Supplementary Materials for "Subsampled Hessian Newton Methods for Supervised Learning"

Chien-Chih Wang, Chun-Heng Huang, Chih-Jen Lin

Department of Computer Science, National Taiwan University, Taipei 10617, Taiwan

I Introduction

This document presents some materials that are not included in the paper. In Section II, we show experiments of using a subset of data for calculating the gradient. In Section III, we discuss some variants of the proposed method. Section IV gives details of applying our method to maximum entropy.

II Using a Subset of Data for Calculating the Gradient

We mentioned in Section 1 that Byrd et al. [2011] consider using a subset R so that

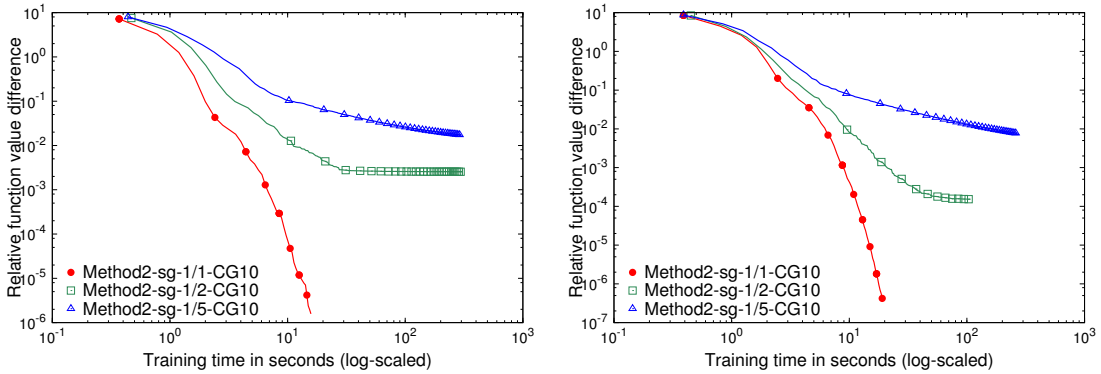
$$\nabla f(\boldsymbol{w}) \approx \frac{1}{|R|} \sum_{i \in R} \nabla \xi(\boldsymbol{w}; \boldsymbol{x}_i, y_i).$$

Byrd et al. [2011] conducted experiments by using $R = \{1, \ldots, l\}$, so only the Hessian is approximated by a subset of data. We follow the same setting in the paper. One reason of not subsampling points for gradient evaluation is that in the line search procedure we still need to access the whole set for computing the function value.

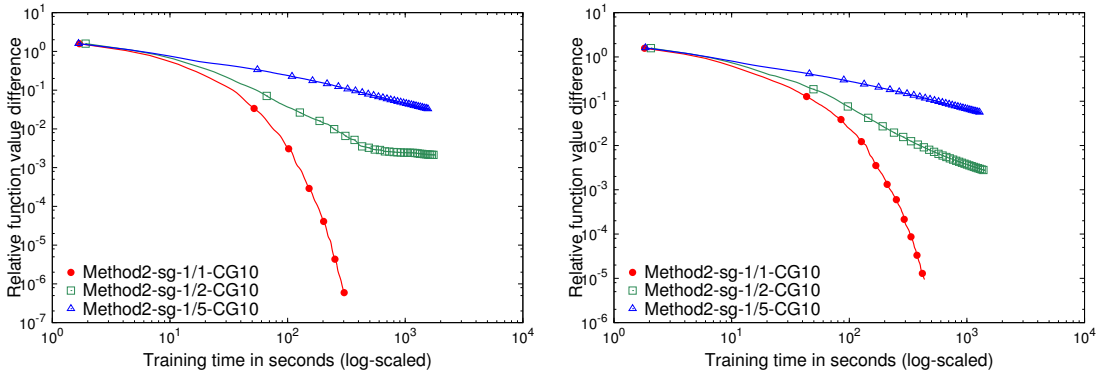
Here we compare the following two settings:

1. *Method 2*: the proposed method in the paper; see Section 6.1.
2. *Method 2-sg*: the method is the same as *Method 2* except that data are subsampled for calculating the gradient. For example, *Method 2-sg-1/2-CG10* is that we only use a subset R with $|R| = 50\%l$ to derive the gradient.

