Math405: Learning From Data
Exam 1

(c) Slim Belhaiza, Ph.D.[a]

[a] *Semester 211*

---

1. Spectral Factorization & Diagonalization

a. Use spectral factorization principles to find the matrices X and Λ such that $A = X\Lambda X^t$:

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

b. Use diagonalization principles to find the matrices X and Λ such that $A = X\Lambda X^{-1}$:

$$A = \begin{pmatrix} 4 & 1 \\ 2 & 5 \end{pmatrix}$$

2. Cholesky Factorization

a. For the given matrix S, determine whether it is possible to obtain a Cholesky factorization $S = A^t A$:

$$S = \begin{pmatrix} 3 & 2 & 1 \\ 2 & 3 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

b. Perform a Cholesky factorization on S if it satisfies the conditions in (a).

3. Singular Vector Decomposition

a. Find the matrix A, such that $A = U\Sigma V^t$, for:

$$U = \frac{1}{\sqrt{5}}\begin{pmatrix} 2 & -1 \\ 1 & 2 \end{pmatrix}, \Sigma = \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{pmatrix} \text{ and } V^t = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}.$$

b. Detail the two rank-one matrices obtained A_1 and A_2, such that:

$$A = A_1 + A_2.$$

c. If $\sigma_1 = 5$ and $\sigma_2 = 2$, write A and A_1 is the closest rank-one matrix to A.

4. SVD with Python

The following Python code was written to perform the SVD factorization of a given matrix A_0 of size 3 by 5.

Complete the code such that you can compute and display the best rank one matrix approximation A_1.

Computing $A_0^t * A_0$ and $A_0 * A_0^t$

B = np.dot(A0.transpose(),A0)

C = np.dot(A0, A0.transpose())

Computing the eigenvalues and the eigenvectors of B and C

LambdaB, V= LA.eigh(B)

LambdaC, U= LA.eigh(C)

Picking the eigenvectors in V corresponding to the common eigenvalues between :LambdaB and LambdaC.

v1=np.dot(V, np.array([[0],[0],[1],[0],[0]]))

v2=np.dot(V, np.array([[0],[0],[0],[1],[0]]))

v3=np.dot(V, np.array([[0],[0],[0],[0],[1]]))

Computing sigma1,sigma2 and sigma3, square roots of the common eigenvalues

```
sig1=np.sqrt(LambdaC[0])
sig2=np.sqrt(LambdaC[1])
sig3=np.sqrt(LambdaC[2])
....
```