Homework 2

Convex Optimization 10-725

Due Friday, September 28 at 11:59pm

Submit your work as a single PDF on Gradescope. Make sure to prepare your solution to each problem on a separate page. (Gradescope will ask you select the pages which contain the solution to each problem.)

Total: 86 points
v1.1

1 Subgradients and Proximal Operators (18 pts) [Wenbo]

(a) Recall that subgradient can be viewed as a generalization of gradient for general functions. Let f be a function from $\mathbb{R}^n$ to $\mathbb{R}$. The subdifferential of f at x is defined as $\partial f(x) = \{g \in \mathbb{R}^n : g \text{ is a subgradient of } f \text{ at } x\}$.

(i, 2 pts) Show that $\partial f(x)$ is a convex and closed set.

(ii, 2 pts) Show that $\partial f(x) \subseteq N_{\{y:f(y)\leq f(x)\}}(x)$, where recall $N_C(x)$ denotes the normal cone to a set C at a point x. Give an example to show that this containment can be strict.

(iii, 2 pts) Let $p, q > 0$ such that $\frac{1}{p} + \frac{1}{q} = 1$. Consider the function $f(x) = ||x||_p = (\sum_{i=1}^n x_i^p)^{1/p}$. Show that $\forall x, y$:

$$x^T y \leq ||x||_p ||y||_q.$$

The above inequality is known as Hölder's inequality. Hint: you may use the dual representation of the ℓ_p norm, namely, $||x||_p = \max_{||z||_q \leq 1} z^T x$.

(iv, 3 pts) Use Hölder's inequality to show that for $f(x) = ||x||_p$, its subdifferential is $\partial f(x) = \text{argmax}_{||z||_q \leq 1} z^T x$. (You are not allowed to use the rule for the subdifferential of a max of functions for this problem.)

(b) The proximal operator for function $h : \mathbb{R}^n \mapsto \mathbb{R}$ and $t > 0$ is defined as:

$$\text{prox}_{h,t}(x) = \underset{z}{\text{argmin}} \frac{1}{2}||z - x||_2^2 + th(z)$$

Compute the proximal operators $\text{prox}_{h,t}(x)$ for the following functions.

(i, 2 pts) $h(z) = \frac{1}{2} z^T A z + b^T z + c$, where $A \in \mathbb{S}_+^n$.

(ii, 2 pts) $h(z) = \sum_{i=1}^n z_i \log z_i$, where $z \in \mathbb{R}_{++}^n$. Hint: you may refer to the Lambert W-function when solving for the proximal.

(iii, 2 pts) $h(z) = ||z||_2$.

(iv, 3 pts) $h(z) = ||z||_0$, where $||z||_0$ is defined as $||z||_0 = |\{z_i : z_i \neq 0, i = 1, \ldots, n\}|$.

(**Bonus**) $h(z) = \sum_{i=1}^n \lambda_i |z|_{(i)}$, where $z \in \mathbb{R}^n$, $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$, and $|z|_{(1)} \geq |z|_{(2)} \geq \ldots \geq |z|_{(n)}$ are the ordered absolute values of the coordinates of z. This is called the sorted-l_1 norm of z. Hint: you may consider the relation of the sign of x_i and z_i; and sort the entries in x and consider their correspondence with the sorted entries in $|z|$.

2 Properties of Proximal Mappings and Subgradients (18 points) [Akash]

(a, 4pts) Prove one direction of the finite pointwise maximum rule for subdifferentials: The subdifferential of $f(x) = \max_{i=1,\dots,n} f_i(x)$, for convex f_i, $i = 1, \dots, m$, satisfies

$$\partial f(x) \supseteq \text{conv}\left(\bigcup_{i:f_i(x)=f(x)} \partial f_i(x)\right). \tag{1}$$

(b, 4pts) Recall the definition of the proximal mapping: For a function h, the proximal mapping prox_t is defined as

$$\text{prox}_t(x) = \underset{u}{\text{argmin}} \frac{1}{2t}\|x-u\|_2^2 + h(u). \tag{2}$$

Show that $\text{prox}_t(x) = u \Leftrightarrow h(y) \geq h(u) + \frac{1}{t}(x-u)^\top(y-u) \quad \forall y$.

