Homework 4

ORF522: Linear and Nonlinear Optimization

Instructor: Bartolomeo Stellato
AI: Irina Wang

Due on December 1

Problem 1 - Convergence rate for proximal gradient descent

In this problem, you will show that the convergence rate for proximal gradient descent (also known as the proximal gradient method) is $O(1/k)$, where $k \geq 1$ is the number of iterations that the algorithm is run for. The derivation will not be based on notions of operator theory.

As a reminder, the setup for proximal gradient descent is as follows. We assume that the objective $f(x)$ can be written as

$$f(x) = g(x) + h(x).$$

We compute the iterates

$$x^i = \mathbf{prox}_{t_i h}(x^{i-1} - t_i \nabla g(x^{i-1})),$$

where $i \geq 1$ is an iteration counter, x^0 is the initial point, and the $t_i > 0$ are step sizes (chosen appropriately, during iteration i). These are the assumptions for this problem:

- (A1) g is convex, differentiable, and $\mathbf{dom}\, g = \mathbf{R}^n$.
- (A2) ∇g is Lipschitz, with constant $L > 0$.
- (A3) h is convex, not necessarily differentiable, and we take $\mathbf{dom}\, h = \mathbf{R}^n$ for simplicity.
- (A4) If the step sizes t_i are either taken to be constant, i.e., $t_i = t = 1/L$, or chosen by backtracking line search; either way, the following inequality holds:

  $$g(x^i) \leq g(x^{i-1}) - t \nabla g(x^{i-1})^T G_t(x^{i-1}) + (t/2)\|G_t(x^{i-1})\|_2^2,$$

  where t is the step size at any iteration of the algorithm, and we define

  $$G_t(x^{i-1}) = (1/t)(x^{i-1} - x^i).$$

This inequality follows from assumption (A2), but you can just take it to be true for this problem. G_t is often referred to as the "gradient mapping" operator.

Now, let's assume, for all parts of this exercise, that the step size is fixed, i.e., $t = 1/L$.

1. Show that
$$s = G_t(x^{i-1}) - \nabla g(x^{i-1})$$
is a subgradient of h evaluated at x^i. (As a reminder, h is the potentially nondifferentiable function in our decomposition of the objective f.)

2. Derive the following inequality:
$$f(x^i) \leq f(z) + G_t(x^{i-1})^T(x^{i-1} - z) - (t/2)\|G_t(x^{i-1})\|_2^2, \quad z \in \mathbf{R}^n.$$

3. Show that the sequence of objective function evaluations $\{f(x^i)\}$, $i = 0, \ldots, k$, is nonincreasing (don't worry about the case when x^i is a minimizer of f). This result says that proximal gradient descent is a "descent method".

4. Derive the following inequality:
$$f(x^i) - f(x^\star) \leq \frac{1}{2t}(\|x^{i-1} - x^\star\|_2^2 - \|x^i - x^\star\|_2^2),$$
where $x^\star$ is a minimizer of f (we assume $f(x^\star)$ is finite). This result, taken together with what you showed in part (3), implies that we move closer to the optimal point(s) on each iteration of proximal gradient descent.

5. Show that after k iterations, the accuracy that proximal gradient descent (with a fixed step size of $1/L$) obtains is $O(1/k)$, i.e.,
$$f(x^k) - f(x^\star) \leq \frac{1}{2kt}\|x^0 - x^\star\|_2^2,$$
meaning that the convergence rate for proximal gradient descent is $O(1/k)$. In other words, if you desire ϵ-level accuracy, roughly speaking, then you must run proximal gradient descent for $O(1/\epsilon)$ iterations.)

Problem 2 - Properties of proximal operators

We will inspect various properties and examples of proximal operators. Unless otherwise specified, take h to be a convex function with domain $\mathbf{R}^n$, and $t > 0$ be arbitrary, and consider its associated proximal operator
$$\mathbf{prox}_{th}(x) = \operatorname*{argmin}_z \left(th(z) + \frac{1}{2}\|z - x\|_2^2 \right).$$

1. Prove that $\mathbf{prox}_{th}(x) = u$ if and only if

$$h(y) \geq h(u) + \frac{1}{t}(x-u)^T(y-u), \quad \forall y.$$

Hint: use subgradient optimality.

2. Prove that $\mathbf{prox}_{th}$ is nonexpansive, meaning

$$\|\mathbf{prox}_{th}(x) - \mathbf{prox}_{th}(y)\|_2 \leq \|x-y\|_2, \quad \forall x, y.$$

Hint: use the previous question, and the monotonicity of subgradients.

3. The proximal minimization algorithm (a special case of proximal gradient descent) repeats the updates:

$$x^{k+1} = \mathbf{prox}_{th}(x^k), \quad k = 1, 2, 3, \ldots.$$

Write out these updates when applied to $h(x) = (1/2)x^T Ax - b^T x$, where $A \in \mathbf{S}^n_+$. Show that this is equivalent to the *iterative refinement* algorithm for solving the linear system $Ax = b$, i.e.,

$$x^{k+1} = x^k + (A + \epsilon I)^{-1}(b - Ax^k), \quad k = 1, 2, 3, \ldots,$$

where $\epsilon > 0$ is some constant.

