STA 250	Optimization	Fall 2017

Homework 2

Lecturer: Cho-Jui Hsieh — Date Due: Nov 21, 11:59am, 2017

Keywords: *Newton method, projected gradient*

Problem 1. Newton-CG Method for L2-regularized Logistic Regression

Given a set of instance-label pairs $(\boldsymbol{x}_i, y_i), i = 1, \ldots, n, \boldsymbol{x}_i \in \mathbb{R}^d$, $y_i \in \{+1, -1\}$, L2-regularized logistic regression estimates the classification model $\boldsymbol{w}$ by solving the following optimization problem:

$$\min_{\boldsymbol{w}\in\mathbb{R}^d} \left\{ \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \sum_{i=1}^{n} \log(1 + \exp(-y_i \boldsymbol{w}^T \boldsymbol{x}_i)) \right\} := f(\boldsymbol{w}), \tag{1}$$

The data matrix is $X \in \mathbb{R}^{n\times d}$ is dense, where each row of X is a training sample. We test the algorithm using the "breast-cancer" dataset with $n = 44$ and $d = 7129$: the training data matrix X is stored in "X_cancer", and the training labels $\boldsymbol{y}$ is stored in "y_cancer" (the format is the same with hw1). Note that you are allowed to use any coding language or package in this homework.

(a) (5 pt) Derive the gradient and Hessian of $f(\boldsymbol{w})$.

(b) (15 pt) We plan to apply CG to solve the linear system (compute $\nabla^2 f(\boldsymbol{w})^{-1}\nabla f(\boldsymbol{w})$), where the main computation is the matrix vector product. The matrix vector product of $\nabla^2 f(\boldsymbol{w}^0)\boldsymbol{v}$ will need $O(d^2)$ time if we form the d by d matrix $\nabla^2 f(\boldsymbol{w})$ and then directly compute the matrix vector product. Explain a faster way to compute $\nabla^2 f(\boldsymbol{w})\boldsymbol{v}$ for an arbitrary vector $\boldsymbol{v} \in \mathbb{R}^d$ with $O(nd)$ time.

(c) (15 pt) Use the matrix-vector product derived in (c) to implement Conjugate Gradient method (see Algorithm 1) for solving the linear system $(\nabla^2 f(\boldsymbol{w}^0)^{-1}\nabla f(\boldsymbol{w}^0))$ up to $\|\boldsymbol{r}\|/\|\boldsymbol{r}_0\| \le 10^{-3}$ (where $\boldsymbol{r}$ is the residual in the algorithm). Report how much time does the algorithm take on the breast-cancer dataset. Report $\|\Delta\boldsymbol{x}^{\mathrm{CG}}\|_2$ where $\Delta\boldsymbol{x}^{\mathrm{CG}}$ is the output of CG.

(d) (15 pt) Implement the Newton-CG algorithm (see Algorithm 2). Initial from $\boldsymbol{w}^0 = \boldsymbol{0}$ and run 20 iterations. What is the objective function value $f(\boldsymbol{w}^{20})$? Plot the iteration vs error figure: x-axis is $k = 1, \ldots, 18$ and y-axis is the corresponding error $\log(f(\boldsymbol{w}^k) - f(\boldsymbol{w}^*))$. In practice it is impossible to get the exact optimal solution $\boldsymbol{w}^*$, so instead we assume $f(\boldsymbol{w}^*) = f(\boldsymbol{w}^{20})$ when we plot the figure.

Problem 2. Nonnegative Matrix Factorization

Given an input matrix $X \in \mathbb{R}^{m\times n}$, we try to factorize it into $X \approx WH^T$ by solving

$$\text{(NMF)} \quad \min_{W\in\mathbb{R}^{m\times k}, H\in\mathbb{R}^{n\times k}} \frac{1}{2}\|A - WH^T\|_F^2 + \frac{\lambda}{2}\|W\|_F^2 + \frac{\lambda}{2}\|H\|_F^2,$$
$$\text{s.t. } W_{ij} \ge 0, H_{ij} \ge 0 \quad \forall i, j$$

We will use the cbcl dataset in "cbcl.txt". In this data, $m = 361$, $n = 2,429$, and we set $k = 49$.