(c, 5 pts) Show how we can compose an affine mapping with the proximal operator. That is, assuming $f(x) = g(Ax+b)$, where $x \in R^n$, $A \in R^{m^* n}$, and $b \in R^m$, and also assuming $AA^T = aI_m$, for some scalar $a > 0$, then

$$\text{prox}_f(x) = x + \frac{1}{a}A^T\left(\text{prox}_{ag}(Ax+b) - Ax - b\right) \tag{3}$$

Hint: you may find it helpful to reparameterize $g(Ax+b)$ as $g(z)$ with the constraint that $z = Ax+b$, and then apply this constraint as a Lagrange multipler.

(d, 5 pts) Show that if $\forall y \in \text{dom}(g)$, $\partial g(\text{prox}_f(y)) \supseteq \partial g(y)$, then

$$\text{prox}_{f+g}(x) = \text{prox}_f(\text{prox}_g(x)) \tag{4}$$

Hints:

1. Consider $\text{prox}_{f+g}(x)$, $\text{prox}_g(x)$, and $\text{prox}_f(\text{prox}_g(x))$.
2. The solution of the proximal can be characterized as:

$$u = \text{prox}_h(x) := \underset{u}{\text{argmin}} \frac{1}{2}\|u-x\|_2^2 + h(u) \iff 0 \in u - x + \partial h(u)$$

3. $\partial(f+g) = \partial f + \partial g$

3 Convergence Rate for Proximal Gradient Descent (20 pts) [Po-Wei]

In this problem, you will show the sublinear convergence for gradient descent and proximal gradient descent, which was presented in class.

To be clear, we assume that the objective $f(x)$ can be written as $f(x) = g(x) + h(x)$, where

(A1) g is convex, differentiable, and $\text{dom}(g) = \mathbb{R}^n$.

(A2) ∇g is Lipschitz, with constant $L > 0$.

(A3) h is convex, not necessarily differentiable, and we take $\text{dom}(h) = \mathbb{R}^n$ for simplicity.

(a) We begin with the simple case $f(x) = g(x)$; that is, $h(x) = 0$ and can be ignored. We will prove that the gradient descent converges sublinearly in this case. As a reminder, the iterates of gradient descent is computed by

$$x^+ = x - t\nabla g(x), \tag{5}$$

where x^+ is the iterate succeeding x. Henceforth, we will set $t = 1/L$ for simplicity.

(i, 3pt) Show that

$$g(x^+) - g(x) \leq -\frac{1}{2L}\|\nabla g(x)\|^2.$$

That is, the objective value is monotonically decreasing in each update. This is why gradient descent is called a "descent method."

(ii, 3pt) Using convexity of g, show the following helpful inequality:

$$g(x^+) - g(z) \leq \nabla g(x)^T(x - z) - \frac{1}{2L}\|\nabla g(x)\|^2, \quad \forall z \in \mathbb{R}^n.$$

(iii, 2pt) Show that

$$g(x^+) - g(x^\star) \leq \frac{L}{2}\left(\|x - x^\star\|^2 - \|x^+ - x^\star\|^2\right),$$

where $x^\star$ is the minimizer of g, assuming $g(x^\star)$ is finite.

(iv, 2pt) Now, aggregating the last inequality over all steps $i = 0, \ldots, k$, show that the accuracy of gradient descent at iteration k is $O(1/k)$, i.e.,

$$g(x^{(k)}) - g(x^\star) \leq \frac{L}{2k}\|x^{(0)} - x^\star\|^2.$$

Put differently, for an ϵ-level accuracy, you need to run at most $O(1/\epsilon)$ iterations.

(b) Now consider the general h in assumption (A3). We will prove that the proximal gradient descent converges sublinearly under such assumptions. Specifically, the iterates of proximal gradient descent is computed by

$$x^+ = \text{prox}_{th}\left(x - t\nabla g(x)\right), \tag{6}$$

where again we will set $t = 1/L$ for simplicity. Further, we define the useful notation

$$G(x) = \frac{1}{t}\left(x - x^+\right).$$

We will see (in the following proofs) that $G(x)$ behaves like $\nabla g(x)$ in gradient descent.