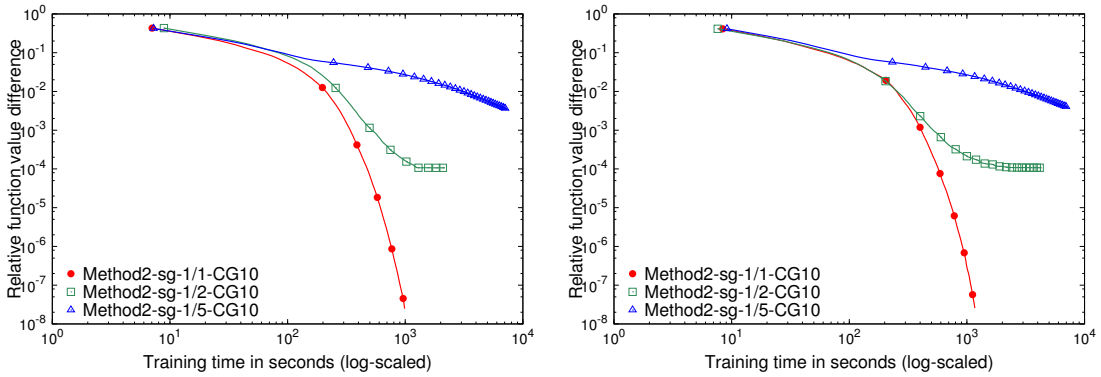
Note that a subset S_k of R_k is further selected for the Hessian Calculation. The comparison results on logistic regression are in Figure II.1. Clearly, *Method 2-sg* is much slower. Our results indicate that because one pass of data can yield both function and gradient values, there may be no need to subsample points for the gradient calculation.



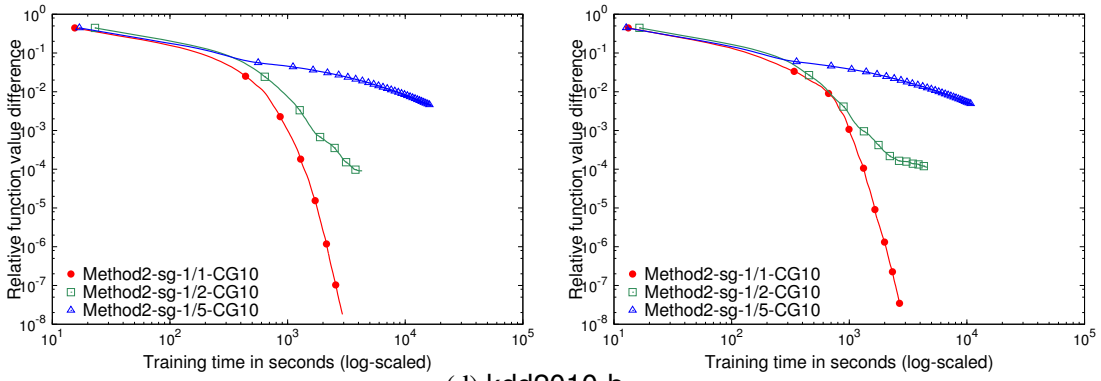
(a) news20



(b) yahoo-korea



(c) kdd2010-a



(d) kdd2010-b

Figure II.1: Experiments on approaches with/without using a subset for gradient calculation. Logistic regression is considered. We present running time (in seconds) versus the relative difference to the optimal function value. Both x-axis and y-axis are log-scaled. Left: $|S_k|/l = 5\%$. Right: $|S_k|/l = 1\%$.

III Some Variants of the Proposed Method

In Section III.1, we discuss the selection of $\bar{\boldsymbol{d}}_k$. In Section III.2, we investigate the use of non-negative β_1 and β_2 for calculating the direction $\beta_1 \boldsymbol{d}_k + \beta_2 \bar{\boldsymbol{d}}_k$ in our proposed method. Experimental comparisons are in Section III.3.

III.1 Selection of $\bar{\boldsymbol{d}}_k$

An important issue for *Method 2* is how to select an appropriate $\bar{\boldsymbol{d}}_k$. The choice of $\bar{\boldsymbol{d}}_k$ affects the convergence speed. In the paper, we use $\bar{\boldsymbol{d}}_k = \boldsymbol{d}_{k-1},\ k \geq 1$. Here we try another setting

$$\bar{\boldsymbol{d}}_k = -\nabla f(\boldsymbol{w}_k).$$

III.2 Using Non-Negative β_1 and β_2

The coefficients β_1 and β_2 of solving (13) may be negative. It is interesting to check if imposing non-negativity can lead to a better direction. Therefore, we replace (13) with the following optimization problem.

$$\begin{aligned} \min_{\beta_1,\beta_2} \quad & \frac{1}{2}(\beta_1 \boldsymbol{d}_k + \beta_2 \bar{\boldsymbol{d}}_k)^T H_k (\beta_1 \boldsymbol{d}_k + \beta_2 \bar{\boldsymbol{d}}_k) + \nabla f(\boldsymbol{w}_k)^T (\beta_1 \boldsymbol{d}_k + \beta_2 \bar{\boldsymbol{d}}_k) \\ \text{subject to} \quad & \beta_1 \geq 0,\ \beta_2 \geq 0. \end{aligned} \tag{III.1}$$

Although (13) has a closed-form solution, (III.1) does not. We derive a solution procedure by checking its optimality condition. Let

$$\begin{aligned} a = \boldsymbol{d}_k^T H_k \boldsymbol{d}_k,\ b = \bar{\boldsymbol{d}}_k^T H_k \boldsymbol{d}_k,\ c = \bar{\boldsymbol{d}}_k^T H_k \bar{\boldsymbol{d}}_k, \\ e = -\nabla f(\boldsymbol{w}_k)^T \boldsymbol{d}_k,\ f = -\nabla f(\boldsymbol{w}_k)^T \bar{\boldsymbol{d}}_k. \end{aligned}$$

