ECE 3803, Fall 2021

Homework #5

Due Thursday, November 4, at 11:59pm

1. Prepare a one paragraph summary of what we talked about in class since the last assignment. I do not want just a bulleted list of topics, I want you to use complete sentences and establish context (Why is what we have learned relevant? How does it connect with other classes?). The more insight you give, the better.

2. In the notes we showed that the subdifferential of $\|\boldsymbol{x}\|_1$ at $\boldsymbol{x}$ is given by vectors $\boldsymbol{u}$ that satisfy

$$\begin{aligned} u_n &= \operatorname{sign}(x_n) && \text{if } x_n \neq 0, \\ |u_n| &\leq 1 && \text{if } x_n = 0. \end{aligned}$$

   Note that we could also write this as

$$\partial\|\boldsymbol{x}\|_1 = \left\{\boldsymbol{u} \,:\, \|\boldsymbol{u}\|_\infty = 1, \boldsymbol{u}^{\mathrm{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_1\right\}.$$

   It turns out that the subdifferential of $\|\boldsymbol{x}\|_\infty$ takes the related form:

$$\partial\|\boldsymbol{x}\|_\infty = \left\{\boldsymbol{u} \,:\, \|\boldsymbol{u}\|_1 = 1, \boldsymbol{u}^{\mathrm{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_\infty\right\}.$$

   (a) Describe a simple procedure for constructing a vector $\boldsymbol{u} \in \partial\|\boldsymbol{x}\|_\infty$ from $\boldsymbol{x}$.

   (b) Show that if $\boldsymbol{u}$ satisfies $\|\boldsymbol{u}\|_1 = 1$ and $\boldsymbol{u}^{\mathrm{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_\infty$, then it must be a subgradient.

   (c) (Optional.) Provide an argument that if $\boldsymbol{u}$ is a subgradient of $\|\boldsymbol{x}\|_\infty$, then it must satisfy $\|\boldsymbol{u}\|_1 = 1$ and $\boldsymbol{u}^{\mathrm{T}}\boldsymbol{x} = \|\boldsymbol{x}\|_\infty$.

3. We have now spent quite a while looking at least squares problems where we aim to minimize $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2$. While minimizing the ℓ_2 norm of the error is *often* a good idea, a big part of why it is so popular is just that it is easy to minimize. But now that we know about nonsmooth optimization, we can explore situations where other ℓ_p norms might be more appropriate. In particular, we will consider minimizing $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_1$ and $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_\infty$ using the subgradient method.

   (a) In the notes we described how to construct a subgradient for $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_1$. Using a similar approach (and guided by the previous problem) show how to construct a subgradient for $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_\infty$.

   (b) Download the file `hw06_prob3.py`. This file sets up a simple regression problem in which $\boldsymbol{b}$ consists of noisy observations of a smooth function $f(t)$ and considers three noise models: Gaussian noise, sparse Gaussian noise, and uniform noise. Compute and plot the least squares solution for each case.

   (c) Now implement subgradient descent for $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_1$. Use a backtracking line search to set the step size α_k. Note that there are two related implementation challenges to think about. First, you can pick *any* subgradient. This gives you some freedom whenever an entry of $\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}$ is equal to zero. Feel free to pick any rule you like here. The other consideration is that, unless you are lucky, your procedure for selecting a subgradient will not necessarily result in a subgradient of $\mathbf{0}$ at the solution. Thus, the norm of the subgradient is not a good test for convergence. Instead you can either check to see when the objective function stops improving or when the iterates $\boldsymbol{x}_k$ stop changing. (You should probably check this to make sure it is constant for several iterations in a row.) Compute and plot the resulting estimate of $\boldsymbol{x}$ for each of the three noise cases.

(d) Next repeat the implementation process from part (c) for $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_\infty$. Again, compute and plot the results for each of the three noise cases.

(e) Compare the results that you obtained in the previous parts. Which method seems best suited for each case?

4. In the notes (on page 86) we claimed (without proving) that the prox operator for the ℓ_1 norm

$$\text{prox}_{\alpha h}(\boldsymbol{z}) = \arg\min_{\boldsymbol{x} \in \mathbb{R}^N} \left(\tau \|\boldsymbol{x}\|_1 + \frac{1}{2\alpha} \|\boldsymbol{x} - \boldsymbol{z}\|_2^2 \right)$$

is given by

$$\text{prox}_{\alpha h}(\boldsymbol{z}) = T_{\tau\alpha}(\boldsymbol{z}),$$

where $T_{\tau\alpha}$ is the soft-thresholding operator, whose n^{th} entry is given by

$$[T_{\tau\alpha}(\boldsymbol{z})]_n = \begin{cases} z_n - \tau\alpha, & z_n \geq \tau\alpha, \\ 0, & |z_n| \leq \tau\alpha, \\ z_n + \tau\alpha, & z_n \leq -\tau\alpha. \end{cases}$$

Prove that $\boldsymbol{x}^\star = T_{\tau\alpha}(\boldsymbol{z})$ is indeed a minimizer of

$$f(\boldsymbol{x}) = \tau \|\boldsymbol{x}\|_1 + \frac{1}{2\alpha} \|\boldsymbol{x} - \boldsymbol{z}\|_2^2$$

by showing that $\mathbf{0} \in \partial f(\boldsymbol{x}^\star)$. [Note: $\partial f(\boldsymbol{x})$ in this problem is just a special case of the subdifferential calculated on page 71 of the notes.]

5. **The LASSO.** In this problem you will implement both subgradient descent and proximal gradient descent to solve the LASSO:

$$\underset{\boldsymbol{x} \in \mathbb{R}^N}{\text{minimize}} \ \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \tau \|\boldsymbol{x}\|_1.$$

You will evaluate your code by testing it on the problem defined by the following code:

```
import numpy as np
np.random.seed(2021) # Set random seed so results are repeatable
# Set parameters
M = 100
N = 1000
S = 10

# Define A and y
```

```
A = np.random.randn(M,N)
ind0 = np.random.choice(N,S,0) # index subset
x0 = np.zeros(N)
x0[ind0] = np.random.rand(S)
y = A@x0 + .25*np.random.randn(M)
```

In all of the problems below, set $\tau = 1.5$.

(a) Implement subgradient descent for this problem. Produce a plot showing the value of the objective function as a function of iteration number. Show results for the following step size selection rules: $\alpha_k = \alpha$, $\alpha_k = \alpha/\sqrt{k}$, $\alpha_k = \alpha/k$. For each rule tune α to get reasonable performance.

(b) Implement the proximal gradient method for this problem (without acceleration). Use a fixed step size α. You may tune this manually, but there is also a principled choice. Produce a plot showing the value of the objective function as a function of iteration number.

(c) Implement the proximal gradient method with acceleration. Use the same choice of α as in the previous part and use the rule $\beta_k = (k-1)/(k+2)$. Produce a plot showing the value of the objective function as a function of iteration number.