Work Integrated Learning Programmes Division
M. Tech. in Data Science and Engineering

Assignment 2

DSECL ZC416 - Mathematical Foundations for Data Science

Instructions

1. Use Python for Q1 and Q2. We recommend Octave for Q3 due to availability of built in function 'solve'.
2. Attach only the relevant data in your submission as per the deliverables mentioned.
3. By random entries, we mean a system generated random number. No marks would be awarded for deterministic entries.
4. This is not a group activity. Each student should do the problems and submit individually.
5. Submissions beyond 24th of February, 2022, **17.00** hrs would not be graded
6. Assignments sent via email would not be accepted
7. Copying is strictly prohibited. Adoption of unfair means would lead to disciplinary action.
8. Assignment have to be scanned as a pdf document and uploaded on canvas. File name should be your bitsid.pdf

---

Answer all the questions

Q1) Gram-Schmidt Algorithm and QR decomposition (5 marks)

i) Write a code to generate a random matrix $\mathbf{A}$ of size $m \times n$ with $m > n$ and calculate its Frobenius norm, $\| \cdot \|_F$. The entries of $\mathbf{A}$ must be of the form $r.dddd$ (example 5.4316). The inputs are the positive integers m and n and the output should display the the dimensions and the calculated norm value.
Deliverable(s) : The code with the desired input and output (0.5)

ii) Write a code to decide if Gram-Schmidt Algorithm can be applied to columns of a given matrix $\mathbf{A}$ through calculation of rank. The code should print appropriate messages indicating whether Gram-Schmidt is applicable on columns of the matrix or not.
Deliverable(s) : The code that performs the test. (1)

iii) Write a code to generate the orthogonal matrix Q from a matrix $\mathbf{A}$ by performing the Gram-Schmidt orthogonalization method. Ensure that $\mathbf{A}$ has linearly independent columns by checking the rank. Keep generating $\mathbf{A}$ until the linear independence is obtained.
Deliverable(s) : The code that produces matrix $\mathbf{Q}$ from A (1)

iv) Write a code to create a QR decomposition of the matrix $\mathbf{A}$ by utilizing the code developed in the previous sub-parts of this question. Find the matrices $\mathbf{Q}$ and $\mathbf{R}$ and then display the value $\|\mathbf{A} - (\mathbf{Q}.\mathbf{R})\|_F$, where $\|\cdot\|_F$ is the Frobenius norm. The code should also display the total number of additions, multiplications and divisions to find the result.
Deliverable(s) : The code with the said input and output. The results obtained for $\mathbf{A}$ generated with $m = 7$ and $n = 5$ with random entries described above. (2.5)

Q2) Gradient Descent Algorithm (2 marks)

i) Consider the last 4 digits of your mobile number (Note : In case there is a 0 in one of the digits replace it by 3). Let it be $n_1n_2n_3n_4$. Generate a random matrix A of size $n_1n_2 \times n_3n_4$. For example, if the last four digits are 2311, generate a random matrix of size 23×11. Write a code to calculate the l_∞ norm of this matrix.
Deliverable(s) : The code that generates the results. (0.5)

ii) Generate a random vector b of size $n_1n_2 \times 1$ and consider the function $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{Ax} - \mathbf{b}\|_2^2$ where $\|\cdot\|_2$ is the vector ℓ_2 norm. Its gradient is given to be $\nabla f(\mathbf{x}) = \mathbf{A}^\top\mathbf{Ax} - \mathbf{A}^\top\mathbf{b}$. Write a code to find the local minima of this function by using the gradient descent algorithm (by using the gradient expression given to you). The step size τ in the iteration $\mathbf{x}_{k+1} = \mathbf{x}_k - \tau\nabla f(\mathbf{x}_k)$ should be chosen by the formula

$$\tau = \frac{\mathbf{g_k^T g_k}}{\mathbf{g_k^T A^T A g_k}}$$

where $\mathbf{g}_k = \nabla f(\mathbf{x_k}) = \mathbf{A}^\top\mathbf{A}\mathbf{x_k} - \mathbf{A}^\top\mathbf{b}$. The algorithm should execute until $\|\mathbf{x_k} - \mathbf{x_{k-1}}\|_\mathbf{2} < \mathbf{10^{-4}}$.
Deliverable(s) : The code that finds the minimum of the given function

and the expression for τ. The values of $\mathbf{x_k}$ and $f(\mathbf{x_k})$ should be stored in a file. (1)

iii) Generate the graph of $f(\mathbf{x_k})$ vs k where k is the iteration number and $\mathbf{x_k}$ is the current estimate of x at iteration k. This graph should convey the decreasing nature of function values.
Deliverable(s) : The graph that is generated. (0.5)

Q3) Critical Points of a function (3 marks)

i) Generate a third degree polynomial in x and y named $g(x, y)$ that is based on your mobile number (Note : In case there is a 0 in one of the digits replace it by 3). Suppose your mobile number is 9412821233, then the polynomial would be $g(x, y) = 9x^3 - 4x^2y + 1xy^2 - 2y^3 + 8x^2 - 2xy + y^2 - 2x + 3y - 3$, where alternate positive and negative sign are used.
Deliverable(s) : The polynomial constructed should be reported. (0.5)

ii) Write a code to find all critical points of $g(x, y)$. You may use built in functions like 'solve' (or other similar functions) in Octave/Matlab to find the critical points .
Deliverable(s) : The code that finds the critical points along with the display of all the calculated critical points. (1)

iii) Write a code to determine whether they correspond to a maximum, minimum or a saddle point.

Deliverable(s) : The code that identifies the type of critical points. The critical points and their type must be presented in the form of the table generated by code for the above polynomial. (1.5 marks)