Logistic Regression Implementation and Optimization

Zhu Li

January 2017

1 Introduction to Logistic Regression

Suppose we have a sample vector X that can be classified into 2 categories $Y : 0, 1$. We could estimate its probability of being in class $Y = 1$ and in class $Y = 0$ with:

$$P(Y = 1|X) = \frac{1}{1 + exp(w_0 + \sum_{i=1}^{n} w_i X_i)}$$

and

$$P(Y = 0|X) = \frac{exp(w_0 + \sum_{i=1}^{n} w_i X_i)}{1 + exp(w_0 + \sum_{i=1}^{n} w_i X_i)}$$

The class of X is determined by the larger of the above results. The parameters $W([w_0, w_1, w_2, ...])$ can be initialized with small random values and updated with the following formula(the deduction is skipped here):

$$w_i \Leftarrow w_i + \eta \sum_{l} X_i^l (Y^l - \hat{P}(Y^l = 1|X^l, W))$$

, where η is a parameter that controls the speed of convergence, X^l represents the lth training sample, and $\hat{P}(Y^l = 1|X^l, W)$ represents the Logistic Regression prediction using the first 2 formulas.

2 Project Plan

1. Implement the Logistic Regression algorithm with a single thread C program.

2. Optimize the single thread C program with several ways like making proper use of the registers and compiler options, simplifying for loops, reusing allocated space as mentioned in Peter's video, simplying the process with some unique characters of Logistic Regression and so on.

3. Optimize the original program with multiple threads. This is possible because, as we can see, the above formulas involve a lot of vector multiplications and sums.

4. Optimize the program with GPU.