EECS 127/227AT Optimization Models in Engineering UC Berkeley Spring 2023

Homework 12

This homework is due at 11 PM on April 20, 2023.

Submission Format: Your homework submission should consist of a single PDF file that contains all of your answers (any handwritten answers should be scanned).

1. **Newton's Method, Coordinate Descent and Gradient Descent**

In this question, we will compare three different optimization methods: Newton's method, coordinate descent and gradient descent. We will consider the simple set-up of unconstrained convex quadratic optimization; i.e we will consider the following problem:

$$\min_{\vec{x}\in\mathbb{R}^d} \vec{x}^\top A\vec{x} - 2\vec{b}^\top \vec{x} + c \tag{1}$$

where $A \succ 0$ and $\vec{b} \in \mathbb{R}^d$.

(a) How many steps does Newton's method take to converge to the optimal solution? Recall that the update rule for Newton's method is given by the equation:

$$\vec{x}_{t+1} = \vec{x}_t - (\nabla^2 f(\vec{x}_t))^{-1}\nabla f(\vec{x}_t). \tag{2}$$

when optimizing a function f.

(b) Now, consider the simple two variable quadratic optimization problem for $\sigma > 0$:

$$\min_{\vec{x}\in\mathbb{R}^2} f(\vec{x}) = \sigma x_1^2 + x_2^2. \tag{3}$$

How many steps does coordinate descent take to converge on this problem? Assume that we start by updating the variable x_1 in the first step, x_2 in step two and so on; therefore, we will update x_1 and x_2 in odd and even iterations respectively:

$$(x_{t+1})_1 = \begin{cases} \operatorname{argmin}_{x_1} f(x_1,(x_t)_2) & \text{for odd t} \\ (x_t)_1 & \text{otherwise} \end{cases} \text{ and } (x_{t+1})_2 = \begin{cases} \operatorname{argmin}_{x_2} f((x_t)_1,x_2) & \text{for even t} \\ (x_t)_2. & \text{otherwise} \end{cases} \tag{4}$$

Here, $(x_t)_2$ represents x_2 at time t and so on.

(c) We will now analyze the performance of coordinate descent on another quadratic optimization problem:

$$\min_{\vec{x}\in\mathbb{R}^2} f(\vec{x}) = \sigma(x_1 + x_2)^2 + (x_1 - x_2)^2. \tag{5}$$

where we have, as before, $\sigma > 0$. Note that $(0, 0)$ is the optimal solution to this problem. Now, starting from the point $\vec{x}_0 = (1, 1)$, write how each coordinate of $(\vec{x}_{t+1})_i$ relates to $(\vec{x}_t)_i$ for $i = 1, 2$. Use this to show how the algorithm converges from the initial point $(1, 1)$ to $(0, 0)$. What happens when σ grows large? *HINT: First find the update rule for $(\vec{x}_t)_1$, i.e. keep $(\vec{x}_t)_2$ fixed and figure out how $(\vec{x}_t)_1$ changes when t is odd. Then do the same for $(\vec{x}_t)_2$ when $(\vec{x}_t)_1$ is fixed for even t.*

(d) Now, let $f(\vec{x}) = \frac{1}{2}\vec{x}^\top A\vec{x} + \vec{x}^\top \vec{b} + c$ where A is PD. When we run gradient descent on $f(\vec{x})$, the convergence along each of the unit eigenvectors $\vec{v}_i$ of A is

$$|1 - \eta\,(\lambda_i\{A\})| \tag{6}$$

This can be derived similar to HW 8 Question 1e, which you may reference. Formally, in the current setting, we have

$$(\vec{x}_k - \vec{x}^\star) = (I - \eta A)^k (\vec{x}_0 - \vec{x}^\star)$$

One way we can derive an "optimal" learning rate $\eta^\star$ is to minimize the largest rate of convergence:

$$\eta^\star \in \operatorname*{argmin}_{\eta \in \mathbb{R}} \max_{i \in \{1,\ldots,n\}} \left|1 - \eta\left(\lambda_i\{A\}\right)\right|. \tag{7}$$

One important property of $\eta^\star$ is that it makes the rates of convergence $|1 - \eta\left(\lambda_i\{A\}\right)|$ associated with the largest and smallest singular values of A equal, i.e.,

$$|1 - \eta(\lambda_{\max}\{A\})| = |1 - \eta(\lambda_{\min}\{A\})|$$

Use this property to show that

$$\eta^\star = \frac{2}{\lambda_{\max}\{A\} + \lambda_{\min}\{A\}} \tag{8}$$

where $\lambda_{\min}\{A\} = \lambda_n\{A\}$ is the n^{th} largest singular value of A and similar for the maximum.

