Homework 3

Conjugate Gradient Descent, Accelerated Gradient Descent Newton, Quasi Newton and Projected Gradient Descent

CMU 10-725/36-725: Convex Optimization (Fall 2017)

OUT: Sep 29
DUE: **Oct 13, 5:00 PM**

START HERE: Instructions

- **Collaboration policy:** Collaboration on solving the homework is allowed, after you have thought about the problems on your own. It is also OK to get clarification (but not solutions) from books or online resources, again after you have thought about the problems on your own. There are two requirements: first, cite your collaborators fully and completely (e.g., "Jane explained to me what is asked in Question 3.4"). Second, write your solution independently: close the book and all of your notes, and send collaborators out of the room, so that the solution comes from you only.
- **Submitting your work:** Assignments should be submitted as PDFs using Gradescope unless explicitly stated otherwise. Each derivation/proof should be completed on a separate page. Submissions can be handwritten, but should be labeled and clearly legible. Else, submissions can be written in LaTeX. Upon submission, label each question using the template provided by Gradescope. Please refer to Piazza for detailed instruction for joining Gradescope and submitting your homework.
- **Programming**: All programming portions of the assignments should be submitted to Gradescope as well. We will not be using this for autograding, meaning you may use any language which you like to submit.

1 Convergence of Accelerated Gradient Descent (25 points) [Yichong]

In this problem we prove the convergence rate of Nesterov's accelerated gradient descent. Suppose $f : \mathbb{R}^d \to \mathbb{R}$ is a convex function and ∇f is L-Lipschitz. Starting with a starting point x_0, the accelerated gradient descent is defined as

$$\begin{aligned} p_k &= -\nabla f\left(x_k + \beta_k(x_k - x_{k-1})\right) + \beta_k p_{k-1}, \\ x_{k+1} &= x_k + \alpha_k p_k. \end{aligned}$$

We will work with the following setting of parameters: Let $\theta_0 = 0$, $\mathrm{d}\theta_k = \frac{1+\sqrt{1+4\theta_{k-1}^2}}{2}$, and $\mathrm{d}\beta_k = \frac{\theta_{k-1}-1}{\theta_k}$. Also $\alpha_k = 1/L$ for all k.

(a) [**4 pts**] Let $t_k = x_k + \beta_k(x_k - x_{k-1})$. Express $\nabla f(t_k)$ using t_k and x_{k+1}.

(b) [**6 pts**] Show that for any $y \in \mathbb{R}^d$ we have

$$f(x_{k+1}) - f(y) \le -\frac{1}{2L}\|\nabla f(t_k)\|_2^2 + \nabla f(t_k)^T(t_k - y).$$

(c) [**2 pts**] Express t_{k+1} using x_{k+1} and x_k (and possibly α and β).

(d) [**8 pts**] Apply (b) to $y = x_k$ and $y = x^*$, where x^* is the global minimum, and show that

$$\begin{aligned} &\theta_k^2\left(f(x_{k+1}) - f(x^*)\right) - \theta_{k-1}^2\left(f(x_k) - f(x^*)\right) \\ \le &\frac{L}{2}(\|\theta_k t_k - (\theta_k - 1)x_k - x^*\|_2^2 - \|\theta_{k+1}t_{k+1} - (\theta_{k+1} - 1)x_{k+1} - x^*\|_2^2). \end{aligned}$$

(e) [**5 pts**] Show that for every $t > 1$ we have

$$f(x_t) - f(x^*) \le \frac{2L\|t_1 - x^*\|_2^2}{t^2}.$$

i.e., a quadratic convergence.

2 Newton, Quasi Newton (20 points) [Yifeng]

2.1 Invariance under affine transformation of Newton update

[5 pts]

Let $f : \mathbb{R}^n \to \mathbb{R}$ be convex and twice differentiable, $b \in \mathbb{R}^n$ and $A \in \mathbb{R}^{n\times n}$ be invertible. Define g as $g(x) = f(Ax + b)$ for all x and let $u_0 \in \mathbb{R}^n$ be arbitrary but fixed. A step of Newton's method applied to f at u_0 results in

$$u_1 = u_0 - \left(\nabla^2 f(u_0)\right)^{-1}\nabla f(u_0). \tag{1}$$

Show that a step of the Newton's method applied to g at $x_0 = A^{-1}(u_0 - b)$ results in $x_1 = A^{-1}(u_1 - b)$.

