Problem Set 1: Matrix Algebra Review

Stat 154, Fall 2017, Prof. Sanchez

Due date: Tu Sep-12 (before midnight)

The purpose of this assignment is to apply in R some of the introductory material, giving you the opportunity to do some work with matrices and vectors.

Use an R markdown (`.Rmd`) file to write your code and answers. You can *knit* the `Rmd` file as html or pdf. Please submit both your `Rmd` and knitted file to bCourses. Make sure to include your name, and your lab section. No late assignments will be accepted.

Problem 1

Create the following matrices in R and compute the operations in parts (a) to (e)

$$\mathbf{A} = \begin{bmatrix} 1 & 2 & -3 \\ 4 & 0 & 1 \end{bmatrix}; \qquad \mathbf{B} = \begin{bmatrix} 2 & 3 & 4 \\ -1 & 2 & 0 \end{bmatrix}; \qquad \mathbf{C} = \begin{bmatrix} 0 & 1 & 0 \\ 4 & -1 & -2 \end{bmatrix}$$

a. $\mathbf{A} + \mathbf{B}$

b. $(\mathbf{A} + \mathbf{C}) + \mathbf{B}$

c. $\mathbf{A} - (\mathbf{C} + \mathbf{B})$

d. $-(\mathbf{A} + \mathbf{B})$

e. $(\mathbf{A} - \mathbf{B}) + \mathbf{C}$

Problem 2

Assume the following matrix $\mathbf{X}$

$$\mathbf{X} = \begin{array}{c|cccc} & Y & X_1 & X_2 & X_3 \\ a & 2 & 1 & 0 & 9 \\ b & 4 & 2 & 3 & 8 \\ c & 3 & 5 & 2 & 4 \\ d & 7 & 3 & 4 & 5 \\ e & 8 & 7 & 7 & 2 \\ f & 9 & 8 & 7 & 1 \end{array}$$

Note that $\mathbf{X}$ has row-names and column-names.

Create the matrix **X** in R (with row and column names), and find, via vector-matrix operations:

a. $\sum Y$

b. $\bar{X}_1$

c. $\sum Y X_2$

d. $\sum X_3^2 - (\sum X_3)^2/6$

e. the mean-centered matrix

f. the (sample) covariance matrix **S**

You are not allowed to use `sum()`, `apply()`, `sweep()`, nor loops. You can only use these operators: `+`, `-`, `*`, `/` and `%*%`.

Problem 3

Let **a** and **b** be vectors with given lengths and angle θ. Use R to compute their scalar product under the conditions:

a. $\|\mathbf{a}\| = 0.5; \quad \|\mathbf{b}\| = 4; \quad \theta = 45^o$

b. $\|\mathbf{a}\| = 4; \quad \|\mathbf{b}\| = 1; \quad \theta = 90^o$

c. $\|\mathbf{a}\| = 1; \quad \|\mathbf{b}\| = 1; \quad \theta = 120^o$

Problem 4

Refer to the Gram-Schmidt orthonormalization process described in the following wikipedia entry:

https://en.wikipedia.org/wiki/Gram%E2%80%93Schmidt_process

This procedure is a method for orthonormalizing a set of vectors in an inner product space. In other words, it allows you to find an orthogonal basis for a set of vectors.

Write an R function `proj()` for the *projection operator* given by:

$$proj_{\mathbf{u}}(\mathbf{v}) = \frac{< \mathbf{u}, \mathbf{v} >}{< \mathbf{u}, \mathbf{u} >}\mathbf{u}$$

This projector operator projects the vector **v** orthogonally onto the line spanned by vector **u**. Given two vectors `u` and `v`, you should be able to call your function like:

```r
proj(v, u)
```

Test `proj(u, v)` with $\mathbf{u} = (1, 3, 5)$ and $\mathbf{v} = (2, 4, 6)$

Problem 5

Once you have your function `proj()`, write R code to apply the Gram-Schmidt orthonormalization procedure to the following sets of vectors:

a. $\mathbf{x} = (1, 2, 3)$; $\mathbf{y} = (3, 0, 2)$; $\mathbf{z} = (3, 1, 1)$

b. $\mathbf{x} = (2, 1)$; $\mathbf{y} = (1, 2)$; $\mathbf{z} = (1, 1)$

Start by setting $\mathbf{u_1} = \mathbf{x}$, and report the set of vectors $\mathbf{u_k}$ and the orthonormalized vectors $\mathbf{e_k}$, for $k = 1, 2, 3$.

