Problem Description

Consider a composite optimization problem

$$\min_x f(x) + h(x), \tag{0.1}$$

where $f(x)$ is differentiable and $h(x)$ is a function whose proximal operator is easily available. Both $f(x)$ and $h(x)$ may be nonconvex. Let $h(x)$ be a proper and close function, and $\inf\limits_{x\in dom(h)} h(x) > -\infty$. The *proximal operator* of $h(x)$ is defined as

$$\text{prox}\ \ h(x) = \arg\min_u \quad h(u) + \frac{1}{2}\|u-x\|^2. \tag{0.2}$$

Then starting from a suitable initial point x_0, the gradient method (key iterative strategy) is performed as

$$x_{k+1} = \text{prox}\ \ h(x_k - t_k\nabla f(x_k)), \tag{0.3}$$

where t_k is a chosen step size.

1. Design a software package for the proximal gradient method using C++ or Matlab.
2. Strategies for choosing the step size t_k:
   - backtracking line search to achieve the Armijo condition
   - backtracking line search to achieve a non-monotone condition using the **BB** step size
3. A few typical scenarios of $f(x)$ are listed as follows. Here x can be either a vector or a matrix. Your code are supposed to be support at least two of them. Calculate and written down their gradient.
   - Least Square: $f(x) = \frac{1}{2}\|Ax-b\|_2^2$ or $f(x) = \frac{1}{2}\|Ax-b\|_F^2$
   - Logistic regression:
     $$f(x) = \frac{1}{m}\sum_{i=1}^{m}\log(1+\exp(-b_i a_i^T x)).$$
   - Other functions are also welcome. (It depends on yourself).
4. A few typical scenarios of $h(x)$ are listed as follows. Your code should support at least three of them. Other functions not in the list are also welcome. Note that the variable x can either be a vector or a matrix. Furthermore, x may also be divided into a few blocks. Calculate and written down the corresponding *proximal operator* explicitly

- General functions:
  - vectors: ℓ_0, ℓ_1, ℓ_2, ℓ_∞ norm.
  - matrices: $\ell_{1,2}$, $\ell_{2,1}$ norm, or nuclear norm,
  - 1D, 2D TV-norm
- Indicator functions $\mathbf{1}_\Omega(x)$:
  - vectors: ℓ_0, ℓ_1, ℓ_2, ℓ_∞-ball
  - vectors: simple box,
  - matrices: nuclear norm ball.

5. **Requirements**:

   (a) Generate both random data and collect a few real data to test the codes.

   (b) Prepare a report including
   - detailed description of the design of the package
   - detailed answers to each question
   - tables of numerical results (including the total number of iterations, the optimality measures, the CPU time and etc) and their interpretation

   (c) Pack the report and all of your codes in one file named as "Final-PGM-StudentID-Name-date.zip" and upload it "course.zju.edu.cn"

   (d) If you get significant help from others on one routine, write down the source of references at the beginning of this routine.

   (e) Please talk to the teaching assistant(GU HAO in Dingding Group) if you have questions.

   (f) **Deadline: June 25, 2024.**