Additional Materials for Two-variable Block Dual Coordinate Descent Methods for Large-scale Linear Support Vector Machines

Chi-Cheng Chiu* **Pin-Yen Lin*** **Chih-Jen Lin***

I. One-variable CD Method in Hsieh et al. (2008a)

A one-variable CD by Hsieh et al. (2008a) for linear SVM is in Algorithm I.

Algorithm I A one-variable CD by Hsieh et al. (2008a) for linear SVM

Input: Specify a feasible $\boldsymbol{\alpha}$
calculate $\boldsymbol{u} = \sum_j y_j \alpha_j \boldsymbol{x}_j$
while $\boldsymbol{\alpha}$ is not optimal **do**
  Obtain the permuted indices $\{\pi(1), \pi(2), \ldots, \pi(l)\}$
  for $j = 1, \ldots, l$ **do**
    $i \leftarrow \pi(j)$

$$G \leftarrow \begin{cases} y_i \boldsymbol{u}^T \boldsymbol{x}_i - 1 & l1 \text{ loss} \\ y_i \boldsymbol{u}^T \boldsymbol{x}_i - 1 + \frac{\alpha_i}{2C_i} & l2 \text{ loss} \end{cases}$$

    $d = \max(-\alpha_i, \min(C_i - \alpha_i, -G/Q_{ii}))$
    $\alpha_i \leftarrow \alpha_i + d$
    $\boldsymbol{u} \leftarrow \boldsymbol{u} + d y_i \boldsymbol{x}_i$
  end for
end while
$\boldsymbol{w} \leftarrow \boldsymbol{u}$
output: $(\boldsymbol{w}, \boldsymbol{\alpha})$ as approximate primal and dual solutions.

II. Details of Two-variable CD for Dual SVM without the Bias Term

II.I Solving Two-variable Sub-problems (3.17)

For easy understanding, we rewrite (3.17) to a more general two-variable optimization problem:

$$\begin{aligned} \min_{d_1, d_2} \quad & \frac{1}{2} \begin{bmatrix} d_1 & d_2 \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12} & Q_{22} \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} + \begin{bmatrix} p_1 & p_2 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \end{bmatrix} \\ \text{subject to} \quad & L_1 \le d_1 \le U_1, \quad L_1 \le d_2 \le U_2, \end{aligned} \tag{i}$$

∗. Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan

where $L_1,\ U_1,\ L_2,\ U_2 \in R$. The sub-problem (i) is the same as the one solved in Steinwart et al. (2011), which studies two-variable CD for kernel SVM. We briefly describe their solution procedure before ours. They begin with considering (i) without constraints. With (3.16), the solution is easily seen as

$$d_1^* = \frac{-Q_{22}p_1 + Q_{12}p_2}{Q_{11}Q_{22} - Q_{12}^2},\ d_2^* = \frac{-Q_{11}p_2 + Q_{12}p_1}{Q_{11}Q_{22} - Q_{12}^2}. \tag{ii}$$

Let the objective function of (i) be

$$\hat{f}(d_1, d_2).$$

If (d_1^*, d_2^*) is infeasible with

$$d_1^* > U_1 \text{ and } d_2^* \in [L_2, U_2],$$

an optimal solution must be on the line of $d_1 = U_1$. A conceptual proof is in Figure Ia: if a solution $\hat{\boldsymbol{d}}$ is not on this line, then the line segment connecting $\hat{\boldsymbol{d}}$ and $\boldsymbol{d}^*$ leads to a point on $d_1 = U_1$ with a smaller function value because of the strict convexity of the function $\hat{f}(d_1, d_2)$. Thus by fixing $d_1 = U_1$ one can solve a one-variable optimization problem to get the optimal solution $\bar{d}_2$. That is,

$$\bar{d}_1 = P[d_1^*], \quad \bar{d}_2 = \arg\min_{d_2 \in [L_2, U_2]} \hat{f}(\bar{d}_1, d_2),$$

where

$$P[d_i] = \min(U_i, \max(L_i, d_i)),\ \forall i = 1, 2,$$

is a projection operation. However, if

$$d_1^* > U_1,\ d_2^* > U_2, \tag{iii}$$

the above argument can only imply that the solution must be on either $d_1 = U_1$ or $d_2 = U_2$; see the illustration in Figure Ib. Thus Steinwart et al. (2011) proposes solving two one-dimensional problems where one is by fixing $\bar{d}_1 = U_1$ and the other is by fixing $\bar{d}_2 = U_2$. Then they compare two objective values to decide the solution.

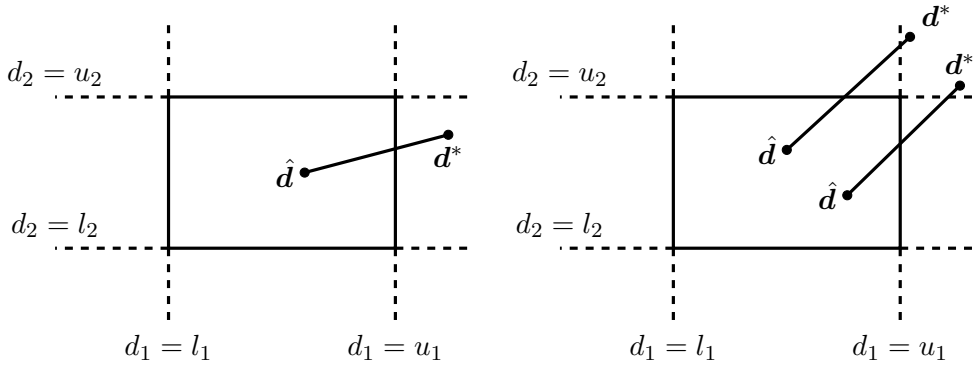
However, this implementation is slightly complicated. From Figure 1, eight out-of-boundary cases must be considered. Further for the situation in Figure 1b, it would be better if we solve one rather than two one-variable sub-problems.

To have a simple procedure, we notice that for the situation in Figure Ib, it is possible to use the gradient information for deciding which boundary line the optimal solution is at. Specifically, in Figure II we assume (iii) and have

$$(P[d_1^*], P[d_2^*]) = (U_1, U_2). \tag{iv}$$

We then consider two cases. For the first one in Figure IIa, the optimal solution of (i) is on the line

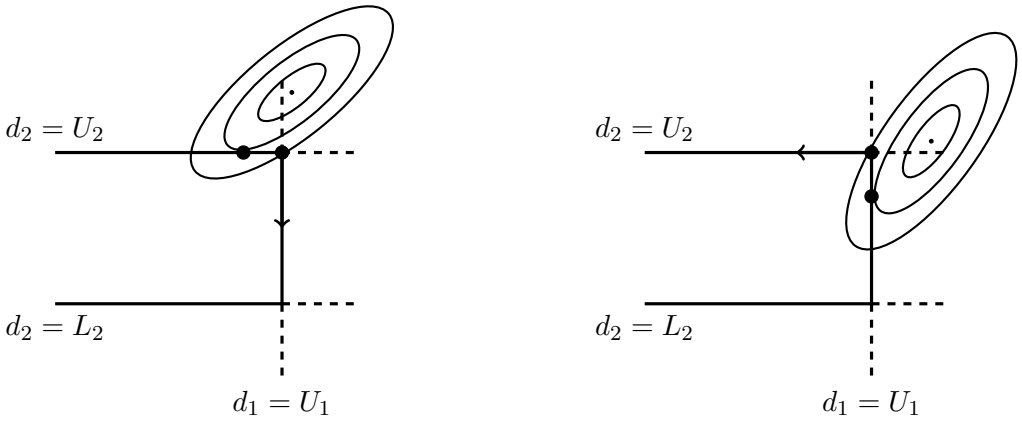
$$d_2 = U_2 \text{ and satisfies } \nabla_2 \hat{f}(P[d_1^*], P[d_2^*]) \leq 0,$$



(a) if $d_1^* \geq u_1$ and $d_2^* \in [l_2, u_2]$, then the solution is on $d_1 = u_1$.