(i, 3pt) Show that

$$g(x^+) - g(x) \leq -\frac{1}{L}\nabla g(x)^T G(x) + \frac{1}{2L}\|G(x)\|^2.$$

(ii, 3pt) Show that

$$f(x^+) - f(z) \leq G(x)^T(x - z) - \frac{1}{2L}\|G(x)\|^2, \quad \forall z \in \mathbb{R}^n.$$

Note that setting $z := x$ verifies the proximal gradient descent is a "descent method." (Hint: Look back at what you did in Q2 part (b) and add the missing h to (i).)

(iii, 4pt) Show that

$$f(x^+) - f(x^\star) \leq \frac{L}{2}\left(\|x - x^\star\|^2 - \|x^+ - x^\star\|^2\right),$$

where $x^\star$ is the minimizer of f. Then show that

$$f(x^{(k)}) - f(x^\star) \leq \frac{L}{2k}\|x^{(0)} - x^\star\|^2.$$

That is, the proximal descent method achieves $O(1/k)$ accuracy at the k-th iteration.

Bonus. If we further assume g being strongly convex with constant m, show that the proximal gradient descent converges linearly, that is,

$$f(x^+) - f(x^\star) \leq \left(1 - \frac{m}{L}\right)(f(x) - f(x^\star)).$$

You can use the following lemma.

Lemma 1 (Proximal Polyak-Łojasiewicz Inequality)**.** *Let $\lambda > 0$ be a scalar. Define*

$$\phi(x;\lambda) = -2\lambda \min_y \left(\nabla g(x)^T(y - x) + \frac{\lambda}{2}\|y - x\|^2 + h(y) - h(x)\right),$$

then

$$\phi(x;\lambda_1) \leq \phi(x;\lambda_2) \quad \textit{if} \quad \lambda_1 \leq \lambda_2.$$

Note that $\phi(x;\lambda)$ is related the minimum objective value in the proximal operators.
Hint: Bound $f(x) - f(x^\star)$ and $f(x) - f(x^+)$ using ϕ.

4 Stochastic & Proximal Gradient Descent (30 points) [Po-Wei, Wenbo, Akash]

Suppose predictors (columns of the design matrix $X \in \mathbb{R}^{n\times(p+1)}$) in a regression problem split up into J groups:

$$X = \begin{bmatrix}\mathbf{1} & X_{(1)} & X_{(2)} & \dots & X_{(J)}\end{bmatrix} \tag{7}$$

where $\mathbf{1} = (1\ 1\ \cdots\ 1) \in \mathbb{R}^n$. To achieve sparsity over non-overlapping groups rather than individual predictors, we may write $\beta = (\beta_0, \beta_{(1)}, \dots, \beta_{(J)})$, where β_0 is an intercept term and each $\beta_{(j)}$ is an appropriate coefficient block of β corresponding to $X_{(j)}$, and solve the regularized regression problem:

$$\min_{\beta\in\mathbb{R}^{p+1}} \; g(\beta) + h(\beta). \tag{8}$$

In the following problems, we will use linear regression to predict the Parkinsons disease (PD) symptom score on the `Parkinsons` dataset. The PD symptom score is measured on the unified Parkinsons disease rating scale (UPDRS). This data contains $5,785$ observations, 18 predictors (in `X_train.csv`), and an outcome—the toal UPDRS (in `y_train.csv`). The data were collected at the University of Oxford, in collaboration with 10 medical centers in the US and Intel Corporation. The 18 columns in the predictor matrix have the following groupings (in column ordering):

- `age`: Subject age in years
- `sex`: Subject gender, '0'–male, '1'–female
- `Jitter(%), Jitter(Abs), Jitter:RAP, Jitter:PPQ5, Jitter:DDP`: Several measures of variation in fundamental frequency of voice

- `Shimmer, Shimmer(dB), Shimmer:APQ3, Shimmer:APQ5, Shimmer:APQ11, Shimmer:DDA`: Several measures of variation in amplitude of voice
- `NHR, HNR`: Two measures of ratio of noise to tonal components in the voice
- `RPDE`: A nonlinear dynamical complexity measure
- `DFA`: Signal fractal scaling exponent
- `PPE`: A nonlinear measure of fundamental frequency variation