This will imply that $g(x_1) = f(u_1)$, that is, the criterion values match after a Newton step. This will continue to be true at all iterations, and thus we say that Newton's method is affine invariant.

2.2 Quadratic convergence of Newton update

Let $f : \mathbb{R} \to \mathbb{R}$ be convex and three times continuously differentiable with $|f'''(x)| \leq C_1$ bounded and $f''(x) \geq C_2 > 0$ for all x. Let x^* be a local minimum. $x_1, x_2, \ldots$ are the points that Newton method iterates through.

(a) **[2 pts]** Represent $f'(x)$ using x, x^k, $f'(x_k)$, $f''(x_k)$, $f'''(\xi)$. Here $x \in \mathbb{R}$, and ξ is a value between x and x^k. Hint: Try Taylor series with Lagrange remainder.

(b) **[3 pts]** Substitute x with x^* in the above representation, show that

$$|x_{k+1} - x^*| = O(|x_k - x^*|^2). \tag{2}$$

2.3 Derivation of Davidon-Fletcher-Powell (DFP) update

One way to derive the DFP updates is to find B^+ closest to B in some norm so that B^+ satisfies the secant equation and is symmetric.

(a) **[5 pts] Simple Frobenius norm**

Assume $y, s \in \mathbb{R}^n$ are non-zero vectors and $B \in \mathbb{R}^{n \times n}$ is a symmetric matrix. Then the solution (you don't have to prove it) to the Frobenius norm minimization problem

$$\begin{aligned} \min_{B^+} \quad & \|B^+ - B\|_F^2 \\ \text{subject to} \quad & (B^+)^T = B^+ \\ & B^+ s = y \end{aligned}$$

is

$$B^+ = B + \frac{(y - Bs)s^T}{s^T s} + \frac{s(y - Bs)^T}{s^T s} - \frac{(y - Bs)^T s}{(s^T s)^2} ss^T. \tag{3}$$

Show via a counterexample that B^+ may not be positive definite even if B is symmetric positive definite and $y^T s > 0$.

Hint: observe that B^+ can be written as

$$B^+ = \left(I - \frac{ss^\top}{s^\top s}\right) B \left(I - \frac{ss^\top}{s^\top s}\right) + \frac{ys^\top}{s^\top s} + \frac{sy^\top}{s^\top s} - \frac{y^\top s}{(s^\top s)^2} ss^\top.$$

(b) **[5 pts] Weighted Frobenius norm**

Assume $y, s \in \mathbb{R}^n$ are such that $y^T s > 0$ and $B \in \mathbb{R}^{n \times n}$ is a symmetric matrix. Let $W \in \mathbb{R}^{n \times n}$ be a non-singular matrix such that $WW^T s = y$. Show that the solution to the weighted Frobenius norm minimization problem

$$\begin{aligned} \min_{B^+} \quad & \|W^{-1}(B^+ - B)W^{-T}\|_F^2 \\ \text{subject to} \quad & (B^+)^T = B^+ \\ & B^+ s = y \end{aligned}$$

is the DFP update

$$B^+ = B + \frac{(y - Bs)y^T}{y^T s} + \frac{y(y - Bs)^T}{y^T s} - \frac{(y - Bs)^T s}{(y^T s)^2} yy^T.$$

Hint: use Eq. (3) and a suitable change of variables.

3 Conjugate Gradient Descent (20 points) [Hao]

[10pts] Linear Search methods Show that the Polak-Ribiere formula given by $\beta_k^{PR} = \frac{(g_{k+1}-g_k)^T g_{k+1}}{g_k^T g_k}$ can be reduced to the Fletcher-Reeves formula $\beta_k^{FR} = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k}$ (Course Slides P36), when applied to a quadratic function, with exact line searches. Please note that $g_k = \bigtriangledown f(x_k)$.