Problem (III.1) can be rewritten as

$$\begin{aligned} \min_{\beta_1,\beta_2} \quad & \frac{1}{2} \begin{bmatrix} \beta_1 & \beta_2 \end{bmatrix} \begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} - \begin{bmatrix} e & f \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} \\ \text{subject to} \quad & \beta_1 \geq 0,\ \beta_2 \geq 0. \end{aligned} \tag{III.2}$$

We show that if

$$\boldsymbol{d}_k \neq \mathbf{0},\ \bar{\boldsymbol{d}}_k \neq \mathbf{0},\ \text{and } \boldsymbol{d}_k \neq \bar{\boldsymbol{d}}_k, \tag{III.3}$$

then

$$a > 0,\ c > 0,\ \text{and } ac - b^2 > 0. \tag{III.4}$$

The first two inequalities hold because H_k is positive definite. For the third inequality, we have Cholesky factorization of H_k

$$H_k = LL^T,$$

where L is lower triangular and invertible. Then

$$ac - b^2 = ||L\boldsymbol{d}_k||^2 ||L\bar{\boldsymbol{d}}_k||^2 - ((L\boldsymbol{d}_k)^T (L\bar{\boldsymbol{d}}_k))^2 > 0$$

by Cauchy inequality and the assumption $\boldsymbol{d}_k \neq \bar{\boldsymbol{d}}_k$. We check if the three conditions in (III.3) can be easily fulfilled. Because $\boldsymbol{w}_k$ is not an optimal solution yet, $\nabla f(\boldsymbol{w}_k) \neq \mathbf{0}$ and so is $\boldsymbol{d}_k$ from the CG procedure of using $\nabla f(\boldsymbol{w}_k)$ at the first step. The direction $\bar{\boldsymbol{d}}_k$ can be easily chosen so that the other two conditions hold.

The KKT optimality condition of (III.2) is that there are λ_1 and λ_2 such that

$$\begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} - \begin{bmatrix} e \\ f \end{bmatrix} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix},$$
$$\lambda_1 \beta_1 = 0,\ \lambda_2 \beta_2 = 0,$$
$$\beta_1 \geq 0,\ \beta_2 \geq 0,\ \lambda_1 \geq 0,\ \lambda_2 \geq 0.$$

We consider the following three situations

$$ec - bf \geq 0,\ af - be \geq 0 \tag{III.5}$$
$$bf - ce \geq 0,\ f \geq 0 \tag{III.6}$$
$$be - af \geq 0,\ e \geq 0 \tag{III.7}$$

- If (III.5) holds,

$$\beta_1 = \frac{ec - bf}{ac - b^2},\ \beta_2 = \frac{af - be}{ac - b^2},\ \lambda_1 = 0,\ \lambda_2 = 0$$

  satisfy the KKT condition.

- If (III.6) holds,

$$\beta_1 = 0,\ \beta_2 = \frac{f}{c},\ \lambda_1 = \frac{bf}{c} - e,\ \lambda_2 = 0$$

  satisfy the KKT condition.

- If (III.7) holds,

$$\beta_1 = \frac{e}{a},\ \beta_2 = 0,\ \lambda_1 = 0,\ \lambda_2 = \frac{be}{a} - f$$

  satisfy the KKT condition.

The following theorem shows that for all remaining situations, $(\beta_1, \beta_2) = (0, 0)$ is an optimal solution.

Theorem 1. *If none of* (III.5)*,* (III.6)*,* (III.7) *holds, then*

$$\beta_1 = \beta_2 = 0$$

is optimal for (III.1).

Proof. We show that if none of (III.5), (III.6), (III.7) holds, then

$$f \leq 0 \text{ and } e \leq 0. \tag{III.8}$$

Then

$$\beta_1 = 0,\ \beta_2 = 0,\ \lambda_1 = -e,\ \lambda_2 = -f$$

satisfy the KKT condition.

If none of (III.5), (III.6), (III.7) holds, then we have

$$\begin{aligned} &(ec - bf < 0 \text{ or } af - be < 0), \text{ and} \\ &(bf - ce < 0 \text{ or } f < 0), \text{ and} \\ &(be - af < 0 \text{ or } e < 0). \end{aligned} \tag{III.9}$$

We argue that the above condition implies (III.8). Otherwise, if (III.8) does not hold, we have

$$f > 0 \text{ or } e > 0. \tag{III.10}$$

Consider the first situation where

$$f > 0.$$

Then (III.9) implies

$$\begin{aligned} &bf - ce < 0, \\ &af - be < 0, \\ &e < 0. \end{aligned}$$

From (III.4),

$$b < \frac{ce}{f} < 0 \text{ and } b < \frac{af}{e} < 0$$

lead to

$$b^2 > ac,$$

which is a contradiction to (III.4). The situation for

$$e > 0$$

is similar. Therefore, (III.8) holds and the proof is complete. □

III.3 Experiments

We compare the following three methods in Figures III.2 and III.3 respectively for logistic regression and l2-loss SVM.

