the output of $colMeans(Z)$ and $apply(Z, 2, var)$

```r
Z = Xc%*%D_1S
Z
```

```
##             [,1]
## [1,]  2.7951543
```

```
## [2,] -2.2734403
## [3,]  0.7707443
## [4,]  1.9832932
## [5,] -3.2757516
```

```
colMeans(Z)
```

```
## [1] -2.531308e-15
```

```
apply(Z, 2, var)
```

```
## [1] 7.059866
```

(h)

Compute the (population) correlation matrix $R = D_{1/S} V D_{1/S}$. Display the matrix R.

```
R = D_1S%*%V%*%D_1S
R
```

```
##          [,1]
## [1,] 5.647893
```

(i)

Confirm that R can also be obtained as $R = Z^T D Z$.

```
R = t(Z)%*%D%*%Z
R
```

```
##          [,1]
## [1,] 5.647893
```

Comments and Reflections

1. Math part, such as eigen-decomposition and singular value decomposition.
2. R programming.
3. Yes, I use Google and textbook resource.
4. Around 4 hours.
5. Problem 11