CAP 6610 Machine Learning, Spring 2025

Homework 4

Due Apr. 22 11:59PM

1. *Monotonicity of loss and regularizer as the regularization parameter changes.* In regularized empirical risk minimization, we choose the parameter $\boldsymbol{\theta} \in \mathbb{R}^m$ to minimize the regularized empirical risk, $L(\boldsymbol{\theta}) + \lambda r(\boldsymbol{\theta})$, where $L(\boldsymbol{\theta})$ is the empirical risk, $r(\boldsymbol{\theta})$ is the regularizer, and $\lambda > 0$ is the regularization hyperparameter. (The exact form of the functions L and r is irrelevant in this problem.) The hyperparameter $\lambda > 0$ is used to trade off the two objectives, $L(\boldsymbol{\theta})$ and $r(\boldsymbol{\theta})$. Intuition suggests that as λ increases, $r(\boldsymbol{\theta})$ decreases while $L(\boldsymbol{\theta})$ increases. In this exercise we verify that this is the case.

   Suppose $0 < \lambda < \tilde{\lambda}$. Let $\boldsymbol{\theta}^\star$ minimize $L(\boldsymbol{\theta}) + \lambda r(\boldsymbol{\theta})$ and $\widetilde{\boldsymbol{\theta}}^\star$ minimize $L(\boldsymbol{\theta}) + \tilde{\lambda} r(\boldsymbol{\theta})$.

   (a) Show that $r(\boldsymbol{\theta}^\star) \geq r(\widetilde{\boldsymbol{\theta}}^\star)$. In other words, increasing λ will never make our regularization error larger.

   (b) Show that $L(\boldsymbol{\theta}^\star) \leq L(\widetilde{\boldsymbol{\theta}}^\star)$. In other words, increasing λ will never decrease our training error.

   Hint. Use the fact that $\boldsymbol{\theta}^\star$ is the minimizer of $L(\boldsymbol{\theta}) + \lambda r(\boldsymbol{\theta})$ and similarly for $\widetilde{\boldsymbol{\theta}}^\star$. This means that for any $\boldsymbol{\theta}$, we have $L(\boldsymbol{\theta}) + \lambda r(\boldsymbol{\theta}) \geq L(\boldsymbol{\theta}^\star) + \lambda r(\boldsymbol{\theta}^\star)$, and similarly for $\widetilde{\boldsymbol{\theta}}^\star$.

2. *MAP interpretation of regularized empirical loss minimization.* We have seen that some (unregularized) empirical risk minimization problems can be interpreted as maximum likelihood estimation (MLE) if we choose certain parametric form for the conditional probability $p(y|\boldsymbol{x};\boldsymbol{\theta})$. Assuming the data samples are i.i.d., MLE of $p(y|\boldsymbol{x};\boldsymbol{\theta})$ is equivalent to

$$\underset{\boldsymbol{\theta}}{\text{minimize}} \quad \sum_{i=1}^{n} -\log p(y_i|\boldsymbol{x}_i;\boldsymbol{\theta}).$$

   After some trivial transformations, we can recover some supervised learning models such as least squares regression and logistic classification.

   Some statisticians, who call themselves Bayesians, believe that we should treat $\boldsymbol{\theta}$ as random as well, and impose probability distributions on them. In this case, the probability that we really care about is $p(\boldsymbol{\theta}|Y,\boldsymbol{X})$, the conditional probability of $\boldsymbol{\theta}$ given data $\boldsymbol{X} = \{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n\}$ and $Y = \{y_1,\ldots,y_n\}$. According to Bayes rule,

$$p(\boldsymbol{\theta}|Y,\boldsymbol{X}) = \frac{p(Y|\boldsymbol{X},\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{X})}{p(Y|\boldsymbol{X})}.$$

   Furthermore, it is common to assume that $\boldsymbol{\theta}$ is independent of $\boldsymbol{X}$ and $(\boldsymbol{x}_i, y_i)$ are i.i.d. conditioned on $\boldsymbol{\theta}$, leading to

$$p(\boldsymbol{\theta}|Y,\boldsymbol{X}) = \frac{p(\boldsymbol{\theta})\prod_{i=1}^{n} p(y_i|\boldsymbol{x}_i,\boldsymbol{\theta})}{p(Y|\boldsymbol{X})}.$$

   Here, $p(\boldsymbol{\theta})$ is called the prior (*a priori* in Latin), $p(y|\boldsymbol{x},\boldsymbol{\theta})$ is called the likelihood, and $p(\boldsymbol{\theta}|Y,\boldsymbol{X})$ is called the posterior (*a posteriori* in Latin).

   Depending on the definition of the prior and the likelihood, the denominator $p(Y|\boldsymbol{X})$ may be very hard to evaluate. Instead, we can try to find a point estimate $\boldsymbol{\theta}$ that maximizes the posterior probability, which is called maximum *a posteriori* (MAP), since the denominator does not depend on $\boldsymbol{\theta}$ and can be omitted in maximization. This is equivalent to

$$\underset{\boldsymbol{\theta}}{\text{minimize}} \quad \sum_{i=1}^{n} -\log p(y_i|\boldsymbol{x}_i, \boldsymbol{\theta}) - \log p(\boldsymbol{\theta}).$$

For each of the following cases, give an explicit MAP formulation for estimating $\boldsymbol{\theta}$. Find their relationship to the corresponding regularized empirical risk minimization problems. Specifically, give an exact expression for the regularization parameter λ in terms of the prior and likelihood distributions.