(b) if $d_1^* \geq u_1$ and $d_2^* \geq u_2$, the solution is on either $d_1 = u_1$ or $d_2 = u_2$.

Figure I: illustrations of different situations of d^*, the solution without constraints.



(a) $\nabla_2 \hat{f}(P[d_1^*], P[d_2^*]) \leq 0$ (see the arrow in the figure) and the solution is on $d_2 = U_2$.

(b) $\nabla_1 \hat{f}(P[d_1^*], P[d_2^*]) \leq 0$ (see the arrow in the figure) and the solution is on $d_1 = U_1$.

Figure II: We can check the optimality condition at the point $(P[d_1^*], P[d_2^*]) = (U_1, U_2)$ to decide which line the optimal solution is at.

while for the second, it is on

$$d_1 = U_1 \text{ and satisfies } \nabla_1 \hat{f}(P[d_1^*], P[d_2^*]) \leq 0. \tag{v}$$

Let us look at the case of Figure IIb in detail. With (iv), the inequality in (v) means that on the line of $d_2 = U_2$, we must increase $P[d_1^*] = U_1$ to a larger value (i.e., the negative gradient direction) in order to decrease the function value. However, this is not possible because $P[d_1^*]$ is already at the upper bound. In other words, the optimality condition of d_1 has been satisfied. Therefore, the optimal solution must be on the line of $d_1 = U_1$. The case of Figure IIb can be formally extended to the following result.

Theorem II.1 *Assume $\bar{d}_1 = P[d_1^*]$ is bounded and $(P[d_1^*], P[d_2^*])$ satisfies the optimality condition at d_1; that is,*

$$Q_{11}P[d_1^*] + Q_{12}P[d_2^*] + p_1 \begin{cases} \leq 0 & \text{if } P[d_1^*] = U_1, \\ \geq 0 & \text{if } P[d_1^*] = L_1. \end{cases} \tag{vi}$$

Then $(\bar{d}_1, \bar{d}_2)$ *with*

$$\bar{d}_2 = \arg \min_{d_2 \in [L_2, U_2]} \frac{1}{2} \begin{bmatrix} \bar{d}_1 & d_2 \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12} & Q_{22} \end{bmatrix} \begin{bmatrix} \bar{d}_1 \\ d_2 \end{bmatrix} + \begin{bmatrix} p_1 & p_2 \end{bmatrix} \begin{bmatrix} \bar{d}_1 \\ d_2 \end{bmatrix}$$

$$= \min(U_2, \max(L_2, -\frac{Q_{12}\bar{d}_1 + p_2}{Q_{22}})) \tag{vii}$$

is an optimal solution of (i).

All proofs in this section are given in Section VI. The remaining task is to have a clever setting so that we do not need to separately handle the eight cases, where (d_1^*, d_2^*) is not in the feasible region.

While the strategy of checking (v) avoids solving two one-variable problems and comparing their objective values, it seems we still need to check all eight regions separately. Fortunately, we can handle Figure Ia and part of Figure Ib together because for the situation in Figure Ia, the following theorem shows that $(P[d_1^*], P[d_2^*])$ also satisfies the optimality condition of d_1.

Theorem II.2 *If*

$$d_1^* \notin (L_1, U_1),$$
$$d_2^* \in [L_2, U_2],$$

then $(P[d_1^*], P[d_2^*])$ *satisfies the optimality condition of* d_1.

Therefore, by Theorem II.1, we can cover a rather general situation by checking the optimality condition at $(P[d_1^*], P[d_2^*])$. Further, Theorem II.1 can hold if the roles of d_1^* and d_2^* are swapped. To ensure that every $\boldsymbol{d}^*$ in the situation of Figure Ib is covered (i.e., Theorem II.1 on either d_1^* or d_2^* is applicable), we need the following theorem.

Theorem II.3 *If*

$$d_1^* \notin (L_1, U_1),\ d_2^* \notin (L_2, U_2),$$

then $(P[d_1^*], P[d_2^*])$ *satisfies either the optimality condition of* d_1 *or* d_2.

Based on the above theorems we can derive a simple procedure for solving (i). To begin, if $d_1^* \notin (L_1, U_1)$ then we know that $P[d_1^*]$ is bounded. We may apply Theorem II.1 by checking if $(P[d_1^*], P[d_2^*])$ satisfies the optimality condition of d_1. If it does, then (vii) is an optimal solution.

There are two remaining situations:

$$d_1^* \in (L_1, U_1) \tag{viii}$$

or

$$d_1^* \notin (L_1, U_1) \text{ and } (P[d_1^*], P[d_2^*]) \text{ does not satisfy (vi).} \tag{ix}$$

For both situations, we argue that

$$\begin{aligned} \bar{d}_2 &= P[d_2^*], \\ \bar{d}_1 &= \min(U_1, \max(L_1, -\frac{Q_{12}\bar{d}_2 + p_1}{Q_{11}})) \end{aligned} \tag{x}$$

is an optimal solution. For (viii), we can further consider two situations.

$$d_2^* \in [L_2, U_2], \tag{xi}$$

$$d_2^* \notin [L_2, U_2]. \tag{xii}$$

If (xi) holds, then

$$P[d_1^*] = d_1^* \text{ and } P[d_2^*] = d_2^*$$

are already an optimal solution. Though we do not need to apply (x), if we do, then $\bar{d}_1 = d_1^*$ is obtained. On the other hand, if (xii) holds, then from Theorem II.2, $(P[d_1^*], P[d_2^*])$ satisfies the optimality condition of d_2. With the boundedness of $P[d_2^*]$, we can apply Theorem II.1 to have (x).

For the situation of (ix), we argue that $d_2^* \notin [L_2, U_2]$. Otherwise, $d_2^* \in [L_2, U_2]$ and $d_1^* \notin (L_1, U_1)$ imply from Theorem II.2 that $(P[d_1^*], P[d_2^*])$ satisfies the optimality condition of d_1, a contradiction to the condition in (ix). Next, the property $d_2^* \notin [L_2, U_2]$, (ix) and Theorem II.3 imply that $(P[d_1^*], P[d_2^*])$ must satisfy the optimality condition of d_2.

A summary of the procedure is in Algorithm II, in which we switch back to α_i, α_j from d_1, d_2 for practical implementations. Besides, p_1 and p_2 are changed back to $\nabla_i f(\boldsymbol{\alpha})$ and $\nabla_j f(\boldsymbol{\alpha})$, respectively. Clearly, by using the gradient information rather than comparing objective values, the procedure becomes simple and short. Note that before applying the procedure discussed in this section, we should check if (α_i, α_j) is already optimal for the sub-problem (3.17). See details in Algorithm III described later in Section II.III

II.II Proof of Linear Convergence

We prove the linear convergence of the two-variable CD by using (3.20) for the working-set selection.

The work Wang and Lin (2014) considers two classes of problems (see their Assumptions 2.1 and 2.2) from the following convex optimization problem

$$\min_{\boldsymbol{\alpha} \in \mathcal{X}} f(\boldsymbol{\alpha}), \tag{xiii}$$

where $f(\boldsymbol{\alpha})$ is proper convex, and $\mathcal{X}$ is nonempty, closed, and convex. It is shown in Section 3.1 of Wang and Lin (2014) that the dual problem of both $l1$-loss SVM and $l2$-loss SVM are within the problems considered by them.[1] They then analyzes feasible-descent algorithms,

1. Note that for $l1$-loss SVM, they point out that some zero data instances must be removed first. This can be easily handled before solving the optimization problem.

where at the kth iteration the current and the next iterates satisfy

$$\boldsymbol{\alpha}^{k+1} = \left[\boldsymbol{\alpha}^k - \omega_k \nabla f\left(\boldsymbol{\alpha}^k\right) + \boldsymbol{e}^k\right]_{\mathcal{X}}^{+}, \tag{xiv}$$

$$\left\|\boldsymbol{e}^k\right\| \leq \beta \left\|\boldsymbol{\alpha}^k - \boldsymbol{\alpha}^{k+1}\right\|, \tag{xv}$$

$$f\left(\boldsymbol{\alpha}^k\right) - f\left(\boldsymbol{\alpha}^{k+1}\right) \geq \gamma \left\|\boldsymbol{\alpha}^k - \boldsymbol{\alpha}^{k+1}\right\|^2, \tag{xvi}$$

where $\inf_r \omega_r > 0,\ \beta > 0, \gamma > 0$, and $[\cdot]_{\mathcal{X}}^{+}$ is the following convex projection operator to the set $\mathcal{X}$:

$$[\boldsymbol{x}]_{\mathcal{X}}^{+} = \arg\min_{\boldsymbol{y}\in\mathcal{X}} \|\boldsymbol{x} - \boldsymbol{y}\|. \tag{xvii}$$

For dual SVM,

$$\mathcal{X} = [0, C_1] \times \cdots \times [0, C_l].$$

From (xvii),

$$[\boldsymbol{\alpha}]_{\mathcal{X}}^{+} = \left[\max(\min(\alpha_1, C_1), 0), \ldots, \max(\min(\alpha_l, C_l), 0)\right]^T.$$

Based on Theorem 2.8 of Wang and Lin (2014), we can prove the following linear-convergence result.

