Project Description: In this project you will implement algorithms for (regularized) logistic regression. The functions should be implemented in C and the plotting should be done in R.

Notes: (1) Use double precision for all real numbers; (2) Set your starting point $\beta^0 = 0$ for easy comparison; (3) Use the stopping criteria $\max_j |\nabla f_j| < 10^{-4}$ where $f(.)$ is the objective function in the optimization; (4) assume $\beta \in \mathbb{R}^p$.

Problem. 1

(1) Implement C functions `logisticGD` and `logisticCD` which solve the regularized logistic regression

$$\hat{\beta}(\lambda) = \arg \min_{\beta} \left\{ \frac{1}{n} \sum_{i=1}^{n} \log(1 + \exp(-y_i \beta^T x_i)) + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

using gradient descent and coordinate descent, respectively. Use the dataset provided in `dataset_X.txt` and `dataset_Y.txt`. Report the final function’s value, number of iterations to converge for each algorithm for each $\lambda$ in the set $\{0.0001, 0.001, 0.01, 0.1, 1\}$.

(2) Use $\lambda = 0.01$ in this problem. Report the final function’s value and plot both positive and negative examples (using different symbols) and the decision boundary in R. What might happen if we use $\lambda = 0$ in this case and explain why?

Problem. 2: Consider again the optimization problem

$$\hat{\beta}(\lambda) = \arg \min_{\beta} \left\{ \frac{1}{n} \sum_{i=1}^{n} \log(1 + \exp(-y_i \beta^T x_i)) + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

and let the objective function be $f(\beta)$.

(1) Compute its Hessian matrix $H(\beta) \in \mathbb{R}^{p \times p}$ and find a positive definite matrix $U$ which is a uniform upperbound of $H$ (that does not depend on $\theta$) (i.e. $U - H(\beta)$ is positive semi-definite for any $\beta$). Prove that the following iterative update rule will monotonically decrease the objective functions’s value:

$$\beta_{t+1} = \beta_t - U^{-1}g(\beta_t)$$

where $g(\beta_t) \in \mathbb{R}^p$ is the gradient of $f(.)$ at $\beta_t$, $\beta_t$ is the parameter value in the previous optimization step, and $\beta_{t+1}$ is the new parameter value.

(2) Implement your algorithm using this approach and compare it with the one with gradient descent and coordinate descent in terms of computing time and number of iterations.

Problem. 3: Suppose your input data $x_1, \ldots, x_n$ are sparse, how should you change your implementation to make it more efficient (for the above regularized logistic regression)? Describe your solution in terms of both the data structure and algorithms.

Date: March 30, 2010.