Homework 7

Important submission instructions:

- All explanations and answers must be clearly and neatly written. Explain each step in your solution. Your solutions should make very clear to the instructor that you understand all of the steps and the logic behind the steps.
- You are allowed to discuss the homework problems with other students (in particular via Canvas discussion board). However, what you hand in should reflect your own understanding of the material. You are NOT allowed to copy solutions from other students or other sources. Also, please list at the end of the problem set the sources you consulted and people you worked with on this assignment.
- The final document should be saved and submitted as a single .pdf file, and please be sure all problem solutions are presented starting from the first to the last (that is, the first solution must correspond to problem 1 and the second to problem 2 and so on).
- Typed submissions (for example in LaTeX) will be positively considered in the grade. Overleaf is an easy avenue to start learning LaTeX. See the tutorials at https://www.overleaf.com/learn
- Be neat. There should not be text crossed out. Do not hand in your rough draft or first attempt, copy and re-copy your solution if needed.
- Very important: Honor code applies fully. You must submit your own work only. In particular, it is prohibited to post the following problems on any website/forum or any other virtual means (for example Chegg).

Particular instructions for this HW:

Computer code is not an acceptable answer to any of the questions. Instead, the answers must be provided in formulas, explanations, figures, comments, and conclusions. It is important to note that any computer code used to arrive at the answers must be submitted as an appendix. By submitting any computer code used as an appendix, you allow the instructor to review your work and verify that the answers provided are accurate. However, it is important to note that the computer code will not be graded or considered part of your final score.

Problem 0

Qualitatively reproduce the plots from the section "Minimizing functions of one variable" (slides `GM.pdf`).

Problem 1

Consider the problem of minimizing $f(x) = \|ax - b\|^2$, where a and b are vectors in $\mathbb{R}^n$, $a \neq 0$, and $x \in \mathbb{R}$.

a. Derive an expression (in terms of a and b) for the solution to this problem.

b. Suppose we use an iterative algorithm

$$x^{(k+1)} = x^{(k)} - \alpha f'(x^{(k)}),$$

where f' is the derivative of f to solve the problem. Find the largest range of values of α (in terms of a and b) for which the algorithm converges to the solution for all starting points $x^{(0)}$.

Hint: See theorem "Convergence cuadratic case - Fixed step size" in slides.

Problem 2

Given $f : \mathbb{R}^n \to \mathbb{R}$, consider the iterative algorithm

$$x^{k+1} = x^k + \alpha_k d^k,$$

where $d^1, d^2, \ldots$ are given vectors in $\mathbb{R}^n$ and

$$\alpha_k = \arg\min_{\alpha \geq 0} f(x^k + \alpha d^k).$$

Show that $x^{k+1} - x^k$ is orthogonal to $\nabla f(x^{k+1})$ for all k.

Problem 3

Consider the problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \ \frac{1}{2}\|Ax - b\|_2^2,$$

where $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ are given in the text files `A.txt` and `b.txt` attached with this homework [1].

1. Write the optimality condition for this problem. Express and compute the exact solution x^*.

2. Starting from $x^0 = (0, 0, ...0)^\top$, use the steepest descent method to find an approximate solution. Use the following stepsizes: a) fixed step $\alpha = \frac{1}{\lambda_{\max}(A^\top A)}$, b) exact line search, d) Armijo's rule.

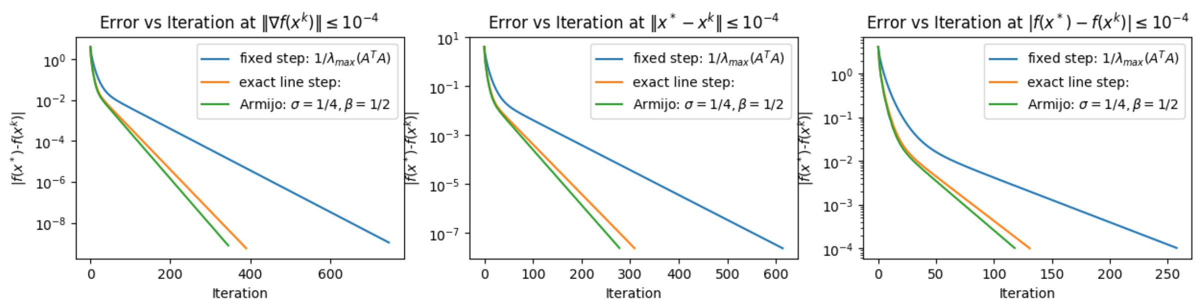
[1] This is a comma separated text file. For example, In MATLAB, you can use the command `A = dlmread('A.txt', ',', 0, 0);` to read it.

3. Try different stopping conditions: a) $\|\nabla f(x^k)\| \le \epsilon$, b) $\|x^* - x^k\| \le \epsilon_0$, c) $|f(x^*) - f(x^k)| \le \epsilon_1$. Use $\epsilon = 10^{-4}$. Choose suitable ϵ_0 and ϵ_1.

4. Plot (*use the appropriate log scale whenever suitable*)

   - the norm of the gradient $\|\nabla f(x^k)\|$ vs iteration number k.
   - the relative error $\|x^* - x^k\|/\|x^*\|$ vs. iteration number k.
   - the error $|f(x^*) - f(x^k)|$ vs iteration number k.
   - the stepsize vs iteration number k.
   - the error $|f(x^*) - f(x^k)|$ vs the error $|f(x^*) - f(x^{k-1})|$ to visualize the rate of convergence.