4. (Optional, bonus points) Assuming the proximal minimization converges to the minimizer of $h(x) = \frac{1}{2}x^T Ax - b^T x$ (which it does, under suitable step sizes), what would the iterations of *iterative refinement* converge to in the case when A is singular, $Ax = b$, and $x^0 = 0$?

Problem 3 - Proximal gradient descent for group Lasso

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

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

where $\mathbf{1} \in \mathbf{R}^n$ is a all-one vector. To achieve sparsity over non-overlapping groups rather than individual predictions, we may write $\beta = (\beta_0, \beta_{(1)}, \ldots, \beta_{(J)})$, 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:

$$\underset{\beta \in \mathbf{R}^{p+1}}{\text{minimize}} \quad g(\beta) + \lambda \sum_{j=1}^{J} w_j \|\beta_{(j)}\|_2$$

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

- Derive the proximal operator $\mathbf{prox}_{th}(x)$ for the nonsmooth component

$$h(\beta) = \lambda \sum_{j=1}^{J} w_j \|\beta_{(j)}\|_2$$

Note: to read `csv` files, you can use the function `read_csv` from the Python `pandas` package.

Birthweight group Lasso

- Download the birthweight data set `birthweight.zip` from Canvas. This data contains 189 observations, 16 predictors (in `X.csv`), and an outcome, birthweight (in `y.csv`). The data were collected at Baystate Medical Center, Springfield, Mass during 1986. The 16 columns in the predictor matrix have groupings according to the following categories:
  - `age1,age2,age3`: Orthogonal polynomials of first, second, and third degree representing mothers age in years
  - `lwt1,lwt2,lwt3`: Orthogonal polynomials of first, second, and third degree representing mothers weight in pounds at last menstrual period
  - `whit,black`: Indicator functions for mothers race; "other" is reference group.
  - `smoke`: Smoking status during pregnancy
  - `ptl1,ptl2m`: Indicator functions for one or for two or more previous premature labors, respectively. No previous premature labors is the reference category.
  - `ht`: History of hypertension
  - `ui`: Presence of uterine irritability
  - `ftv1,ftv2,ftv3m`: Indicator functions for one, for two, or for three or more physician visits during the first trimester, respectively. No visits is the reference category.

  Remark: A reference category/group is the *baseline* level in categorical data when it is coded as dummy variables. For instance, if a categorical variable has 3 levels (say, three drugs administered to patients), then a baseline category variable may be the first drug, and two dummy variables may be created each with ith entry is coded as 1 if the ith person was treated with that drug. Why do this? This allows for the model's fitted coefficient for the dummy variables measure the average difference between the response level between the second or third category and the first category (adjusting for the effect of all other variables).

- Let $g(\beta) = \|y - X\beta\|_2^2$, in which case the problem above is called the least squares group lasso problem. Derive the gradient of g in this case.

- Use two methods to solve the least squares group lasso problem with the birthweight date set:

1. proximal gradient descent with $\lambda = 4$, and a fixed step size $t = 0.002$ and 1000 steps;
2. subgradient descent method with $\lambda = 4$, and a diminishing step size $t = 0.003/\sqrt{k+1}$ and 1000 steps. For both methods, plot $f^k - f^\star$ versus k (i.e., $f^k - f^\star$ is on the y-axis in log scale, and k on the x-axis), where $f^{(k)}$ denotes the objective value at iteration k, and the optimal objective value is $f^\star = 84.6952$.

- Print the components of the solutions numerically from the two methods in the previous point to see that they are close. What are the selected groups?
- Now implement the lasso (hint: you shouldn't have to do any additional coding), with fixed step size with $\lambda = 0.35$, and compare the lasso solution with your group lasso solutions.

Movie ratings group Lasso

In this problem, we'll 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). Similar to the previous part, we'll 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 $X \in \mathbf{R}^{n \times p}$. We model each $y_i \mid x_i$ with the probabilistic model

$$\log\left(\frac{p_\beta(y_i = 1 \mid x_i)}{1 - p_\beta(y = 1 \mid x_i)}\right) = (X\beta)_i, \quad i = 1, \ldots, n.$$

The logistic group lasso estimator is given by solving the minimization problem in the group lasso problem 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.

- Derive the gradient of g in this case.
- Fit the model parameters on the data available from `movies.zip` on Canvas. The files `X_train.csv` and `y_train.csv` represent the training data. The features have already been arranged into groups and you can find information about the labels of each group in `group_titles.csv` and `group_labels_per_rating.csv`. Solve the logistic group lasso problem using two methods:

1. Proximal gradient descent with regularization parameter $\lambda = 5$ for 1000 iterations with fixed step size $t = 10^{-4}$.
2. Proximal gradient descent with backtracking line search, where the step-size shrinking parameter β is set to 0.1. Use the same λ as before with only 400 outer iterations.

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). For backtracking line search, count the inner iterations towards the iteration number, in order to make a fair comparison.

Hint: The conditions of backtracking line search for proximal gradient descent is given in Problem 1, Assumption (A4).

3. Finally, we will use the proximal gradient descent from point 2 to make predictions on the test set, available in the files `X_test.csv` and `y_test.csv`. What is the classification error? What move genre are important for classifying whether a viewer is under 40 years old?