Theorem II.4 *The two-variable CD for dual l1-loss and l2-loss SVM has global linear convergence. To be specific, the method converges Q-linearly with*

$$f\left(\boldsymbol{\alpha}^{k+1}\right) - f^* \leq \frac{\phi}{\phi + \gamma}\left(f\left(\boldsymbol{\alpha}^k\right) - f^*\right), \quad \forall k \geq 0,$$

where κ is the error bound constant,

$$\phi = \left(\rho + \frac{1+\beta}{\omega}\right)\left(1 + \kappa\frac{1+\beta}{\omega}\right),$$

$$\textit{and} \quad \omega \equiv \min\left(1, \inf_k \omega_k\right).$$

For l1-loss SVM, κ is derived in (7) of Wang and Lin (2014), and for l2-loss SVM,

$$\kappa = 2(1+\rho)\max_{i=1\ldots,l} C_i, \tag{xviii}$$

where $\rho = \lambda_{max}(Q)$, the largest eigenvalue of Q, is the Lipschitz constant of $\nabla f(\boldsymbol{\alpha})$.

Proof

To begin, we show that two-variable CD is a special case of the feasible-descent algorithms. The three conditions (xiv)-(xvi) are satisfied with

$$\omega_k = 1, \quad \beta = 1 - \lambda + \sqrt{l}\rho, \quad \gamma = \frac{\lambda}{2},$$

where λ is the proximal term parameter in (3.14).

We consider one iteration to be the collection of CD steps to go over all variables. From (3.20), we let

$$B_1 = (\pi(1), \pi(2)), \ldots, B_{\bar{l}} = (\pi(l-1), \pi(l))$$

be the working sets considered in one iteration. Let

$$\boldsymbol{\alpha}^{k+1,1}, \boldsymbol{\alpha}^{k+1,2}, \ldots, \boldsymbol{\alpha}^{k+1,\bar{l}} = \boldsymbol{\alpha}^{k+1}$$

be solutions updated after each CD step, and we consider

$$\boldsymbol{\alpha}^1 = \boldsymbol{\alpha}^{1,1} = \boldsymbol{\alpha}^{1,2} = \cdots = \boldsymbol{\alpha}^{1,\bar{l}}.$$

Because $\boldsymbol{\alpha}^{k,\bar{i}}_{B_{\bar{i}}}$ is not changed before we obtain $\boldsymbol{\alpha}^{k+1,\bar{i}}$, $\boldsymbol{d}_B$ in (2.5) corresponds to $\boldsymbol{\alpha}^{k+1,\bar{i}}_{B_{\bar{i}}} - \boldsymbol{\alpha}^{k,\bar{i}}_{B_{\bar{i}}}$. From the optimality condition of the sub-problem (3.14),[2] we have for all $\bar{i} = 1, \ldots, \bar{l}$,

$$\boldsymbol{\alpha}^{k+1,\bar{i}}_{B_{\bar{i}}} = \left[\boldsymbol{\alpha}^{k+1,\bar{i}}_{B_{\bar{i}}} - \nabla_{B_{\bar{i}}} f(\boldsymbol{\alpha}^{k+1,\bar{i}}) - \lambda(\boldsymbol{\alpha}^{k+1,\bar{i}}_{B_{\bar{i}}} - \boldsymbol{\alpha}^{k,\bar{i}}_{B_{\bar{i}}})\right]^+_{\mathcal{X}}. \tag{xix}$$

With

$$\boldsymbol{\alpha}^{k}_{B_{\bar{i}}} = \boldsymbol{\alpha}^{k,\bar{i}}_{B_{\bar{i}}} \text{ and } \boldsymbol{\alpha}^{k+1}_{B_{\bar{i}}} = \boldsymbol{\alpha}^{k+1,\bar{i}}_{B_{\bar{i}}},\ \forall \bar{i} = 1, \ldots, \bar{l}, \tag{xx}$$

we can rewrite (xix) as

$$\boldsymbol{\alpha}^{k+1}_{B_{\bar{i}}} = \left[\boldsymbol{\alpha}^{k+1}_{B_{\bar{i}}} - \nabla_{B_{\bar{i}}} f(\boldsymbol{\alpha}^{k+1,\bar{i}}) - \lambda(\boldsymbol{\alpha}^{k+1}_{B_{\bar{i}}} - \boldsymbol{\alpha}^{k}_{B_{\bar{i}}})\right]^+_{\mathcal{X}}. \tag{xxi}$$

Next, we let

$$\boldsymbol{\alpha}^{k+1} = \left[\boldsymbol{\alpha}^k - \nabla f(\boldsymbol{\alpha}^k) + \boldsymbol{e}^k\right]^+,$$

where from (xxi)

$$\begin{aligned} e^k_i &= \alpha^{k+1}_i - \alpha^k_i + \nabla_i f(\boldsymbol{\alpha}^k) - \nabla_i f(\boldsymbol{\alpha}^{k+1,\bar{i}}) \\ &\quad - \lambda(\alpha^{k+1}_i - \alpha^k_i) \\ &= (1-\lambda)(\alpha^{k+1}_i - \alpha^k_i) + \nabla_i f(\boldsymbol{\alpha}^k) - \nabla_i f(\boldsymbol{\alpha}^{k+1,\bar{i}}), \\ &\quad \forall\ \bar{i} = 1, \ldots, \bar{l} \text{ and } i \in B_{\bar{i}}. \end{aligned}$$

Thus the condition (xiv) holds. Then from the Lipschitz continuity, for all $\bar{i} = 1, \ldots, \bar{l}$ and $i \in B_{\bar{i}}$, we have

$$\begin{aligned} |e^k_i| &\le (1-\lambda)|\alpha^{k+1}_i - \alpha^k_i| + |\nabla_i f(\boldsymbol{\alpha}^k) - \nabla_i f(\boldsymbol{\alpha}^{k+1,\bar{i}})| \\ &= (1-\lambda)|\alpha^{k+1}_i - \alpha^k_i| + |\nabla_i f(\boldsymbol{\alpha}^k) - \nabla_i f(\boldsymbol{\alpha}^{k+1})| \\ &\le (1-\lambda)|\alpha^{k+1}_i - \alpha^k_i| + \rho\|\boldsymbol{\alpha}^{k+1} - \boldsymbol{\alpha}^k\|, \end{aligned} \tag{xxii}$$