(a) (20 pt) Apply a block-coordinate descent algorithm to solve NMF. At each iteration, we first fix W and update H, and then fix H and update W. For each subproblem, update W (or H) using projected gradient descent with 3 steps (see Algorithm 3). Use random initialization for W, H. Run the algorithm for 50 iterations and plot the objective function value vs time curve. Report the final objective function value you get.

(b) (30 pt) Now try block coordinate descent by defining each column of W, H as a block. So there are totally $2k$ blocks (see Algorithm 4). For updating w_i (for some $1 \leq i \leq k$), we want to minimize the function with respect to w_i. Derive the corresponding subproblem and show the close form solution of w_i. Similarly run 50 iterations, report the final objective function value, and plot the objective function value vs time curve. Compare the two methods and discuss your findings.

Algorithm 1 Conjugate Gradient to Solve $A\boldsymbol{x} = \boldsymbol{b}$

- Input: A and $\boldsymbol{b}$
- $\boldsymbol{x}_0 = \underline{0}$
- $\boldsymbol{r}_0 = \boldsymbol{b} - A\boldsymbol{x}_0$
- $\boldsymbol{p}_0 = \boldsymbol{r}_0$
- $k = 0$
- For $k = 0, 1, \dots$
  1. $\alpha_k = \frac{\boldsymbol{r}_k^T \boldsymbol{r}_k}{\boldsymbol{p}_k^T A \boldsymbol{p}_k}$
  2. $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k$
  3. $\boldsymbol{r}_{k+1} = \boldsymbol{r}_k - \alpha_k A \boldsymbol{p}_k$
  4. If $\frac{\|\boldsymbol{r}_{k+1}\|}{\|\boldsymbol{r}_0\|} \leq 10^{-3}$, exit the for loop
  5. $\beta_k = \frac{\boldsymbol{r}_{k+1}^T \boldsymbol{r}_{k+1}}{\boldsymbol{r}_k^T \boldsymbol{r}_k}$
  6. $\boldsymbol{p}_{k+1} = \boldsymbol{r}_{k+1} + \beta_k \boldsymbol{p}_k$

Algorithm 2 Newton method for L2-regularized logistic regression

- Input: $\{\boldsymbol{x}_i, y_i\}_{i=1}^n$, regularization parameter λ, initial $\boldsymbol{w}^0 = \boldsymbol{0}$.
- For iter $= 1, 2, \dots, 10$
  1. Solving $\nabla^2 f(\boldsymbol{w})\boldsymbol{d} = -\nabla f(\boldsymbol{w})$ by Algorithm 1 to get $\boldsymbol{d}$.
  2. Find the maximum step size $\alpha = \max\{1, 2^{-1}, 2^{-2}, \dots\}$ such that $\alpha\boldsymbol{d}$ satisfies the following line search condition:
  $$f(\boldsymbol{w} + \alpha\boldsymbol{d}) < f(\boldsymbol{w}) + 0.01\alpha\boldsymbol{d}^T \nabla f(\boldsymbol{w}).$$
  3. $\boldsymbol{w} \leftarrow \boldsymbol{w} + \alpha\boldsymbol{d}$.

Algorithm 3 Alternating minimization + Projected gradient descent for NMF

- Input: A, λ
- For $k = 0, 1, \dots$
  1. Fix W and update H: solving the following subproblem with 3 projected gradient steps:
  $$\min_H \frac{1}{2}\|A - WH^T\|_F^2 + \frac{\lambda}{2}\|H\|_F^2 \quad \text{s.t.} \quad H_{ij} \geq 0, \ \forall i, j$$
  2. Fix H and update W: solving the following subproblem with 3 projected gradient steps:
  $$\min_W \frac{1}{2}\|A - WH^T\|_F^2 + \frac{\lambda}{2}\|W\|_F^2 \quad \text{s.t.} \quad W_{ij} \geq 0, \ \forall i, j$$

Algorithm 4 Block-coordinate descent for NMF—update one column at a time.

- Input: A, λ, initial values for W, H
- For $t = 0, 1, \ldots$
  1. For $i = 1, \cdots, k$

     Update w_i (i-th column of W)
  2. For $i = 1, \cdots, k$

     Update h_i (i-th column of H)