ITU Computer and Informatics Faculty
BLG 202E Numerical Methods in Computer Engineering
2020 - 2021 Spring
Homework #2

Handed out: 20.04.2021
Due: 05.05.2021 23:59

**Notes:**

- Prepare a report for this homework in PDF format. You can use word, latex or you can use your handwriting. The handwritten parts of the solutions must be presented on a paper legibly and scanned clearly.
- The written Python codes should be uploaded separately.
- Please do not forget to write your name and number at the top of each file you submitted.
- Please submit your report through Ninova e-Learning System. Another way of submission will not be accepted. Also, the late submissions will NOT be accepted.
- In the case of cheating and plagiarism, strong disciplinary action will be taken.
- For any questions about the Homework 2, contact T.A. Yunus Emre Cebeci (cebeci16@itu.edu.tr).

**Questions:**

1. (15 pt.) Consider the problem

$$\begin{aligned} x_1 - x_2 + 3x_3 &= 2 \\ x_1 + x_2 &= 4 \\ 3x_1 - 2x_2 + x_3 &= 1 \end{aligned}$$

   Carry out Gaussian elimination in its simplest form for this problem. What is the resulting upper triangular matrix?

   Proceed to find the solution by backward substitution.

2. (25 pt.) Let

$$A = \begin{pmatrix} 5 & 6 & 7 & 8 \\ 0 & 4 & 3 & 2 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & -2 \end{pmatrix}$$

   (a) The matrix A can be decomposed using partial pivoting as

$$PA = LU$$

   where U is upper triangular, L is unit lower triangular, and P is a permutation matrix. Find the 4 x 4 matrices U, L, and P.

   (b) Given the right-hand-side vector **b=**$(26, 9, 1, -3)^T$, find x that satisfies Ax = b. (Show your method: do not just guess.)

3. (30 pt.) Implement the Power Method in Python by writing a program that inputs a matrix $A \in R^{nxn}$ and an initial guess vector $v_0 \in R^n$. Use your code to find an eigenvector of matrix given below, starting with the initial guess vectors $v_0 = (1, 2, -1)^T$ and $v_0 = (1, 2, 1)^T$

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

   Report the first 5 iterates for each of the two initial vectors. Then find eigenvalues and eigenvectors of A (you can use numpy.linalg.eig[1]). Where do the sequences converge to? Why do the limits not seem to be the same?

   **Note:** You can use NumPy package in Python for the implementation.

4. (30 pt.) In this question you will play with one picture (see Figure 1) that can be found in homework's attachment in Ninova (clown.bmp).

   (a) Write a Python code for computing the truncated SVD of this image. Start with rank $r = 2$ and go up by powers of 2, to $r = 64$. For a compact presentation of your figures, use the matplotlib subplots[2] for each of the pictures, with 3(nrows) and 2(ncols) as the first two arguments.

   (b) Comment on the performance of the truncated SVD for each of the pictures. State how much storage is required as a function of r and how much storage is required for the original picture.

   **Note:** You can use NumPy, matplotlib, opencv packages in Python for the implementation.

[1] `https://numpy.org/doc/stable/reference/generated/numpy.linalg.eig.html`
[2] `https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.subplots.html`

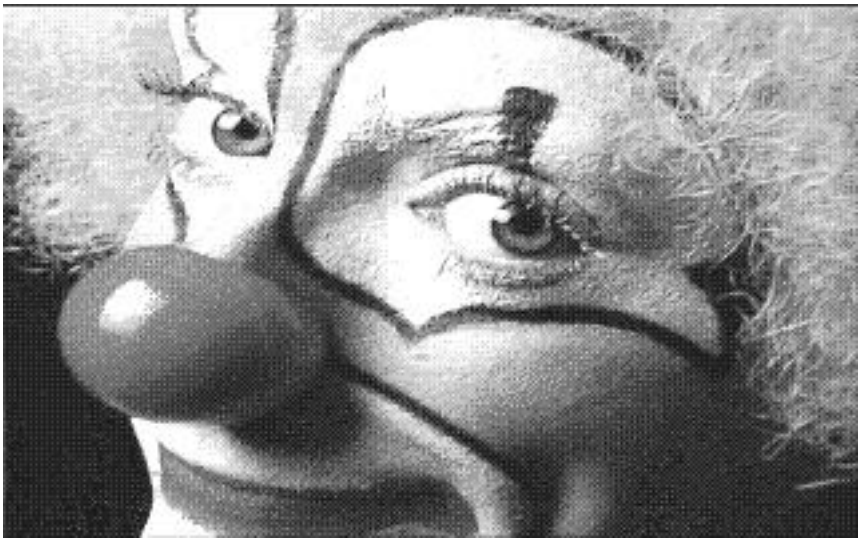

Figure 1: Clown image.