2. See also (xxix) and (xxx).

where (xxii) is from (xx). By summing up all the $|e_i^k|^2$, we can get

$$\begin{aligned}\|\boldsymbol{e}^k\|^2 &\le \sum_{i=1}^{l}((1-\lambda)^2|\alpha_i^{k+1}-\alpha_i^k|^2+\rho^2\|\boldsymbol{\alpha}^{k+1}-\boldsymbol{\alpha}^k\|^2\\ &\quad+2(1-\lambda)\rho|\alpha_i^{k+1}-\alpha_i^k|\|\boldsymbol{\alpha}^{k+1}-\boldsymbol{\alpha}^k\|\\ &=\left((1-\lambda)^2+l\rho^2)\right)\|\boldsymbol{\alpha}^{k+1}-\boldsymbol{\alpha}^k\|^2\\ &\quad+2(1-\lambda)\rho\|\boldsymbol{\alpha}^{k+1}-\boldsymbol{\alpha}^l\|_1\|\boldsymbol{\alpha}^{k+1}-\boldsymbol{\alpha}^l\|\\ &\le\left((1-\lambda)^2+l\rho^2+2\sqrt{l}(1-\lambda)\rho)\right)\|\boldsymbol{\alpha}^{k+1}-\boldsymbol{\alpha}^k\|^2\\ &=\left((1-\lambda+\sqrt{l}\rho)\|\boldsymbol{\alpha}^{k+1}-\boldsymbol{\alpha}^k\|\right)^2\end{aligned}$$

and the condition (xiv) is satisfied as follows.

$$\|\boldsymbol{e}^k\| \le (1-\lambda+\sqrt{l}\rho)\|\boldsymbol{\alpha}^{k+1}-\boldsymbol{\alpha}^k\|.$$

From (3.14) and (xx),

$$f\left(\boldsymbol{\alpha}^{k+1,i-1}\right)+\frac{\lambda}{2}\|\boldsymbol{\alpha}_{B_i}^{k+1}-\boldsymbol{\alpha}_{B_i}^k\|^2 \le f\left(\boldsymbol{\alpha}^{k+1,i}\right) \quad i=1,\ldots,\bar{l}, \tag{xxiii}$$

where we let $\boldsymbol{\alpha}^{k+1,0}$ be $\boldsymbol{\alpha}^k$. The summation of inequalities in (xxiii) leads to

$$f(\boldsymbol{\alpha}^k)-f(\boldsymbol{\alpha}^{k+1}) \ge \frac{\lambda}{2}\|\boldsymbol{\alpha}^k-\boldsymbol{\alpha}^{k+1}\|^2,$$

which is the condition (xvi). Therefore, two-variable CD is a special case of the feasible-descent algorithm in Wang and Lin (2014), so we can use their results to have the linear convergence.

Next we derive the κ value in (xviii) for $l2$-loss SVM. From Wang and Lin (2014), the $l2$ loss satisfies their Assumption 2.1, and therefore κ can be chosen as

$$\kappa = \frac{1+\rho}{\sigma},$$

where ρ is the Lipschitz constant of $\nabla f(\boldsymbol{\alpha})$, and $f(\boldsymbol{\alpha})$ is σ strongly convex. For SVM we have

$$\|\nabla f(\boldsymbol{\alpha}_1)-\nabla f(\boldsymbol{\alpha}_2))\| = \|Q(\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_2)\| \le \lambda_{\max}\|\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_2\|,$$

where $\rho=\lambda_{\max}$ can be the Lipschitz constant. For the σ value, from (2.4),

$$(\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_2)^T Q(\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_2) \ge \min_{i=1,\ldots,l}\left(\frac{1}{2C_i}\right)\|\boldsymbol{\alpha}_1-\boldsymbol{\alpha}_2\|^2.$$

Thus,

$$\kappa = \frac{1+\rho}{\sigma} = 2(1+\lambda_{\max})\max_{i=1,\ldots,l} C_i.$$

■

II.III Shrinking Technique

Because of bound constraints $0 \leq \alpha_i \leq C_i$, it is well developed in SVM literature that some bounded components can be tentatively removed in the optimization process. Then we solve smaller problems to reduce the running time, a strategy usually referred to as the shrinking technique Joachims (1998). Though several ways are available to implement the shrinking technique, we extend the one proposed by Hsieh et al. (2008a) to the two-variable situation. For a bound-constrained convex problem like (2.3), $\boldsymbol{\alpha}$ is optimal if and only if the following projected gradient is zero.

$$\nabla_i^P f(\boldsymbol{\alpha}) = \begin{cases} \nabla_i f(\boldsymbol{\alpha}) & \text{if } 0 < \alpha_i < C_i, \\ \min(0, \nabla_i f(\boldsymbol{\alpha})) & \text{if } \alpha_i = 0, \\ \max(0, \nabla_i f(\boldsymbol{\alpha})) & \text{if } \alpha_i = C_i. \end{cases}$$

For the one-variable CD, let each cycle of updating all the remained variables be an "outer iteration." Assume at the $(k-1)$th outer iteration we have the following sequence of iterates.

$$\boldsymbol{\alpha}^{k-1,1}, \boldsymbol{\alpha}^{k-1,2}, \ldots, \boldsymbol{\alpha}^{k-1,\bar{l}},$$

where $\bar{l}$ is the number of remained variables at the beginning of the outer iteration. We further assume that at $\boldsymbol{\alpha}^{k-1,j}$, the index i_j is selected for possible update. The work Hsieh et al. (2008a) defines the following two values to indicate the violation of the optimality condition.

$$M^{k-1} \equiv \max_j \nabla_{i_j}^P f(\boldsymbol{\alpha}^{k-1,j}), \; m^{k-1} \equiv \min_j \nabla_{i_j}^P f(\boldsymbol{\alpha}^{k-1,j}).$$

Then at each CD step of the next (i.e., the kth) outer iteration, before updating $\alpha_{i_j}^{k,j}$ to $\alpha_{i_j}^{k,j+1}$, the variable α_{i_j} is shrunken if one of the following two conditions holds:

$$\begin{aligned} \alpha_{i_j}^{k,j} = 0 \text{ and } \nabla_{i_j} f(\boldsymbol{\alpha}^{k,j}) > \bar{M}^{k-1}, \\ \alpha_{i_j}^{k,j} = C_i \text{ and } \nabla_{i_j} f(\boldsymbol{\alpha}^{k,j}) < \bar{m}^{k-1}, \end{aligned} \tag{xxiv}$$

where

$$\bar{M}^{k-1} = \begin{cases} M^{k-1} & \text{if } M^{k-1} > 0, \\ \infty & \text{otherwise,} \end{cases}$$
$$\bar{m}^{k-1} = \begin{cases} m^{k-1} & \text{if } m^{k-1} < 0, \\ -\infty & \text{otherwise.} \end{cases}$$

In (xxiv), $\bar{M}^{k-1}$ must be strictly positive, so Hsieh et al. (2008a) set it to be ∞ if $\bar{M}^{k-1} \leq 0$. The situation for $\bar{m}_{k-1}$ is similar. Details of one-variable CD with shrinking can be found in appendix of Hsieh et al. (2008a). The extension of the above setting to two-variable block CD is straightforward because we can consider steps of going through all pairs in (3.20) as an outer iteration for calculating M^{k-1} and m^{k-1}.

A summary of the two-variable CD with a shrinking implementation is in Algorithm III.

III. Additional Discussion on Two-variable CD Methods for Linear SVM with a Bias Term

III.I Solving the Sub-problem: Difference from SVM Without the Bias Term

Interestingly, though it is easy to derive a solution procedure for solving (4.21), a comparison shows that Algorithm II of supplementary materials for solving (i) is shorter in terms of the code length. One reason is that in Algorithm II, gradient information (or optimality condition) is used to avoid the exhaustive check of all out-of-boundary cases of α_i or α_j. Further, for solving (4.21), we must separately handle the situations of $y_i = y_j$ and $y_i = -y_j$.

