Course: Numerical Analysis for Machine Learning
Prof. E. Miglio - February 6th 2024
Duration of the exam: 2.5 hours.

Exercise 1
Consider the following matrix $X \in \mathbb{R}^{400\times 400}$ defined as:

$$X_{ij} = \begin{cases} 110 & 50 \leq i \leq 150, 30 \leq j \leq 130, \\ 150 & 50 \leq i \leq 150, 230 \leq j \leq 330, \\ 180 & 250 \leq i \leq 350, 70 \leq j \leq 170, \\ 220 & 250 \leq i \leq 350, 270 \leq j \leq 370, \\ 0 & \text{elsewhere}, \end{cases}$$

where $i, j = 1, \ldots, 400$.

1. Add a synthetic noise by sampling from a Gaussian distribution with zero mean and standard deviation $\sigma = 0.2$.
2. Implement the singular value truncation (SVT) algorithm and apply it to reconstruct the original matrix X from X_{noise}. Set a maximum number of iteration equal to 50, a tolerance on the increment equal to 10^{-6} and try to optimize the threshold on the singular values by trial and error. Compute the relative reconstruction error between X and the approximation $\hat{X}$, defined as:
$$\epsilon_R = \frac{||X - \hat{X}||_F}{||X||_F},$$
the rank $\hat{r}$ of $\hat{X}$ and visualize the true image and its approximation.
3. Find, by means of exact SVD, the value of k providing a relative reconstruction error equal to the one obtained in the previous point. Comment the results.
4. Repeat the previous point by means of randomized SVD. Comment the results.

Exercise 2
Consider the function
$$J(\mathbf{x}) = 2x^2 - 1.05x^4 + \frac{x^6}{6} + xy + y^2, \tag{1}$$
where $\mathbf{x} = (x, y)^T$.

1. Compute and characterize the stationary points of (1). Moreover compute the corresponding function values.
2. Plot the function $J(\mathbf{x})$ as a contourplot and as a surface (*Hint:* use the commands `contour` and `surf` from Matplotlib).
3. Implement the Newton method for computing the minimum of a function. The input parameters should be: the function J, the gradient of J, the Hessian of J, the initial guess, the tolerance and the maximum number of iterations. The stopping criterium should be $|J(\mathbf{x}^{(k+1)}) - J(\mathbf{x}^{(k)})|$. The output should be the history of the iterates $\mathbf{x}^{(k)}$ and the history of the corresponding functional values $J(\mathbf{x}^{(k)})$.
4. Apply the code developed at the previous point to (1). Use a tolerance $\epsilon = 10^{-6}$ and set the maximum number of iterations equal to 100. Consider the following initial guesses $\mathbf{x}^0 = (2.5, -2.5)$, $\mathbf{x}^0 = (0.8, -2.5)$ and $\mathbf{x}^0 = (-0.4, -2.5)$. For each initial point plot the convergence history on the contourplot and the history of the function values. Comment the results.

Exercise 3

Consider the following matrix

$$A = \begin{bmatrix} 3 & 6 \\ 0 & 1 \\ 4 & 8 \end{bmatrix}. \tag{2}$$

1. Compute, by hand, the QR factorization of the matrix A using the Gram-Schmidt procedure.
2. Use the computed factorization to build the projection matrix on the range(A) (or column space of A).
3. How can you use the projection matrix to determine whether the vector $\mathbf{b} = [1, 1, 0]^T$ belongs to the column space of A ?
4. Use the QR factorization to find a solution (or best-fit solution) to $A\mathbf{x} = \mathbf{b}$.
5. Does the solution exist to $A^T\mathbf{x} = \mathbf{c}$ where $\mathbf{c} = [2, 2]^T$? If no solution exists, find the best-fit. If one or more solution exist, find the one for which $\|\mathbf{x}\|$ is as small as possible.