Homework 4

Due March 7 at 11 pm

1. (Proximal operator) The proximal operator of a function $f : \mathbb{R}^n \to \mathbb{R}$ is defined as

$$\text{prox}_f(y) := \arg\min_x f(x) + \frac{1}{2}||x - y||_2^2. \tag{1}$$

   (a) Derive the proximal operator of the squared ℓ_2 norm weighted by a constant $\alpha > 0$, i.e. $f(x) = \alpha ||x||_2^2$.

   (b) Prove that the proximal operator of the ℓ_1 norm weighted by a constant $\alpha > 0$ is a soft-thresholding operator,

$$\text{prox}_{\alpha||\cdot||_1}(y) = \mathcal{S}_\alpha(y), \tag{2}$$

   where

$$\mathcal{S}_\alpha(y)[i] := \begin{cases} y[i] - \text{sign}(y[i])\,\alpha & \text{if } |y[i]| \geq \alpha, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

   (c) Prove that if $X \in \mathbb{R}^{p \times n}$ has orthonormal rows ($p \leq n$) and $y \in \mathbb{R}^n$, then for any function f

$$\arg\min_\beta \frac{1}{2}\left||y - X^T\beta\right||_2^2 + f(\beta) = \arg\min_\beta \frac{1}{2}||Xy - \beta||_2^2 + f(\beta). \tag{4}$$

   (d) Use the answers to the previous questions to compare the ridge-regression and lasso estimators for a regression problem where the features are orthonormal.

2. (Proximal gradient method)

   (a) The first-order approximation to a function $f : \mathbb{R}^p \to \mathbb{R}$ at $x \in \mathbb{R}^p$ equals

$$f(x) + \nabla f(x)^T (y - x). \tag{5}$$

   We want to minimize this first-order approximation locally. To this end we fix a real constant $\alpha > 0$ and augment the approximation with an ℓ_2-norm term that keeps us close to x,

$$f_x(y) := f(x) + \nabla f(x)^T (y - x) + \frac{1}{2\alpha}||y - x||_2^2. \tag{6}$$

   Prove that the minimizer of f_x is the gradient descent update $x - \alpha\nabla f(x)$.

   (b) Inspired by the previous question, how would you modify gradient descent to minimize a function of the form

$$h(x) = f_1(x) + f_2(x), \tag{7}$$

   where f_1 is differentiable, and f_2 is nondifferentiable but has a proximal operator that is easy to compute?

(c) Show that a vector x^* is a solution to

$$\text{minimize} \quad f_1(x) + f_2(x), \tag{8}$$

where f_1 is differentiable, f_2 is nondifferentiable and both functions are convex, if and only if it is a fixed point of the iteration you proposed in the previous question for any $\alpha > 0$.

3. (Iterative shrinkage-thresholding algorithm)

(a) What is the proximal gradient update corresponding to the lasso problem defined below? Your answer will involve a hyperparameter which we will call α.

$$\frac{1}{2}||y - X\beta||_2^2 + \lambda|\beta|_1.$$

(b) How would you check whether you have reached an optimum? How would you modify this to take into account possible numerical inaccuracies?

(c) Implement the method and apply it to the problem in `pgd_lasso-question.ipynb`. You have to fill in blocks of code corresponds to the proximal gradient update step and termination condition. Report all the generated plots.

4. (Forward selection) A very simple way to fit a sparse linear model is to build it gradually by greedily selecting features that are correlated with the residual. This is called *forward selection* in statistics. Let $y_{\text{train}} \in \mathbb{R}^n$ be a vector containing the response, and $X \in \mathbb{R}^{p \times n}$ the corresponding feature matrix. We initialize the set $\mathcal{S}$ of selected features to be empty and the residual to equal $r := y$. Then we update the model until $\mathcal{S}$ contains a predetermined number of features k (which can be chosen by cross validation). The updates consist of incorporating the feature that is most correlated with the residual, and recomputing the residual,

$$i^* := \arg\max_i \left| x_i^T r \right|, \tag{9}$$

$$\mathcal{S} := \mathcal{S} \cup \{i^*\}, \tag{10}$$

$$r := y - X_{\mathcal{S}}^T \beta_{\mathcal{S}}, \qquad \beta_{\mathcal{S}} := \left(X_{\mathcal{S}}^T X_{\mathcal{S}}\right)^{-1} X_{\mathcal{S}} y. \tag{11}$$

Implement this approach and apply it to the temperature example from the sparse regression notes by filling in the `FSR` function in the notebook `FSR.ipynb` Submit and describe all the plots generated by the notebook.