Summarize all possibilities in few plots. For example (plots are just illustrative):



Problem 4

Consider the problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; f(x) = f(x_1, x_2) = (x_2 - x_1^2)^2 + \delta(1 - x_1)^2 \,,$$

where $0 < \delta \ll 1$. Set $\delta = 0.01$.

We know that the minimizer of this function is $x^* = (1,1)^\top$.

a)

1. Starting from different points: $x^0 = (-0.8, 0.8)^\top$, $x^0 = (0,0)^\top$ and $x^0 = (1.5, 1)^\top$ use the steepest descent method to find an approximate solution of the minimization problem. Use the following stepsize methods: Armijo rule, exact[2] line search and decreasing $c/\sqrt{k}$, where k is the iteration number, where c is a constant that needs to be tuned. Use tolerance $\epsilon = 10^{-5}$ for the stopping condition $\|\nabla f(x^k)\| \le \epsilon$.

2. Plot the error $|f(x^k) - f(x^*)|$ vs. iteration number for each stepsize selection method used. Check the convergence rate is the one you expect for this method.

[2] Use a 1D numerical method to approximate the exact line search, since we have no analytical solution for the line search problem.

b)

Now apply Newton's method. Use same stepsize selection methods, the same tolerance and same initial points.

1. Plot the error $|f(x^k) - f(x^*)|$ vs. iteration number. Check the convergence rate is the one you expect for this method.

2. Compare the convergence rates of this method vs. the steepest method.

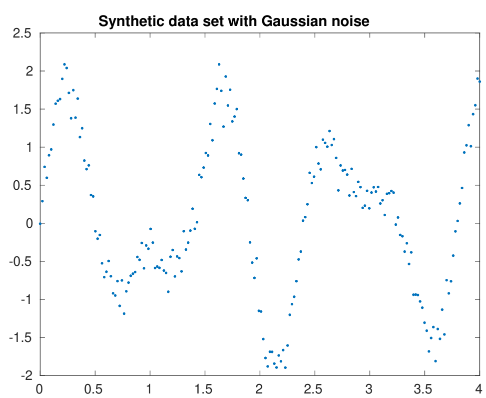
Problem 5

Consider the non-linear LS problem below: fit a model with four parameters ($n = 4$) of the form

$$\eta(t) = a_1 \sin(2\pi a_2 t) + a_3 \sin(2\pi a_4 t)$$

to a given (t_i, y_i) synthetic data set such as the generated below.

```
t = linspace(0, 4, n)';
m = @(t) (a*sin(2*pi*b*t) + c*sin(2*pi*d*t));
y = m(t) + 0.15*randn(n,1);
plot(t, y, '.')
title('Synthetic data set with Gaussian noise')
```



1. The objective function $\frac{1}{2} f^\top f$ where f is an $m \times 1$ vector with elements $f_i = \eta(t_i) - y_i$.

2. The gradient of the objective function $\mathbb{J}^\top f$ where $\mathbb{J}$ is the $m \times n$ Jacobian of f, $J_{ij} = \frac{\partial f_i}{\partial a_j}$.

3. The Hessian of the objective function is $\mathbb{J}^\top \mathbb{J} + \sum_{i=1}^m f_i \nabla^2 f_i$.

4. Use a line search method with a Steepest direction to fit a 4-parameter model to a synthetic data set like the one shown above. Start close from the true parameter values. (see slides "Application: Solving a non-linear system of equations using gradient descent").

5. Use regression to empirically estimate the convergence order and factor.

6. The Gauss-Newton direction is obtained by ignoring the second term in the Hessian of the objective. This is justified because the norm of f tends to zero as the solution is approached. Use a Gauss-Newton method to fit the data set above. Comment.

Problem 6

Consider the following unconstrained problem with $f : X \to \mathbb{R}$, $X \subseteq \mathbb{R}^n$ and $a_j \in \mathbb{R}^n$ constant.

$$\text{minimize} \;\; f(x) = -\sum_{j=1}^{m} \log(1 - a_j^\top x) - \sum_{i=1}^{n} \log(1 + x_i) - \sum_{i=1}^{n} \log(1 - x_i)\,,$$

where $X = \{x \in \mathbb{R}^n : a_j^\top x < 1,\ j{=}1, 2, .., m,\ \ |x_i| < 1,\ i{=}1, 2, .., n\}$. Generate a random matrix $A \in \mathbb{R}^{m \times n}$ whose rows serve as the vectors a_j. Always generate the same radom matrix by using a seed command. In MATLAB: the command `rng(1)` before generating the matrix (use a Normal randon variate generator). In this problem, the optimal solution is not known in advance, therefore to plot the error $|f(x^k) - f(x^*)|$ you need first to determine with high accuracy x^*.

1. Derive expressions for the gradient and Hessian *using the chain rule*. For this function it is harder to derive those expressions by computing partials directly.

2. Starting from $x^0 = (0, 0, ...0)^\top$, use the steepest descent method to find an approximate solution. Find the number of iterations required for $\|\nabla f\| < 10^{-3}$. Use Armijo rule only. Experiment with different size of the parameters, starting with $\sigma = 1/10$ and $\beta = 1/2$. Also try different sizes of m and n starting from 20 and 10 respectively. Hint: For this problem, the conditions $a_j^\top x < 1$ and $|x_i| < 1$ must be satisfied at every step. Observe that, for small enough stepsize α, the conditions can be satisfied as long as the current x satisfies the conditions.

3. Plot the error $|f(x^k) - f(x^*)|$ and the stepsize vs iteration number. Check convergence rate.

4. Repeat steps 2 and 3 using Newton's method. Compare convergence rates of both methods. Show the comparison using plots.