III.II Difference Between Linear and Kernel Situations

We point out a difference in solving (4.21) between linear and kernel situations. For the kernel situation, as mentioned in Section 2.3, Chang and Lin (2011) considers a greedy working set selection by using the gradient information, so their selected set satisfies

$$-y_i y_j \nabla_i f(\boldsymbol{\alpha}) + \nabla_j f(\boldsymbol{\alpha}) \neq 0. \tag{xxv}$$

If

$$Q_{ii} - 2y_i y_j Q_{ij} + Q_{jj} = 0, \tag{xxvi}$$

the following situation in minimizing the quadratic objective function of (4.21) occurs.

$$\frac{-y_i y_j \nabla_i f(\boldsymbol{\alpha}) + \nabla_j f(\boldsymbol{\alpha})}{Q_{ii} - 2y_i y_j Q_{ij} + Q_{jj}} = \infty \text{ or } -\infty. \tag{xxvii}$$

By (xxv), this can be easily handled under the IEEE floating-point standard. However, for linear SVM, because of a random or a cyclic selection, (xxv) does not hold and $0/0$ may occur. It can be easily seen that if

$$-y_i y_j \nabla_i f(\boldsymbol{\alpha}) + \nabla_j f(\boldsymbol{\alpha}) = 0,$$

then the minimum of (4.21) is attained with

$$d_j = 0.$$

Therefore, the selected pair is not useful to reduce the function value. We can conduct a simple check on (xxv) before solving the two-variable sub-problem.

Algorithm II A procedure to solve the two-variable sub-problem (3.17). Note that for practical implementations we switch back to use α_i, α_j rather than d_1, d_2.

Let $p_i \leftarrow \nabla_i f(\boldsymbol{\alpha}),\ p_j \leftarrow \nabla_j f(\boldsymbol{\alpha})$.

Let

$$\delta \leftarrow Q_{ii}Q_{jj} - Q_{ij}^2$$
$$\text{use\_j} \leftarrow \text{FALSE}$$

calculate

$$\bar{\alpha}_i \leftarrow \min(C_i, \max(0, \alpha_i + \frac{-Q_{jj}p_i + Q_{ij}p_j}{\delta}))$$
$$\bar{\alpha}_j \leftarrow \min(C_j, \max(0, \alpha_j + \frac{-Q_{ii}p_j + Q_{ij}p_i}{\delta}))$$

if $\bar{\alpha}_i \geq C_i$ **then**
  if $Q_{ii}(\bar{\alpha}_i - \alpha_i) + Q_{ij}(\bar{\alpha}_j - \alpha_j) + p_i \leq 0$ **then**

$$\bar{\alpha}_j \leftarrow \min(C_j, \max(0, \alpha_j - \frac{Q_{ij}(\bar{\alpha}_i - \alpha_i) + p_j}{Q_{jj}})) \tag{xxviii}$$

  else
    use_j $\leftarrow$ TRUE
  end if
else if $\bar{\alpha}_i \leq 0$ **then**
  if $Q_{ii}(\bar{\alpha}_i - \alpha_i) + Q_{ij}(\bar{\alpha}_j - \alpha_j) + p_i \geq 0$ **then**

$$\bar{\alpha}_j \leftarrow \min(C_j, \max(0, \alpha_j - \frac{Q_{ij}(\bar{\alpha}_i - \alpha_i) + p_j}{Q_{jj}}))$$

  else
    use_j $\leftarrow$ TRUE
  end if
else
  use_j $\leftarrow$ TRUE
end if
if use_j = TRUE

$$\bar{\alpha}_i \leftarrow \min(C_i, \max(0, \alpha_i - \frac{Q_{ij}(\bar{\alpha}_j - \alpha_j) + p_i}{Q_{ii}}))$$

end if

Algorithm III Two-variable block CD for solving (2.3) with a shrinking implementation.

Given $\epsilon, \boldsymbol{\alpha}$ and the corresponding $\boldsymbol{w} = \sum_i y_i \alpha_i \boldsymbol{x}_i$.
Remove indices with $\boldsymbol{x}_i = \mathbf{0}$
Let $\bar{M} \leftarrow \infty, \bar{m} \leftarrow -\infty$ and $A \leftarrow \{1, \ldots, l\}$.
while $\boldsymbol{\alpha}$ is not optimal **do**
  Let $M \leftarrow -\infty, m \leftarrow \infty$.
  for all paris in (3.20) **do**
    Let $\{i, j\}$ be the current pair as the working set.
    $G_i = y_i \boldsymbol{w}^T \boldsymbol{x}_i - 1 + D_{ii}\alpha_i$, $G_j = y_j \boldsymbol{w}^T \boldsymbol{x}_j - 1 + D_{jj}\alpha_j$
    $Q_{ij} = y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j$
    $PG_i \leftarrow 0$, $PG_j \leftarrow 0$
    for $t = i, j$ **do**
      if $\alpha_t = 0$ **then**
        if $G_t > \bar{M}$ **then** $A \leftarrow A \setminus \{t\}$, $PG_i \leftarrow 0, PG_i \leftarrow 0$ and **break**
        if $G_t < 0$ **then** $PG_t \leftarrow G_t$
      else if $\alpha_t = C_t$ **then**
        if $G_t < \bar{m}$ **then** $A \leftarrow A \setminus \{t\}$ $PG_i \leftarrow 0, PG_i \leftarrow 0$ and **break**
        if $G_t > 0$ **then** $PG_t \leftarrow G_t$
      else
        $PG_t \leftarrow G_t$
      end if
      $M \leftarrow \max(M, PG_t), m \leftarrow \min(m, PG_t)$
    end for
    $(\bar{\alpha}_i, \bar{\alpha}_j) \leftarrow (\alpha_i, \alpha_j)$
    if $PG_i \neq 0$ **or** $PG_j \neq 0$ **then**
      $(\bar{\alpha}_i, \bar{\alpha}_j) \leftarrow$ Solve (i) by Algorithm II
    end if
    for $t = i, j$ **do**
      if $\bar{\alpha}_t \neq \alpha_t$ **then**
        $\boldsymbol{w} \leftarrow \boldsymbol{w} + (\bar{\alpha}_t - \alpha_t) y_t \boldsymbol{x}_t$
        $\alpha_t \leftarrow \bar{\alpha}_t$
      end if
    end for
  end for
  if $M - m < \epsilon$ **then**
    if $A = \{1, \ldots, l\}$ **then**
      break
    else
      $A \leftarrow \{1, \ldots, l\}, \bar{M} \leftarrow \infty, \bar{m} \leftarrow -\infty$. (i.e., no shrinking at the next iteration)
    end if
  end if
  if $M \leq 0$ **then** $\bar{M} \leftarrow \infty$ **else** $\bar{M} \leftarrow M$
  if $m \geq 0$ **then** $\bar{m} \leftarrow -\infty$ **else** $\bar{m} \leftarrow m$
end while

Algorithm IV Solve the sub-problem (3.17) without considering the proximal term.

1: Let $p_i \leftarrow \nabla_i f(\boldsymbol{\alpha}),\ p_j \leftarrow \nabla_j f(\boldsymbol{\alpha})$.
2: Let

$$\delta \leftarrow Q_{ii}Q_{jj} - Q_{ij}^2$$
use_j $\leftarrow$ FALSE

3: **if** $\delta = 0$ **and** $(-Q_{jj}p_i + Q_{ij}p_j = 0$ **or** $-Q_{ii}p_j + Q_{ij}p_i = 0)$ **then**
4: // (lxx) and (lxxi) both occur
5: $\bar{\delta} \leftarrow Q_{ii}\alpha_i + Q_{ij}\alpha_j - p_i$
6: **if** $Q_{ij} < 0$ **then**
7: **if** $Q_{ii}C_i \leq \bar{\delta}$ **then**
8: **return** $(C_i, 0)$
9: **else if** $Q_{ij}C_j \geq \bar{\delta}$ **then**
10: **return** $(0, C_j)$
11: **else if** $0 \leq \bar{\delta}$ **then**
12: **return** $(\frac{\bar{\delta}}{Q_{ii}}, 0)$
13: **else**
14: **return** $(0, \frac{\bar{\delta}}{Q_{ij}})$
15: **end if**
16: **else if** $Q_{ij} > 0$ **then**
17: **if** $0 \geq \bar{\delta}$ **then**
18: **return** $(0, 0)$
19: **else if** $Q_{ii}C_i + Q_{ij}C_j \leq \bar{\delta}$ **then**
20: **return** (C_i, C_j)
21: **else if** $Q_{ij}C_j \leq \bar{\delta}$ **then**
22: **return** $(\frac{\bar{\delta} - Q_{ij}C_j}{Q_{ii}}, C_j)$
23: **else**
24: **return** $(0, \frac{\bar{\delta}}{Q_{ij}})$
25: **end if**
26: **end if**
27: **else**
28: run line 3-18 in Algorithm II
29: **end if**

IV. Details of Experimental Settings

Our implementation is extended from the software LIBLINEAR Fan et al. (2008), which provides an implementation of the one-variable CD by Hsieh et al. (2008b). Except in Section V.II, we do not incorporate the shrinking technique. All experiments are conducted on a computer with an AMD EPYC 7401 24-Core Processor.

