UTSA
CS 6243, EE 4463, EE 5573
Spring 2025
Assignment: HW-2
Topic: Numerical Optimization for ML - LS, GD

I. Assignment Instructions

- **Submission:** Submit your report in the designated drop box on CANVAS by the corresponding assignment deadline.
- **Format:** Reports must be typed and uploaded as a PDF. Handwritten reports will not be graded. Any requested code should be presented at the end of your report in an appendix titled "Code." All code must include appropriate comments. Display equations and figures must be numbered, labeled, and captioned accordingly.
- **References:** You may use any source (notes, books, online), but all sources must be cited in a References List at the end of your report.
- **Originality:** You are not allowed to outsource this assignment (or parts of it) to another intelligent entity (human or AI – this includes ChatGPT and similar tools). You can still browse the internet and other sources, find information and ideas, and use them to compile your own solutions. Except for explicitly cited content, by submitting your work you verify and commit that it is your own intellectual work.
- **Grading:** Your assignment will be graded based on three equally weighted factors: Correctness, Completeness, and Clarity. For each problem, each criterion will be rated as follows: 100% (Excellent), 90%, 70%, 50% (Fair Effort), 30%, 10%, or 0% (Missing Effort). For example, a correct final result (100%), with almost complete method (90%), and almost clear presentation (90%) would receive $100\% \times 90\% \times 90\% = 81\%$. If the total points for the problem are 18, this would earn you 14.6 points (rounded to the nearest first decimal).
- **Teams:** Work in your determined project/assignment teams (see CANVAS). Each team member must individually submit a copy of the same team report on CANVAS. Only team members who make a submission before the deadline will receive a grade. It is expected that all team members contribute equally to the assignment. In case of doubt, the instructor will assign individual grades based on individual examination during office hours. In case members submit distinct reports, each team member will be graded based on their own submission.
- **Task Terminology:**

  Present: Show only the final result in math.

  Derive: Show all the necessary mathematical steps leading to final result.

  Compute: Write the Python code and present it.

  Plot: Write python code that computes and plots; present both the code and the plots.

  Discuss: discuss in words and reason/justify in detail (no need for math or code).

- All figures must have the correct numbers, descriptive captions, axis labels, and legends for every curve included. Color, marker, and line-style combination should be such that distinct curves are discernible even in gray-scale printing.

II. Problems

Problem 0 (0 points)
Load arrays A, y, and x_{init} from the csv files provided in this assignment. Denote by N and K the rows and columns of A, respectively. Consider the least squares (LS) problem $x_{opt} = \arg\min_{x \in \mathbb{R}^K} L(x)$, where $L(x) = \|Ax - y\|_2^2$.

Problem 1: Derivations (4 points; 1 point per task)
In this problem, all derived expressions should be presented as functions of x, A, y, or any SVD factors; not numerically.

- Derive the gradient of $L(x)$, $g(x)$.
- Derive the Hessian of $L(x)$, H.
- Derive the Lipschitz constant C.
- Derive the closed form solution of LS by means of SVD: both optimal argument x_{opt} and value L_{opt}.

Problem 2: Computations (6 points; 3 points per task)
In this problem, perform numerical computations in python using the values provided in the csv file.

- Compute H and C.
- Compute LS solutions x_{opt} and L_{opt} by means of SVD.

Problem 3: LS GD with Exact Line Search (ELS) (20 points; 5 points per task)
Implement in Python GD for the LS problem. Initialize at the given x_{init} and run 30 iterations, generating $x_1, x_2, \ldots, x_{30}$. At each iteration n, identify the step size γ_n by means of the ELS method from the class notes.

- **Plot:** In Fig. 1 of your report, plot $L(x_n)$ vs iteration index $n = 1, 2, \ldots, 30$, for the GD with ELS (blue solid line). In the same figure plot L_{opt} as a horizontal benchmark (black dashed line).
- **Discuss** Fig. 1.
- **Plot:** In Fig. 2 of your report, plot γ_n vs iteration index $n = 1, 2, \ldots, 30$, for the GD with ELS. In the same figure, plot $1/C$ as a horizontal benchmark (black dashed line).
- **Discuss** Fig. 2.

Problem 4: LS GD with Fixed Step Size (FSS) (10 points; 5 points per task)
Implement in Python GD for the LS problem. Initialize at the given x_{init} and run 30 iterations, generating $x_1, x_2, \ldots, x_{30}$. In all iteration steps, use the same fixed step size γ, set to $p \times 1/C$, where C is the Lipschitz constant of LS and p is a scaling factor.

- **Plot:** In Fig. 3 of your report, plot $L(x_n)$ vs iteration index $n = 1, 2, \ldots, 30$, for the GD with FSS, for $p = 0.1, 0.5, 1, 1.5, 2$ (5 curves). In the same figure plot L_{opt} as a horizontal benchmark (black dashed line).
- **Discuss** Fig. 3. How does each value of p affect the convergence?

Problem 5: LS GD with Backtracking Line Search (60 points; 5 points per task)
Implement in Python GD for the LS problem. Initialize at the given x_{init} and run 30 iterations, generating $x_1, x_2, \ldots, x_{30}$. At each iteration n, identify the step size γ_n by means of the BLS method from the class notes, with parameters η and c.

- **Plot:** In Fig. 4 of your report, plot $L(x_n)$ vs iteration index $n = 1, 2, \ldots, 30$, for the GD with BLS, for $c = 0.1$ and $\eta = 0.1, 0.2, 0.5, 0.9$ (4 curves). In the same figure plot L_{opt} as a horizontal benchmark (black dashed line).
- **Discuss** Fig. 4. How does convergence vary across η?
- **Plot:** In Fig. 5 of your report, plot runtime bars versus η (for fixed $c = 0.1$). That is, each bar corresponds to the runtime of GD from initialization to termination, for a given value of η.
- **Discuss** Fig. 5. How does η affect runtime?
- **Plot:** In Fig. 6 of your report, plot γ_n vs iteration index $n = 1, 2, \ldots, 30$, for the GD with BLS, for $c = 0.1$ and $\eta = 0.1, 0.2, 0.5, 0.9$ (4 curves). In the same figure, plot $1/C$ as a horizontal benchmark (black dashed line).
- **Discuss** Fig. 6.
- **Plot:** In Fig. 7 of your report, plot $L(x_n)$ vs iteration index $n = 1, 2, \ldots, 30$, for the GD with BLS, for $\eta = 0.5$ and $c = 0.01, 0.1$ (2 curves). In the same figure plot L_{opt} as a horizontal benchmark (black dashed line).
- **Discuss** Fig. 7. How does convergence vary across c?
- **Plot:** In Fig. 8 of your report, plot runtime bars versus c (for fixed $\eta = 0.5$). That is, each bar corresponds to the runtime of GD from initialization to termination, for a given value of c.
- **Discuss** Fig. 8. How does c affect runtime?
- **Plot:** In Fig. 9 of your report, plot γ_n vs iteration index $n = 1, 2, \ldots, 30$, for the GD with BLS, for $\eta = 0.5$ and $c = 0.01, 0.1$ (2 curves). In the same figure, plot $1/C$ as a horizontal benchmark (black dashed line).
- **Discuss** Fig. 9.