1. *Method 2*: the method proposed in the paper.
2. *Method 2-g*: the same as *Method 2* except using $\bar{\boldsymbol{d}}_k = -\nabla f(\boldsymbol{w}_k)$.
3. *Method 2-con*: the same as *Method 2* except using non-negative β_1 and β_2.

We have the following observations.

1. *Method 2-g* is slower than *Method 2*. We have mentioned in the paper that the superiority of $\bar{\boldsymbol{d}}_k = \boldsymbol{d}_{k-1}$ may be because it comes from solving a sub-problem of using some second-order information. Further, $-\nabla f(\boldsymbol{w}_k)$, like $\boldsymbol{d}_k$, uses information of the current iteration while additional information from another subset of data is employed to find $\boldsymbol{d}_{k-1}$.

2. *Method 2-con* is slightly slower than *Method 2*. Although β_1 or β_2 by *Method 2* may be negative and cause some difficulties to interpret what the direction $\beta_1 \boldsymbol{d}_k + \beta_2 \bar{\boldsymbol{d}}_k$ is, they lead to the smallest second-order approximation of the function-value reduction. This property may explain *Method 2*'s faster convergence.

IV Details of Applying the Proposed Method to Maximum Entropy

For maximum entropy, the optimization problem is

$$\min_{\boldsymbol{w}} f(\boldsymbol{w}), \text{ where } f(\boldsymbol{w}) \equiv \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{C}{l}\sum_{i=1}^{l}\big(\log(\sum_{c=1}^{k}\exp(\boldsymbol{w}_c^T\boldsymbol{x}_i)) - \boldsymbol{w}_{y_i}^T\boldsymbol{x}_i\big).$$

Let

$$P_{i,s} = \frac{\exp(\boldsymbol{w}_s^T\boldsymbol{x}_i)}{\sum_{c=1}^{k}\exp(\boldsymbol{w}_c^T\boldsymbol{x}_i)}.$$

The gradient of $f(\boldsymbol{w})$ is

$$\nabla f(\boldsymbol{w}) = \begin{bmatrix} \nabla f^1(\boldsymbol{w}) \\ \vdots \\ \nabla f^k(\boldsymbol{w}) \end{bmatrix},$$

where

$$\nabla f^t(\boldsymbol{w}) = \boldsymbol{w}_t + \frac{C}{l}(\sum_{i=1}^{l} P_{i,t}\boldsymbol{x}_i - \sum_{i:y_i=t}\boldsymbol{x}_i) \in R^{n\times 1}.$$

Because $f(\boldsymbol{w})$ is twice continuously differentiable, the Hessian matrix of $f(\boldsymbol{w})$ can be derived.

- Case 1: When $t = s$,

$$\begin{aligned}\nabla^2 f(\boldsymbol{w})^{t,s} &= I_{n\times n} + \frac{C}{l}\sum_{i=1}^{l}\left(\frac{\exp(\boldsymbol{w}_t^T\boldsymbol{x}_i)\boldsymbol{x}_i\boldsymbol{x}_i^T}{\sum_{c=1}^{k}\exp(\boldsymbol{w}_c^T\boldsymbol{x}_i)} - \frac{(\exp(\boldsymbol{w}_t^T\boldsymbol{x}_i)\boldsymbol{x}_i)(\exp(\boldsymbol{w}_s^T\boldsymbol{x}_i)\boldsymbol{x}_i)^T}{(\sum_{c=1}^{k}\exp(\boldsymbol{w}_c^T\boldsymbol{x}_i))^2}\right) \\ &= I_{n\times n} + \frac{C}{l}\sum_{i=1}^{l}\big((P_{i,t}\boldsymbol{x}_i)\boldsymbol{x}_i^T - (P_{i,t}\boldsymbol{x}_i)(P_{i,s}\boldsymbol{x}_i)^T\big) \in R^{n\times n},\end{aligned}$$

  where $I_{n\times n}$ is an identity matrix.