[10pts] Quadratic: Show that if $f(x)$ is a strictly convex quadratic, then the function $h(\sigma) = f(x_0 + \sigma_0 p_0 + ... + \sigma_{k-1} p_{k-1})$ is also a strictly convex quadratic in the variable $\sigma = (\sigma_0, \sigma_1, ..., \sigma_{k-1})^T$.

4 Group Lasso by Proximal Gradient Descent (25 points) [Hongyang]

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

$$X = \left[\mathbf{1}\ X_{(1)}\ X_{(2)}\ \dots\ X_{(J)}\right],$$

where $\mathbf{1} = (1\ 1\ \cdots\ 1) \in \mathbb{R}^n$ and $X_{(j)} \in \mathbb{R}^{n\times p_j}$ such that $p_1 + ... + p_J = p$. 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 group lasso problem:

$$\min_{\beta\in\mathbb{R}^{p+1}}\ g(\beta) + \lambda \sum_{j=1}^{J} w_j \|\beta_{(j)}\|_2, \tag{4}$$

where $g(\beta)$ is a goodness of fit to data term, and λ is the group Lasso regularization term encouraging sparsity over groups. 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.

(a) (5 pts) Derive the proximal operator $\text{prox}_{h,t}(x) := \text{argmin}_\beta\, h(\beta) + \frac{1}{2t}\|\beta - x\|_2^2$ for the nonsmooth component $h(\beta) = \lambda \sum_{j=1}^{J} w_j \|\beta_{(j)}\|_2$.

Hint: The proximal operator $\text{prox}_{f,t}(x)$ for $f(z) = \|z\|_2$ is given by

$$\text{prox}_{f,t}(x) = \begin{cases} \frac{\|x\|_2 - t}{\|x\|_2} x, & \|x\|_2 \geq t, \\ 0, & \|x\|_2 < t. \end{cases}$$

(b) (20 pts) In this problem, we will use logistic group lasso to classify a person's age group from his movie ratings. The movie ratings can be categorized into groups according to a movie's genre (e.g. all ratings for action movies can be grouped together). Our data does not contain ratings for movies from multiple genre (i.e. has no overlapping groups). We will use proximal gradient descent to solve the group lasso problem.

We formulate the problem as a binary classification with output label $y \in \{0, 1\}$, corresponding to whether a person's age is under 40, and input features $X \in \mathbb{R}^{n\times p}$. We model each $y_i|x_i$ with the probabilistic model

$$\log\left(\frac{p_\beta(y_i = 1|x_i)}{1 - p_\beta(y = 1|x_i)}\right) = (X\beta)_i,$$

$i = 1, \dots, n$. The logistic group lasso estimator is given by solving the minimization problem in (4) with

$$g(\beta) = -\sum_{i=1}^{n} y_i (X\beta)_i + \sum_{i=1}^{n} \log(1 + \exp\{(X\beta)_i\}),$$

the negative log-likelihood under the logistic probability model.

(i) Derive the gradient of g in this case.

(ii) Implement proximal gradient descent to solve the logistic group lasso problem. Fit the model parameters on the training data (`moviesTrain.mat` available on the class website). The features have already been arranged into groups and you can find information about the labels of each group in `moviesGroups.mat`. Use regularization parameter $\lambda = 5$ for 1000 iterations with fixed step size $t = 10^{-4}$.

Now, implement accelerated proximal gradient descent with fixed step size. Use the same λ, t, and number of iterations as before.

For each of the two methods, plot $f^{(k)} - f^\star$ versus k, where $f^{(k)}$ denotes the objective value at iteration k, and now the optimal objective value is $f^\star = 336.207$ on a semi-log scale (i.e. where the y-axis is in log scale).

(iii) Finally, we will use the accelerated proximal gradient descent from part (*ii*) to make predictions on the test set, available in `moviesTest.mat`. What is the classification error?