1) We first consider the ridge regression problem, where $h(\beta) = \frac{\lambda}{2}\|\beta\|_2^2$:

$$\min_{\beta \in \mathbb{R}^{p+1}} \frac{1}{2N}\|X\beta - y\|^2 + \frac{\lambda}{2}\|\beta\|_2^2 \tag{9}$$

where N is the number of samples. Note: in your implementation for this problem, if you added a ones vector to X ($X = \begin{bmatrix}\mathbf{1} & X_{(1)} & X_{(2)} & \dots & X_{(J)}\end{bmatrix}$), you should not include the bias term β_0 associated with the ones vector in the penalty.

(a) (2 pt) Derive the stochastic gradient update w.r.t. a batch-size B and a step-size t. Hint: you will need to a separate update for β_0 since it should not be penalized.

(b) (5 pt) Implement the stochastic gradient descent algorithm to solve the ridge regression problem (9). Initialize β with random normal values. Fit the model parameters on the training data (`X_train.csv`, `Y_train.csv`) and evaluate the objective function after each epoch (you will need to plot these values later). Set $\lambda = 1$. Try different batch-sizes from $\{10, 20, 50, 100\}$ and different step-sizes from $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$. Train for 500 epochs (an epoch is one iteration though the dataset).

(c) (3 pt) Plot $f^k - f^\star$ versus k ($k = 1, \dots, 500$) on a semi-log scale (i.e. where the y-axis is in log scale) for all setting combinations, where f^k denotes the objective value averaged over all samples at epoch k, and the optimal objective value is $f^\star = 57.0410$. What do you find? How do the different step sizes and batch sizes affect the learning curves (i.e. convergence rate, final convergence value, etc.)?

2) Next, we consider the least squares group LASSO problem, where $h(\beta) = \lambda \sum_j w_j \|\beta_{(j)}\|_2$:

$$\min_{\beta \in \mathbb{R}^{p+1}} \frac{1}{2N}\|X\beta - y\|^2 + \lambda \sum_j w_j \|\beta_{(j)}\|_2 \tag{10}$$

A common choice for weights on groups w_j is $\sqrt{p_j}$, where p_j is number of predictors that belong to the jth group, to adjust for the group sizes.

We will solve the problem using proximal gradient descent algorithm (over the whole dataset).

(a) (5 pt) Derive the proximal operator $\text{prox}_{h,t}(x)$ for the non-smooth component $h(\beta) = \lambda \sum_{j=1}^{J} w_j \|\beta_{(j)}\|_2$.

(b) (2 pt) Derive the proximal gradient update for the objective.

(c) (5 pt) Implement proximal gradient descent to solve the least squares group lasso problem on the `Parkinsons` dataset. Set $\lambda = 0.02$. Use a fixed step-size $t = 0.005$ and run for 10000 steps.

(d) (5 pt) Plot $f^k - f^\star$ versus k for the first 10000 iterations ($k = 1, \dots, 10000$) on a semi-log scale (i.e. where the y-axis is in log scale) for the training ~~and testing~~ data, where f^k denotes the objective value averaged over all samples at step k, and the optimal objective value is $f^\star = 49.9649$. Print the components of the solutions numerically. What are the selected groups?

(e) (3 pt) Now implement the LASSO (hint: you shouldn't have to do any additional coding), with fixed step-size $t = 0.005$ and $\lambda = 0.02$. ~~Run accelerated proximal gradient descent for 10000 steps.~~ Compare the LASSO solution with your group lasso solutions.

(f) (**Bonus**) Implement accelerated proximal gradient descent with fixed step-size under the same setting in part (c). Hint: be sure to exclude the bias term β_0 from the proximal update, just use a regular accelerated gradient update. Plot $f^k - f^\star$ versus k for both methods (unaccelerated and accelerated proximal gradient) for $k = 1, \ldots, 10000$ on a semi-log scale and compare the selected groups. What do you find?