- Case 2: When $t \neq s$,

$$\begin{aligned}\nabla^2 f(\boldsymbol{w})^{t,s} &= \frac{C}{l}\sum_{i=1}^{l}\frac{-(\exp(\boldsymbol{w}_t^T\boldsymbol{x}_i)\boldsymbol{x}_i)(\exp(\boldsymbol{w}_s^T\boldsymbol{x}_i)\boldsymbol{x}_i)^T}{(\sum_{c=1}^{k}\exp(\boldsymbol{w}_c^T\boldsymbol{x}_i))^2} \\ &= \frac{C}{l}\sum_{i=1}^{l}(P_{i,t}\boldsymbol{x}_i)(P_{i,s}\boldsymbol{x}_i)^T \in R^{n\times n}.\end{aligned}$$

Therefore, the Hessian-vector product is

$$H\boldsymbol{v} = \begin{bmatrix} (H\boldsymbol{v})^1 \\ \vdots \\ (H\boldsymbol{v})^k \end{bmatrix}, \text{ where } \boldsymbol{v} = \begin{bmatrix} \boldsymbol{v}_1 \\ \vdots \\ \boldsymbol{v}_k \end{bmatrix}$$

and

$$(H\boldsymbol{v})^t = \boldsymbol{v}_t + \frac{C}{l}\sum_{i=1}^{l}\big(P_{i,t}(\boldsymbol{v}_t^T\boldsymbol{x}_i - \sum_{c=1}^{k} P_{i,c}\boldsymbol{v}_c^T\boldsymbol{x}_i)\big)\boldsymbol{x}_i.$$

For the convergence analysis, we represent $\nabla^2 f(\boldsymbol{w})$ in a form similar to (29) for logistic regression.

$$\nabla^2 f(\boldsymbol{w}) = I_{kn\times kn} + \frac{C}{l}\bar{X}^T(D-E)\bar{X}, \tag{IV.11}$$

where

$$\bar{X} = \begin{bmatrix} X & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & X & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & X \end{bmatrix} \in R^{kl\times kn},$$

$$D = \begin{bmatrix} D^1 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & D^2 & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & D^k \end{bmatrix} \in R^{kl\times kl}, \text{ and } E = \begin{bmatrix} E^{11} & \cdots & \cdots & E^{1k} \\ E^{21} & E^{22} & \cdots & E^{2k} \\ \vdots & \vdots & \vdots & \vdots \\ E^{k1} & E^{k2} & \cdots & E^{kk} \end{bmatrix} \in R^{kl\times kl}. \tag{IV.12}$$

In (IV.12), D^s is a diagonal matrix with

$$D^s_{ii} = P_{i,s},$$

and E^{ts} is also a diagonal matrix with

$$E^{ts}_{ii} = P_{i,t}P_{i,s}.$$

Using (IV.11) we will prove Theorem 1 for the convergence. The only difference from that for logistic regression is on the boundedness of $||H_{S_k}||$ and $||\nabla^2 f(\boldsymbol{w})||$. From (IV.11) and the fact that $|D^t_{ii}| \le 1$ and $|E^{ts}_{ii}| \le 1$, $\forall i = 1,\ldots,l$, $||\nabla^2 f(\boldsymbol{w})||$ is bounded by

$$\begin{aligned} ||\nabla^2 f(\boldsymbol{w})|| &\le 1 + \frac{C}{l}||\bar{X}^T||||D-E||||\bar{X}|| \\ &\le 1 + \frac{C}{l}||(\bar{X}^T||(||D|| + ||-E||)||\bar{X}|| \\ &\le 1 + \frac{C}{l}(1+\sqrt{kl})||\bar{X}^T||||\bar{X}||. \end{aligned}$$

For $||H_{S_k}||$, it is easy to have that

$$1 \le ||H_{S_k}|| \le ||\nabla^2 f(\boldsymbol{w}_k)||.$$

V Details of Hessian-free Approaches for Neural Networks

The idea of calculating the Hessian-vector product is to define the following $\mathcal{R}$-operator for any function of $\boldsymbol{\theta}$ and then repeatedly apply the chain rule as follows:

$$\mathcal{R}_{\boldsymbol{v}}\{f\} = \lim_{\epsilon \to 0} \frac{f(\boldsymbol{\theta} + \epsilon \boldsymbol{v}) - f(\boldsymbol{\theta})}{\epsilon}.$$

Then

$$H_k \boldsymbol{v} = \lim_{\epsilon \to 0} \frac{\nabla f(\boldsymbol{\theta} + \epsilon \boldsymbol{v}) - \nabla f(\boldsymbol{\theta})}{\epsilon} = \mathcal{R}_{\boldsymbol{v}}\{\nabla f\}. \tag{V.13}$$

For convenience, we replace $\mathcal{R}_{\boldsymbol{v}}$ with $\mathcal{R}$. To use (V.13), we first obtain ∇f by the following operations. Let $\boldsymbol{x}$ and $\boldsymbol{z}$ denote the $\boldsymbol{x}^m$ and $\boldsymbol{z}^m$ vectors of the mth layer, respectively. Further, we denote w_{ij}, $i = 1, \ldots, n_m$, $j = 1, \ldots, n_{m-1}$ as elements of the weight matrix W^m and let $\bar{\boldsymbol{z}}$ be the vector $\boldsymbol{z}^{m-1}$. Assume $\partial f / \partial z_i$, $i = 1, \ldots, n_m$ are available. From (35), we have

$$\begin{aligned}
\frac{\partial f}{\partial x_i} &= \frac{\partial f}{\partial z_i} \sigma'(x_i), \\
\frac{\partial f}{\partial w_{ij}} &= \frac{\partial f}{\partial x_i} \bar{z}_j, \\
\frac{\partial f}{\partial \bar{z}_j} &= \sum_{i=1}^{n_m} w_{ij} \frac{\partial f}{\partial x_i}.
\end{aligned}$$

This backward process can be computed from the last to the first layer. In the end the collection of $\nabla_{W^1} f, \ldots, \nabla_{W^L} f$ is $\nabla f(\boldsymbol{\theta})$.