IV.I Datasets

For experiments we consider data sets listed in Table I. All sets except `yahoojp` and `yahookr` are publicly available at `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`

V. Complete Experimental Results on $l1$- and $l2$-loss SVM

To begin, Figure III presents results discussed in Section 5.1 for comparing working-set selections.

In the rest of this section we present complete results of using $l1$ and $l2$ losses by including two more data sets, `covtype.scale` and `yahookr`. The experimental settings are the same as in the main paper.

V.I Linear SVM without Bias: Comparison between One-variable and Two-variable CD

We give complete results of experiments conducted in Section 5.2. For $l2$-loss SVM, results with two more sets are presented; see Figures IV and V. For results of $l1$-loss SVM, which are not presented in the main paper, see Figures VI and VII.

V.II Effect of Shrinking Techniques for Two-variable CD for the Dual of Linear SVM Without the Bias Term

To check the effect of the shrinking technique in two-variable CD, we compare the following settings.

- 1-CD: this is the same as 1-CD-perm in Section 5.2.
- 1-CD-shrinking: shrinking technique is incorporated into 1-CD.
- 2-CD: this is the same as 2-CD-random in Section 5.2.
- 2-CD-shrinking: shrinking technique is incorporated into 2-CD.

Table I: Data statistics.

data set	#data	#features	data set	#data	#features
`ijcnn1`	49,990	22	`a9a`	32,561	123
`news20.binary`	19,996	1,355,191	`rcv1_train.binary`	20,242	47,326
`real-sim`	72,309	20,958	`yahoojp`	176,203	832,026
`yahookr`	460,554	156,436,656	`covtype.binary`	581,012	54

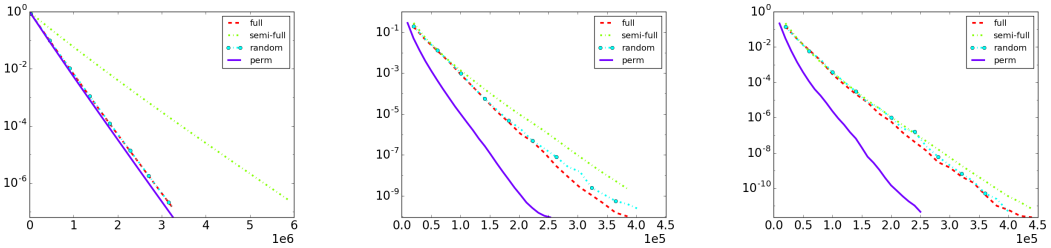
For $l2$-loss SVM, in Figure VIII we present a timing comparison between with and without shrinking by using $C = 1$ and $8,192$. For $l1$-loss SVM, results are presented in Figure IX. From the figures we see that shrinking for two-variable CD is generally as effective as for one-variable CD.

V.III Additional Results to Compare Linear SVM with/without the Bias Term

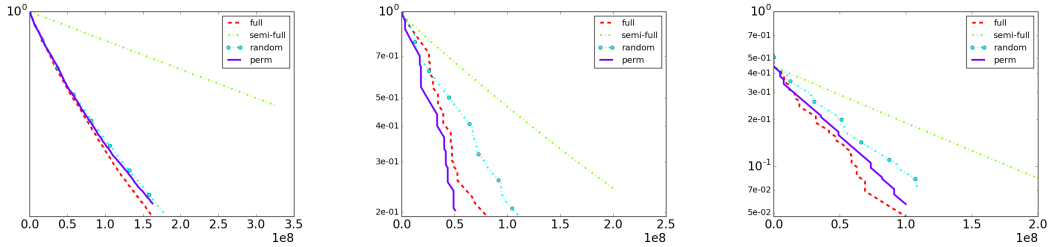
We begin with giving complete results of experiments in Section 5.3 to compare two-variable CD on linear SVM with and without the bias term. Results for $l2$- and $l1$-loss SVMs are respectively in Figure X and Figure XI.

Recall in (2.2) we mentioned a well-known trick to embed the bias term into the model vector $\boldsymbol{w}$. The dual problem then does not have a linear constraint. To see if the same argument in Section 4 holds, we compare two-variable CD for this setting with the one that explicitly handle the bias (i.e., the dual has a linear constraint). Results in Figure XII show that the gap between the two approaches is smaller than that in Figure 5. However, the two-variable CD for the setting in (2.2) is still generally faster because of no linear constraint in the dual.

Next we aim to check the test accuracy. In Figure 5 we have shown the dramatic difference on the function-value reduction when two-variable CD is applied to dual linear SVM with/without a linear constraint. We further check if the same observation holds on the test accuracy. We split each data set to 80% for training and 20% for testing and use cross-validation to find their C parameters. In Figures XIII-XIV we present test accuracy versus the cumulative number of CD steps. In each figure test accuracy of linear SVM with/without the linear constraint is compared, though in Figure XIV the approach without the linear constraint applies the trick in (2.2) to embed the bias term in the model. Results indicate that two-variable CD for linear SVM with the bias term is still slower.



$c = 1$



$c = 8, 192$

Figure III: A comparison of strategies for selecting the two-variable working set for $l2$-loss SVM. The x-axis is the number of CD steps, while the y-axis (log-scaled) is the relative difference to the optimal function value. Only small sets are used because of the $O(l^2)$ storage requirement of full.