(e) Finally, for the objective function (5), write an equation relating $\vec{x}_t$ to $\vec{x}_0$ if $\vec{x}_0 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$. Assume for this part that $\sigma > 1$ and reason about how quickly gradient descent converges when σ grows large. *HINT: What is the optimal step size for gradient descent, using the previous part? HINT: Also note that f is given by:*

$$f(\vec{x}) = \vec{x}^\top A \vec{x} \textit{ where } A = 2\left(\sigma \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} + \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}\right). \tag{9}$$

2. Gradient Descent vs Newton Method

Run the Jupyter notebook `gradient_vs_newton.ipynb` which demonstrates differences between gradient descent and Newton's method.

3. LASSO vs. Ridge

Say that we have the data set $\{(\vec{x}^{(i)}, y^{(i)})\}_{i=1,\ldots,n}$ of samples $\vec{x}^{(i)} \in \mathbb{R}^d$ and values $y^{(i)} \in \mathbb{R}$.

Define $X = \begin{bmatrix} \vec{x}^{(1)} & \ldots & \vec{x}^{(n)} \end{bmatrix}^\top$ and $y = \begin{bmatrix} y^{(1)} & \ldots & y^{(n)} \end{bmatrix}^\top$.

For the sake of simplicity, assume that the data has been centered and whitened so that each feature has mean 0 and variance 1 and the features are uncorrelated, i.e. $X^\top X = nI$. Consider the linear least squares regression with regularization in the ℓ_1-norm, also known as LASSO:

$$\vec{w}^\star = \underset{\vec{w} \in \mathbb{R}^d}{\operatorname{argmin}} \; \|X\vec{w} - \vec{y}\|_2^2 + \lambda \|\vec{w}\|_1 . \tag{10}$$

This problem will compare ℓ_1-regularization with ℓ_2-regularization (ridge regression) to understand their similarities and differences. We will do this by looking at the elements of $\vec{w}^\star$ in the solution to each problem.

(a) First, we decompose this optimization problem into d univariate optimization problems over each element of $\vec{w}$. Let $X = \begin{bmatrix} \vec{x}_1 & \ldots & \vec{x}_d \end{bmatrix}$ and recall that $X^\top X = nI$.

(b) If $w_i^\star > 0$, then what is the value of $w_i^\star$? What is the condition on $\vec{y}^\top \vec{x}_i$ for this to be possible?

(c) If $w_i^\star < 0$, then what is the value of $w_i^\star$? What is the condition on $\vec{y}^\top \vec{x}_i$ for this to be possible?

(d) What can we conclude about $w_i^\star$ if $\left|\vec{y}^\top \vec{x}_i\right| \leq \frac{\lambda}{2}$? How does the value of λ impact the individual entries $w_i^\star$?

(e) Now consider the case of ridge regression, which uses the the ℓ_2 regularization $\lambda \|\vec{w}\|_2^2$.

$$\vec{w}^\star = \underset{\vec{w} \in \mathbb{R}^d}{\operatorname{argmin}} \; \|X\vec{w} - \vec{y}\|_2^2 + \lambda \|\vec{w}\|_2^2 . \tag{11}$$

Write down the new condition for $\vec{w}_i^\star$ to be 0. How does this differ from the condition obtained in part (4) and what does this suggest about LASSO?

4. More Fun with Lasso and Ridge

Complete the Jupyter notebook `ridge_vs_lasso.ipynb` which demonstrates differences between ridge regression and LASSO.

5. Connecting Ridge Regression, LASSO, and Constrained Least Squares

This question aims to help you develop an understanding of how a constraint in an optimization problem has the same effect as a penalty term in the objective.