To obtain $H_k \boldsymbol{v}$, we apply the $\mathcal{R}$-operator to the above terms and obtain

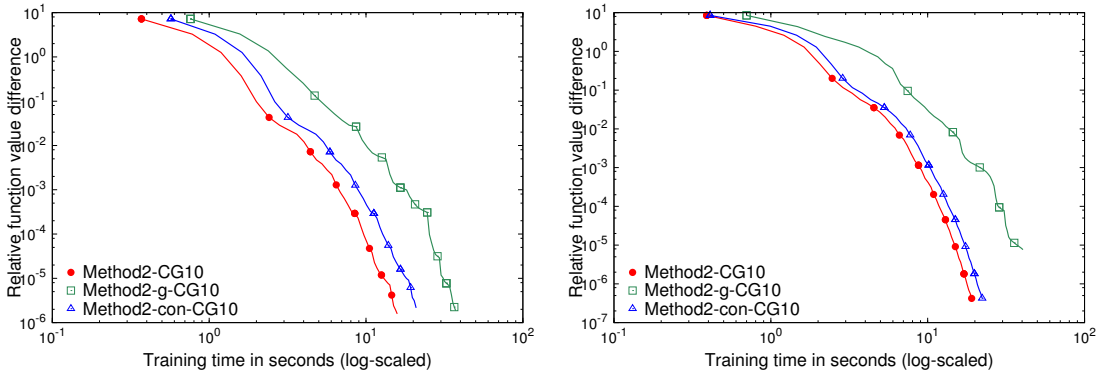
$$\begin{aligned}
\mathcal{R}\{\frac{\partial f}{\partial x_i}\} &= \sigma'(x_i) \mathcal{R}\{\frac{\partial f}{\partial z_i}\} + \frac{\partial f}{\partial z_i} \sigma''(x_i) \mathcal{R}\{x_i\}, \\
\mathcal{R}\{\frac{\partial f}{\partial w_{ij}}\} &= \bar{z}_j \mathcal{R}\{\frac{\partial f}{\partial x_i}\} + \mathcal{R}\{\bar{z}_j\} \frac{\partial f}{\partial x_i}, \\
\mathcal{R}\{\frac{\partial f}{\partial \bar{z}_j}\} &= \sum_{i=1}^{n_m} (w_{ij} \mathcal{R}\{\frac{\partial f}{\partial x_i}\} + v_{ij} \frac{\partial f}{\partial x_i}),
\end{aligned}$$

where v_{ij} is an element of the vector $\boldsymbol{v}$ in (V.13). It corresponds to w_{ij}, so $\mathcal{R}(w_{ij}) = v_{ij}$. We see that $\mathcal{R}\{x_i\}$ and $\mathcal{R}\{\bar{z}_j\}$ are also needed. They are not computed in the backward process. Instead, we can pre-calculate them in the following forward process.

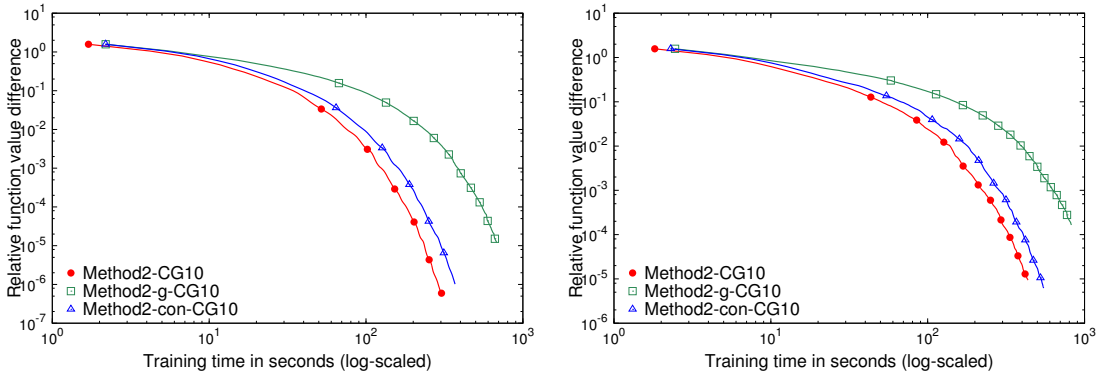
$$\begin{aligned}
\mathcal{R}\{x_i\} &= \mathcal{R}\{\sum_{j=1}^{n_{m-1}} w_{ji} \bar{z}_j\} = \sum_{j=1}^{n_{m-1}} (w_{ji} \mathcal{R}\{\bar{z}_j\} + v_{ji} \bar{z}_j) \\
\mathcal{R}\{z_i\} &= \mathcal{R}\{\sigma(x_i)\} = \mathcal{R}\{x_i\} \sigma'(x_i).
\end{aligned}$$