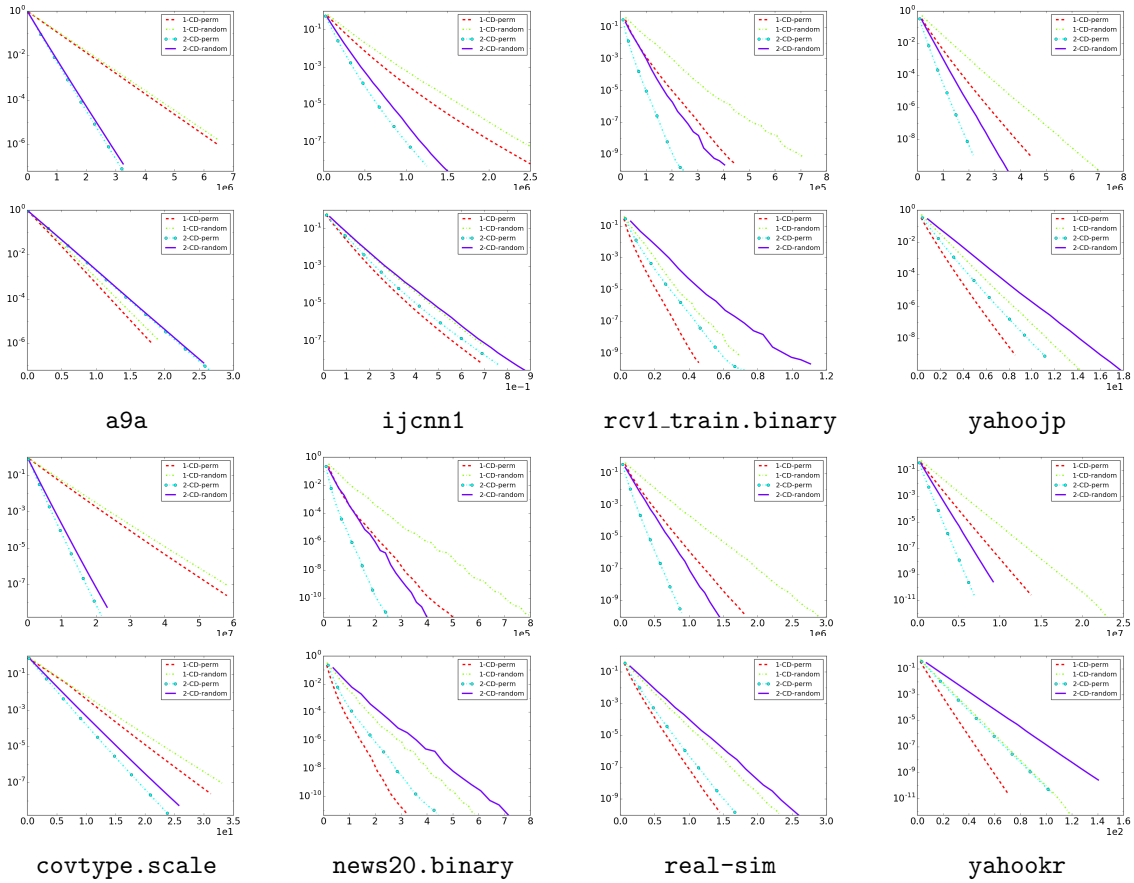


Figure IV: A comparison between one-variable and two-variable CD for $l2$ loss with $C = 1$. For each set, x-axis in the upper sub-figure is the number of CD steps, while x-axis in the lower sub-figure is the running time (in seconds).

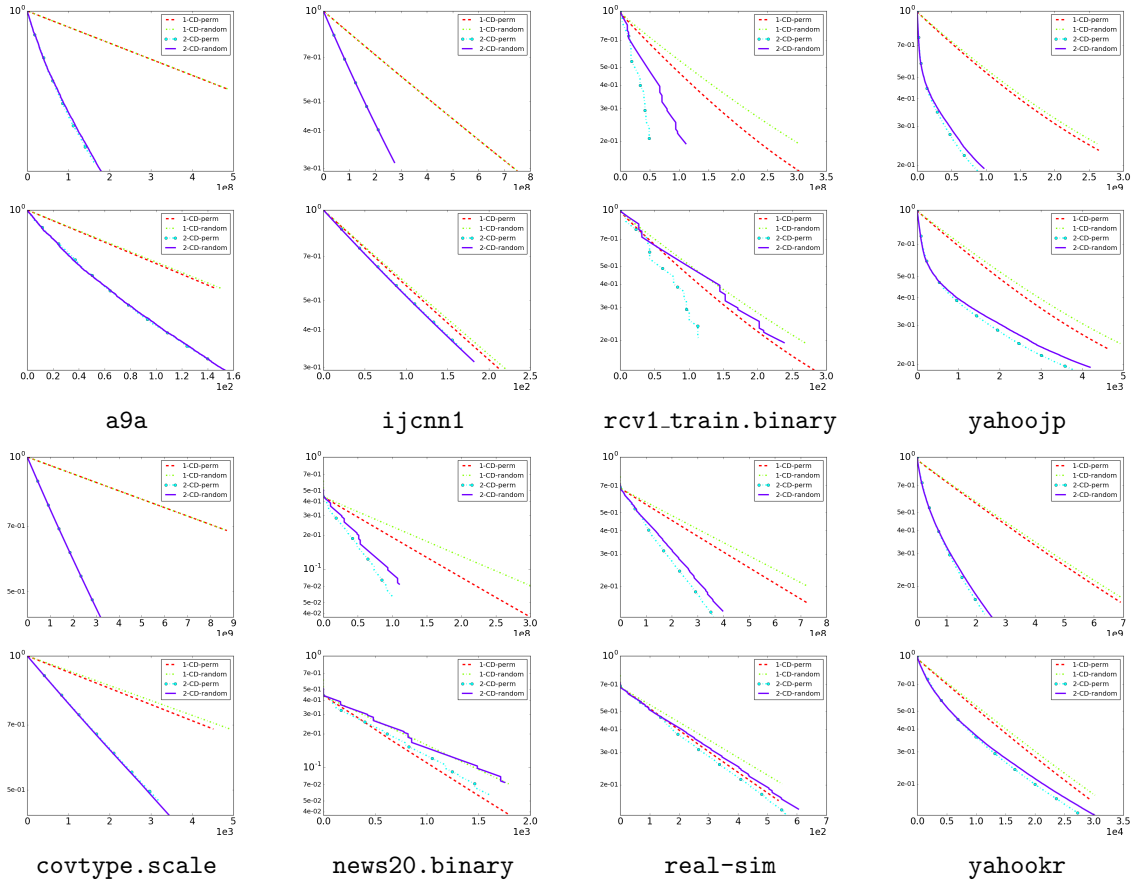


Figure V: A comparison between one-variable and two-variable CD for $l2$ loss with $C = 8,192$. For each set, x-axis in the upper sub-figure is the number of CD steps, while x-axis in the lower sub-figure is the running time (in seconds).

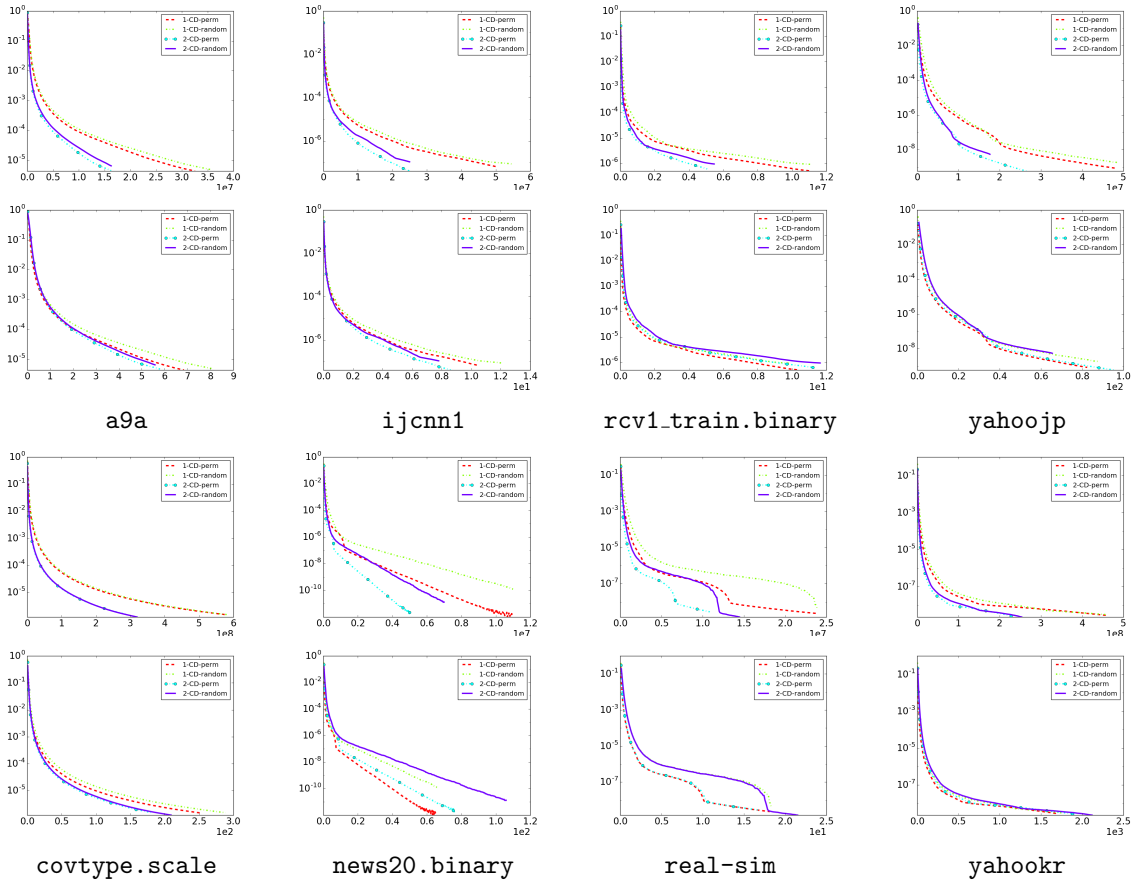


Figure VI: A comparison between one-variable and two-variable CD for $l1$ loss with $C = 1$. For each set, x-axis in the upper sub-figure is the number of CD steps, while x-axis in the lower sub-figure is the running time (in seconds).

