**MA227 (Assignment-6)**

1. Write a function *SelfSVD.m* that takes a real matrix $A$ and a small error tolerance $tol = 10^{-8}$, and returns the compact SVD of $A$. Use the following Algorithm to compute the SVD and Do not consider the singular values that are less than $tol$. Compare the results with the built-in function like `svd` in MATLAB.

   - $\lambda, V \leftarrow eig(A^T A)$.
   - $\sigma \leftarrow \sqrt{\lambda}$.
   - $\sigma \leftarrow sort(\sigma)$. (Sort the singular values from greatest to least)
   - $V \leftarrow sort(V)$. (Sort the eigenvectors the same way as in the previous step.)
   - $U(:, i) \leftarrow AV(:, i)/\sigma_i$.

2. Write a function *SelfLRA.m* that takes a real matrix $A \in \mathbb{R}^{m \times n}$ and a positive integer $k <<< \min\{m, n\}$, and returns a best low approximation $A_k$ of $A$. Use your function from Problem 1 to compute the compact SVD of $A$, then form the truncated SVD to return the $A_k$. Find the $||A - A_k||_2$ and $||A - A_k||_F$ and verify results from the theory.

3. Write a script file *SelfPCA.m* to execute the following commands.

   - Take $N = 100$.
   - Generates a $1 \times N$ vector $x1$ of values chosen from a Gaussian distribution of mean 0 and std deviation 1 (rand(1,N) command in MATLAB).
   - Generates a $1 \times N$ vector $x2$ of values chosen from a Gaussian distribution of mean 0 and std deviation 0.4 (0.4∗rand(1,N) command in MATLAB).
   - Generate matrix $A = [x1; x2]$
   - Rotate the data in matrix $A$ by $\pi/3$ anticlockwise and save it as Data matrix $D$
   - Plot the data in $D$ on $x1 - x2$ plane. (Take square axis($[-4\ 4]$ *and* $[-4\ 4]$).)
   - Find covariance matrix $Q = \frac{1}{N-1} D * D^T$.
   - Find and eigenvectors and eigenvalues of $Q$. ($[V, Di] = eig(Q)$ in MATLAB).
   - Sort the diagonal entries of $Di$ from greatest to least and save in a vector $eigenVal$.
   - Permute the columns of $V$ in the same way as the diagonal elements are permuted in the previous step.
   - Find principal components vectors $u1$. In Matlab: $u1 = V(:, 1)$.
   - Plot the vector $u1$ on the data plotted in one of the previous steps.