References

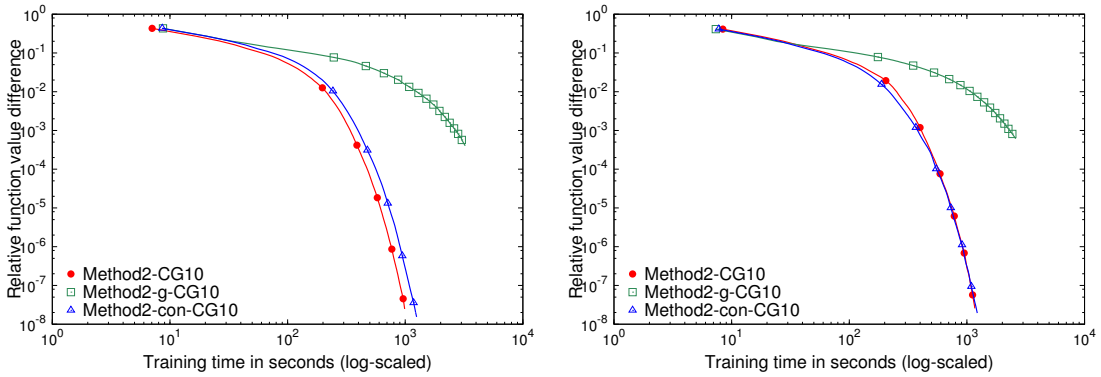
R. H. Byrd, G. M. Chin, W. Neveitt, and J. Nocedal. On the use of stochastic Hessian information in optimization methods for machine learning. *SIAM Journal on Optimization*, 21(3):977–995, 2011.