(a) Let $f\colon \mathbb{R}^n \to \mathbb{R}$ be strictly convex and such that $\lim_{\alpha\to\infty} f(\alpha\vec{v}) = \infty$ for any nonzero $\vec{v} \in \mathbb{R}^n$. Let $g\colon \mathbb{R}^n \to \mathbb{R}_+$ be convex and take non-negative values. Further, suppose that there exists $\vec{x}_0 \in \mathbb{R}^n$ such that $g(\vec{x}_0) = 0$.

For $\lambda \geq 0$ and $k \geq 0$, define the "penalty" and "constraint" programs

$$P(\lambda) \doteq \operatorname*{argmin}_{\vec{x}} \{f(\vec{x}) + \lambda g(\vec{x})\} \tag{12}$$

$$C(k) \doteq \operatorname*{argmin}_{\vec{x}\colon\, g(\vec{x})\leq k} f(\vec{x}). \tag{13}$$

Show that:

- for every $\lambda \geq 0$ there exists $k \geq 0$ such that $P(\lambda) = C(k)$, and
- for every $k > 0$ there exists $\lambda \geq 0$ such that $P(\lambda) = C(k)$.

HINT: First show using strict convexity that, for $k \geq 0$ and $\lambda \geq 0$, both $P(\lambda)$ and $C(k)$ have exactly one element, i.e., each problem has exactly one optimal solution. You may use without proof that $P(\lambda)$ and $C(k)$ have at least one element each (this is true from assumptions but requires some analysis to show).

To show the first direction (i.e. for all λ there exists k...), let $\vec{x}^\star \in P(\lambda)$ and show that $\vec{x}^\star \in C(k)$ for $k = g(\vec{x}^\star)$. You might need the fact that $P(\lambda)$ and $C(k)$ have exactly one element. To show the other direction (i.e. for all k there exists λ...), prove that strong duality holds for the constraint problem, let $\vec{x}^\star \in C(k)$ and $\mu^\star$ be optimal primal and dual variables for the constraint problem and show that $\vec{x}^\star \in P(\lambda)$ for $\lambda = \mu^\star$.

Let $A \in \mathbb{R}^{m\times n}$ have full column rank, and let $\vec{y} \in \mathbb{R}^m$. In the course, we have looked at LASSO:

$$\text{LASSO}(\lambda) \doteq \operatorname*{argmin}_{\vec{x}} \left\{ \|A\vec{x} - \vec{y}\|_2^2 + \lambda \|\vec{x}\|_1 \right\} \tag{14}$$

and ridge regression:

$$\text{Ridge}(\lambda) \doteq \operatorname*{argmin}_{\vec{w}} \left\{ \|A\vec{x} - \vec{y}\|_2^2 + \lambda \|\vec{x}\|_2^2 \right\} \tag{15}$$

which add an ℓ^1 and ℓ^2 norm penalty to the least squares objective, respectively. The analogous constrant programs are the ℓ^1- and ℓ^2-constrained least squares problems:

$$\ell^1\text{CLS}(k) \doteq \operatorname*{argmin}_{\vec{x}\colon\, \|\vec{x}\|_1\leq k} \|A\vec{x} - \vec{y}\|_2^2 \tag{16}$$

$$\ell^2\text{CLS}(k) \doteq \operatorname*{argmin}_{\vec{x}\colon\, \|\vec{x}\|_2^2\leq k} \|A\vec{x} - \vec{y}\|_2^2 . \tag{17}$$

(b) Show that the result from part (a) can be used to show the equivalence of LASSO with ℓ^1CLS and the equivalence of ridge regression with ℓ^2CLS. Namely, for each pair of equivalent formulations, find f and g, prove that f is strictly convex, prove that g is convex, and prove that there is an $\vec{x}_0$ such that $g(\vec{x}_0) = 0$.

(c) Complete the Jupyter notebook, which will use this equivalence to show geometrically why LASSO solutions tend to be sparse (i.e. have many zeros) while ridge regression doesn't, and attach a PDF printout of your answers.

6. Homework Process

With whom did you work on this homework? List the names and SIDs of your group members.

NOTE: If you didn't work with anyone, you can put "none" as your answer.