(a) $p(y|\boldsymbol{x}, \boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{\phi}^\top \boldsymbol{\theta}, \sigma^2)$ and $p(\boldsymbol{\theta}) \sim \mathcal{N}(0, \sigma_0^2 \boldsymbol{I})$;

(b) $p(y|\boldsymbol{x}, \boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{\phi}^\top \boldsymbol{\theta}, \sigma^2)$ and $p(\boldsymbol{\theta})$ follows a multivariate Laplacian distribution:

$$p(\boldsymbol{\theta}) = \prod_{j=1}^{m} \frac{1}{2a} \exp\left(-\frac{|\theta_j|}{a}\right);$$

(c) $p(y|\boldsymbol{x}, \boldsymbol{\theta}) = \Pr[yu \geq 0]$ where $y = \pm 1$, $p(u|\boldsymbol{x}, \boldsymbol{\theta}) \sim \mathcal{N}(\boldsymbol{\phi}^\top \boldsymbol{\theta}, \sigma^2)$ and $p(\boldsymbol{\theta}) \sim \mathcal{N}(0, \sigma_0^2 \boldsymbol{I})$;

(d) $p(y|\boldsymbol{x}, \boldsymbol{\theta}) = 1/(1 + \exp(-y\boldsymbol{\phi}^\top \boldsymbol{\theta}))$ where $y = \pm 1$ and $p(\boldsymbol{\theta})$ follows a multivariate Laplacian distribution as in (b).

3. *Proximal operator for the group lasso regularizer.* In this exercise we derive the proximal operator for the group lasso regularizer. We will be using the notion of subgradient to make the derivation more solid.

Recall that the proximal operator of the function $\rho \sum_{j=1}^{m} \|\boldsymbol{\Theta}_j\|$ at a particular point $\widetilde{\boldsymbol{\Theta}}$, where $\boldsymbol{\Theta}_j$ is the jth row of $\boldsymbol{\theta}$, is the solution to the following optimization problem

$$\underset{\boldsymbol{\Theta}}{\text{minimize}} \quad \rho \sum_{j=1}^{m} \|\boldsymbol{\Theta}_j\| + \frac{1}{2}\|\boldsymbol{\Theta} - \widetilde{\boldsymbol{\Theta}}\|^2.$$

Notice that the problem is separable over $\boldsymbol{\Theta}_j$, so it suffices to consider the simpler problem

$$\underset{\boldsymbol{\Theta}_j}{\text{minimize}} \quad \rho \|\boldsymbol{\Theta}_j\| + \frac{1}{2}\|\boldsymbol{\Theta}_j - \widetilde{\boldsymbol{\Theta}}_j\|^2. \tag{1}$$

(a) Give an expression for the subdifferential of the function $\|\boldsymbol{\Theta}_j\|$. Recall that subdifferential is the set of subgradients, so you need to list all possible subgradients. *Hint.* The function is only non-differentiable when $\boldsymbol{\Theta}_j = 0$; when the function is differentiable, the only element in the subdifferential is the gradient; otherwise it is some convex set.

(b) A necessary and sufficient condition for optimality is that 0 is an element of the subdifferential. Use this condition and the expression of the subdifferential to derive the solution of (1).

4. *News articles classification.* In this problem we revisit the 20 Newsgroup data set from Homework 2 (but this time we keep the term frequencies of all words): `<http://qwone.com/~jason/20Newsgroups/>`. You will design a SGD algorithm for multi-class support vector machine with group-sparse regularization that solves the following optimization problem

$$\underset{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_k}{\text{minimize}} \quad \frac{1}{n} \sum_{i=1}^{n} \max_c (\boldsymbol{x}_i^\top \boldsymbol{\theta}_c - \boldsymbol{x}_i^\top \boldsymbol{\theta}_{y_i} + 1_{y_i \neq c}) + \lambda \sum_{j=1}^{m} \sqrt{\sum_{c=1}^{k} \theta_{jc}^2}.$$

Here we simply assume that the features are the term frequencies themselves (we even ignore the constant 1 here).

(a) Derive the stochastic proximal subgradient algorithm for solving it. For simplicity, you can assume that there is only one term that reaches the maximum value in $\max_c(\boldsymbol{x}^\top\boldsymbol{\theta}_c - \boldsymbol{x}_i^\top\boldsymbol{\theta}_{y_i} + 1_{y_i \neq c})$ throughout the iterations. At iteration t, you can simply denote the step size as $\gamma^{(t)}$.

(b) Implement the algorithm in your favorite programming language.

(c) Run the algorithm with $\lambda = 10, 1, 0.1, 0.01$ and diminishing step size $\gamma^{(t)} = 1/t$, and run the algorithm for 10^6 iterations. At every 1000 iteration, evaluate the prediction accuracy on the test set and plot the progress on a figure.

(d) For the solution of each λ value, list the set of words with zero coefficients in all classes (meaning these words will be ignored when predicting which newsgroup it belongs to). Does the result make sense? Is it true that a large λ leads to a more sparse solution?