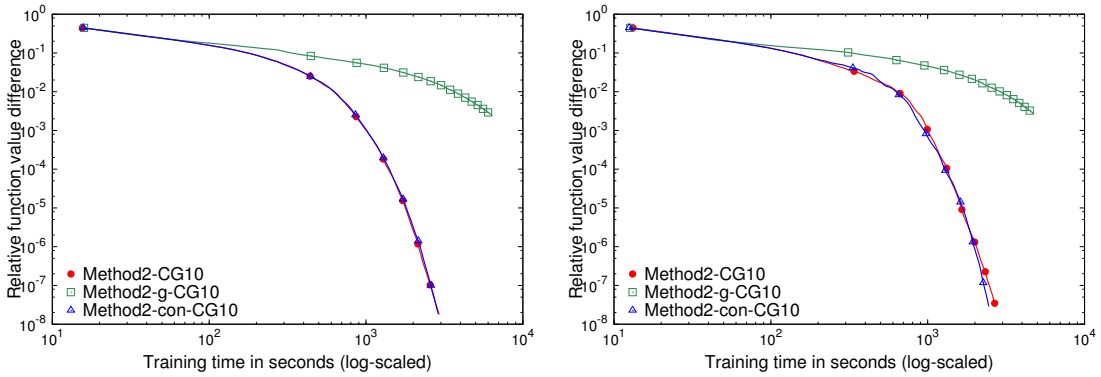
(a) news20



(b) yahoo-korea

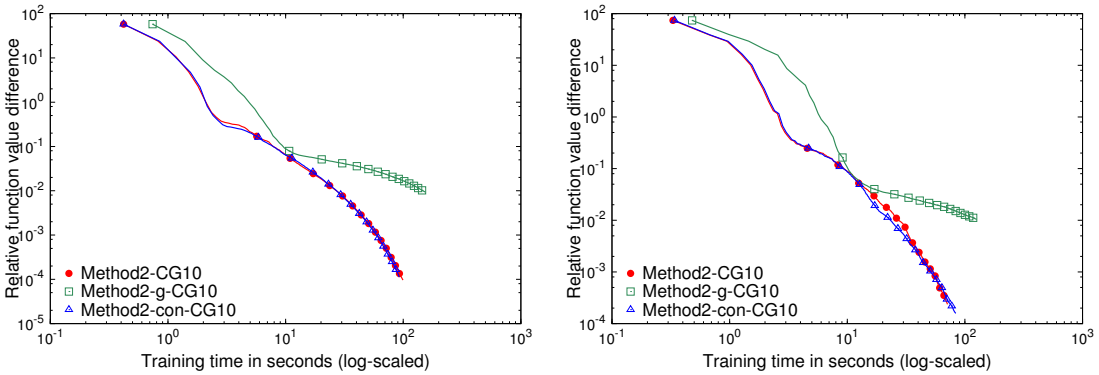


(c) kdd2010-a

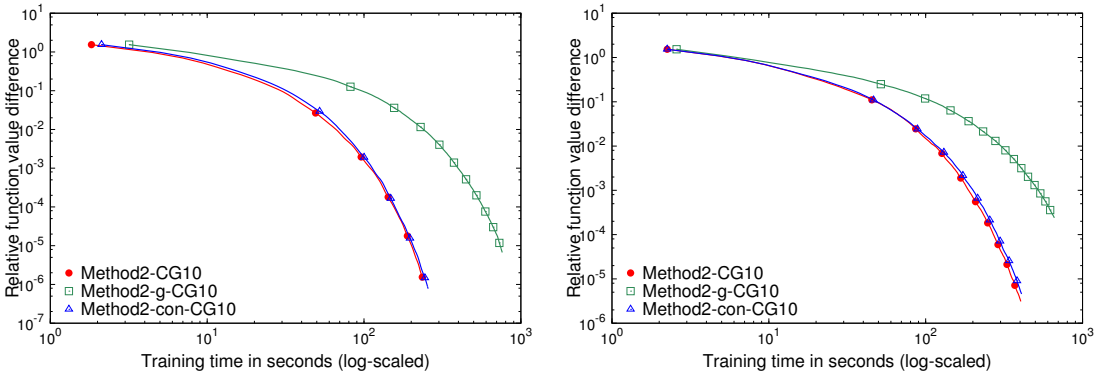


(d) kdd2010-b

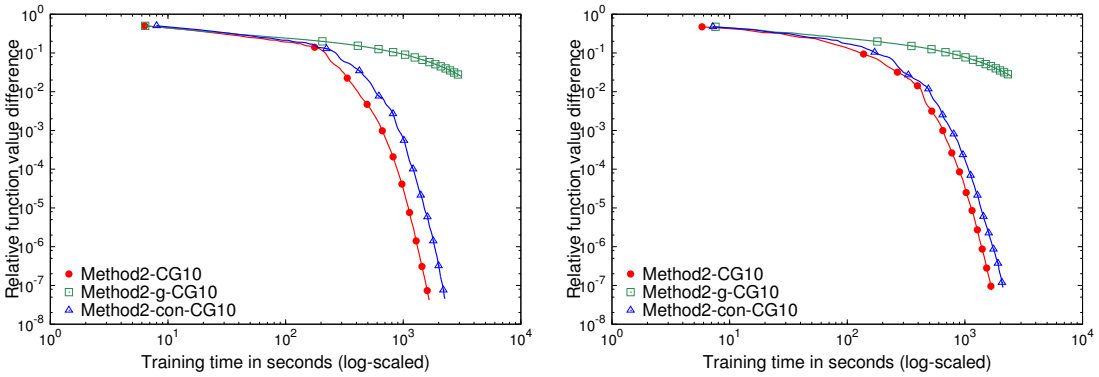
Figure III.2: Experiments on logistic regression using the three settings listed in Section III.3. We present running time (in seconds) versus the relative difference to the optimal function value. Both x-axis and y-axis are log-scaled. Left: $|S_k|/l = 5\%$. Right: $|S_k|/l = 1\%$.



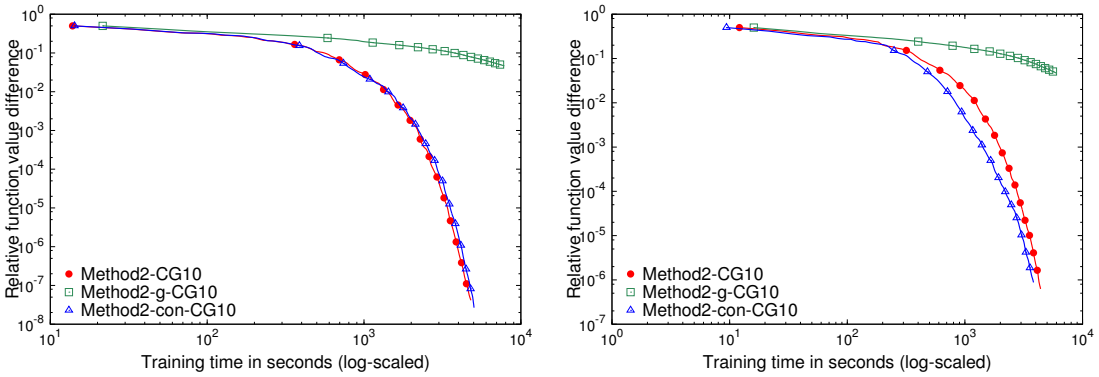
(a) news20



(b) yahoo-korea



(c) kdd2010-a



(d) kdd2010-b

Figure III.3: Experiments on l2-loss linear SVM using the three settings listed in Section III.3. We present running time (in seconds) versus the relative difference to the optimal function value. Both x-axis and y-axis are log-scaled. Left: $|S_k|/l = 5\%$. Right: $|S_k|/l = 1\%$.