Problem 6

The length of a vector $\mathbf{x} = (x_1, x_2, \dots, x_n)$ in the n-dimensional real vector space $\mathbb{R}^n$ is usually given by the Euclidean norm:

$$\|\mathbf{x}\| = (x_1^2 + x_2^2 + \dots + x_n^2)^{1/2}$$

In many situations, the Euclidean distance is insufficient for capturing the actual distances in a given space. The class of p-norms generalizes the notion of *length* of a vector and it is defined by:

$$\|\mathbf{x}\|_p = (|x_1|^p + |x_2|^p + \dots + |x_n|^p)^{\frac{1}{p}}$$

where p is a real number ≥ 1.

Write a function `lp_norm()` that computes the L_p-norm of a vector. This function should take two arguments:

- `x` the input vector
- `p` the value for p
- Give `p` a default value of 1
- Allow the user to specify `p = "max"` to compute the L_∞-norm

You should be able to call `lp_norm()` like this:

```r
lp_norm(x)              # default p = 1
lp_norm(x, p = 2)
lp_norm(x, p = "max")   # L-max norm
```

Test your function `lp_norm()` with the following vectors and values for `p`:

a. `zero <- rep(0, 10)` and `p = 1`

b. `ones <- rep(1, 5)` and `p = 3`

c. `u <- rep(0.4472136, 5)` and `p = 2`

d. `u <- -40:0` and `p = 100`

e. `u <- 1:1000` and `p = "max"`

Problem 7

Show that the set $\{\mathbf{u_1}, \mathbf{u_2}, \mathbf{u_3}\}$ is orthonormal, with:

- $\mathbf{u_1} = \frac{1}{\sqrt{11}}(3, 1, 1)$
- $\mathbf{u_2} = \frac{1}{\sqrt{6}}(-1, 2, 1)$
- $\mathbf{u_3} = \frac{1}{\sqrt{66}}(-1, -4, 7)$

Hint: You need to check that they are orthogonal, and of unit norm.

Problem 8

For this problem, use the data set `USArrests` that comes in R.

a. Convert `USArrests` as a matrix; this will be the data matrix $\mathbf{X}$. Confirm that `X` is an object of class `"matrix"`

b. Let n be the number of rows of $\mathbf{X}$, and p the number of columns of $\mathbf{X}$

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

d. Compute $\mathbf{g} = \mathbf{X}^\mathsf{T}\mathbf{D}\mathbf{1}$ where $\mathbf{1}$ is a vector of 1's of length n. Display `g`.

e. Calculate the mean-centered matrix $\mathbf{X_c} = \mathbf{X} - \mathbf{1}\mathbf{g}^\mathsf{T}$. Display the output of `colMeans(Xc)`.

f. Compute the (population) variance-covariance matrix $\mathbf{V} = \mathbf{X}^\mathsf{T}\mathbf{D}\mathbf{X} - \mathbf{g}\mathbf{g}^\mathsf{T}$. Display the output of `V`.

g. Let $\mathbf{D}_{1/S}$ be a $p \times p$ diagonal matrix with elements on the diagonal equal to $1/S_j$, where S_j is the standard deviation for the j-th variable. Display only the elements in the diagonal of $\mathbf{D}_{1/S}$

h. Compute the matrix of standardized data $\mathbf{Z} = \mathbf{X_c}\mathbf{D}_{1/S}$ Display the output of `colMeans(Z)` and `apply(Z, 2, sd)`

i. Compute the (population) correlation matrix $\mathbf{R} = \mathbf{D}_{1/S}\mathbf{V}\mathbf{D}_{1/S}$. Display the matrix `R`

j. Confirm that $\mathbf{R}$ can also be obtained as $\mathbf{R} = \mathbf{Z}^\mathsf{T}\mathbf{D}\mathbf{Z}$