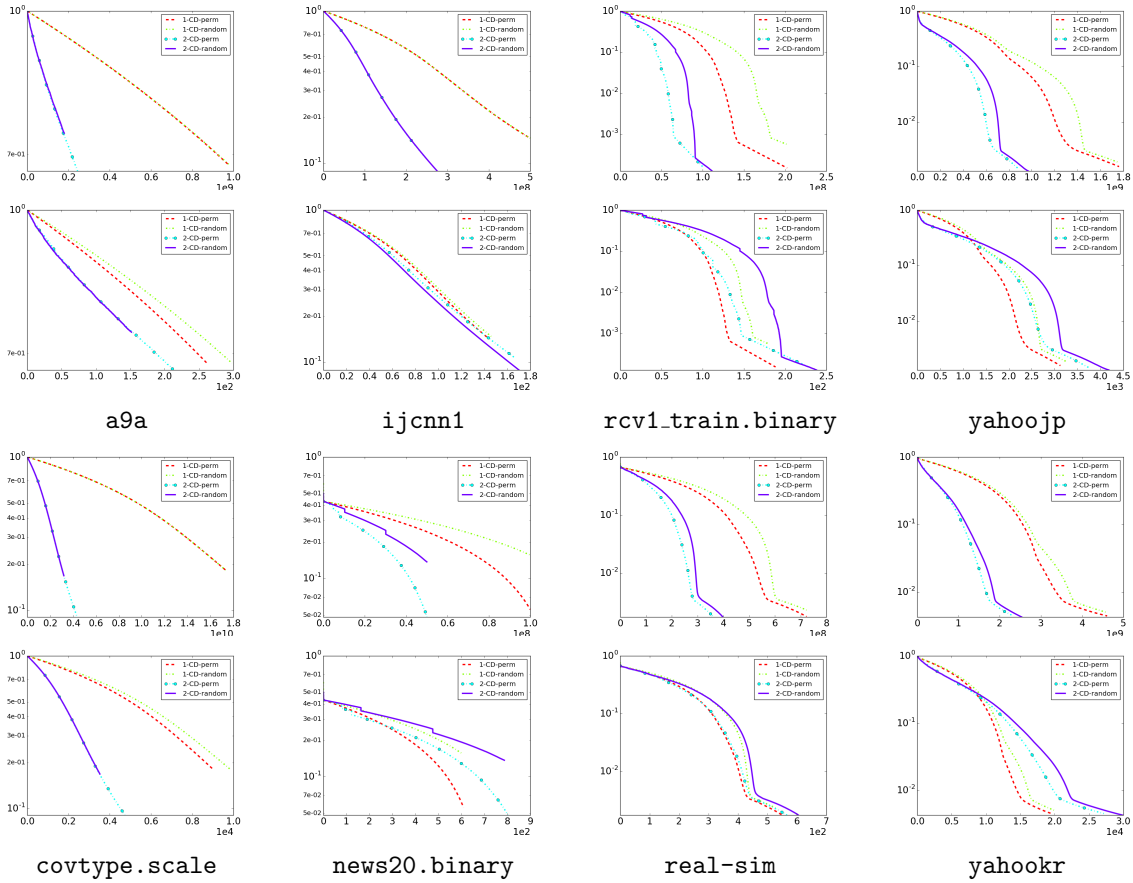
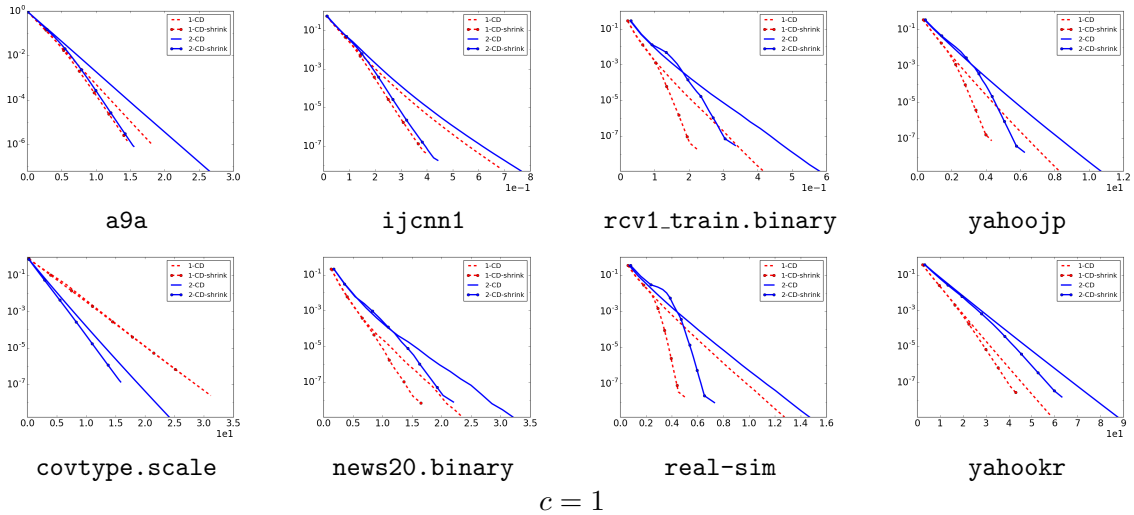
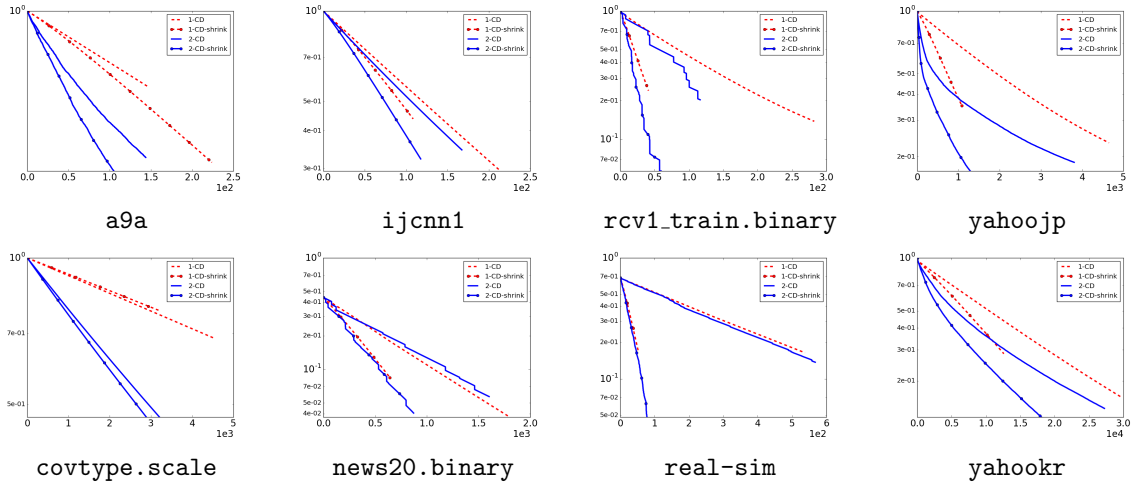


Figure VII: A comparison between one-variable and two-variable CD for $l1$ loss with $C = 8,192$. For each set, x-axis in the upper sub-figure is the number of CD steps, while x-axis in the lower sub-figure is the running time (in seconds).



$c = 1$



$c = 8,192$

Figure VIII: A timing comparison between one-variable and two-variable CD with/without shrinking for $l2$-loss SVM with $C = 1$ and $8,192$. The x-axis is running time in seconds. Note that the shrinking implementation is stopping-tolerance dependent (see line 34 of Algorithm III). Thus the curve is generated by several runs of using different tolerances. It may not be strictly decreasing because of timing fluctuation.

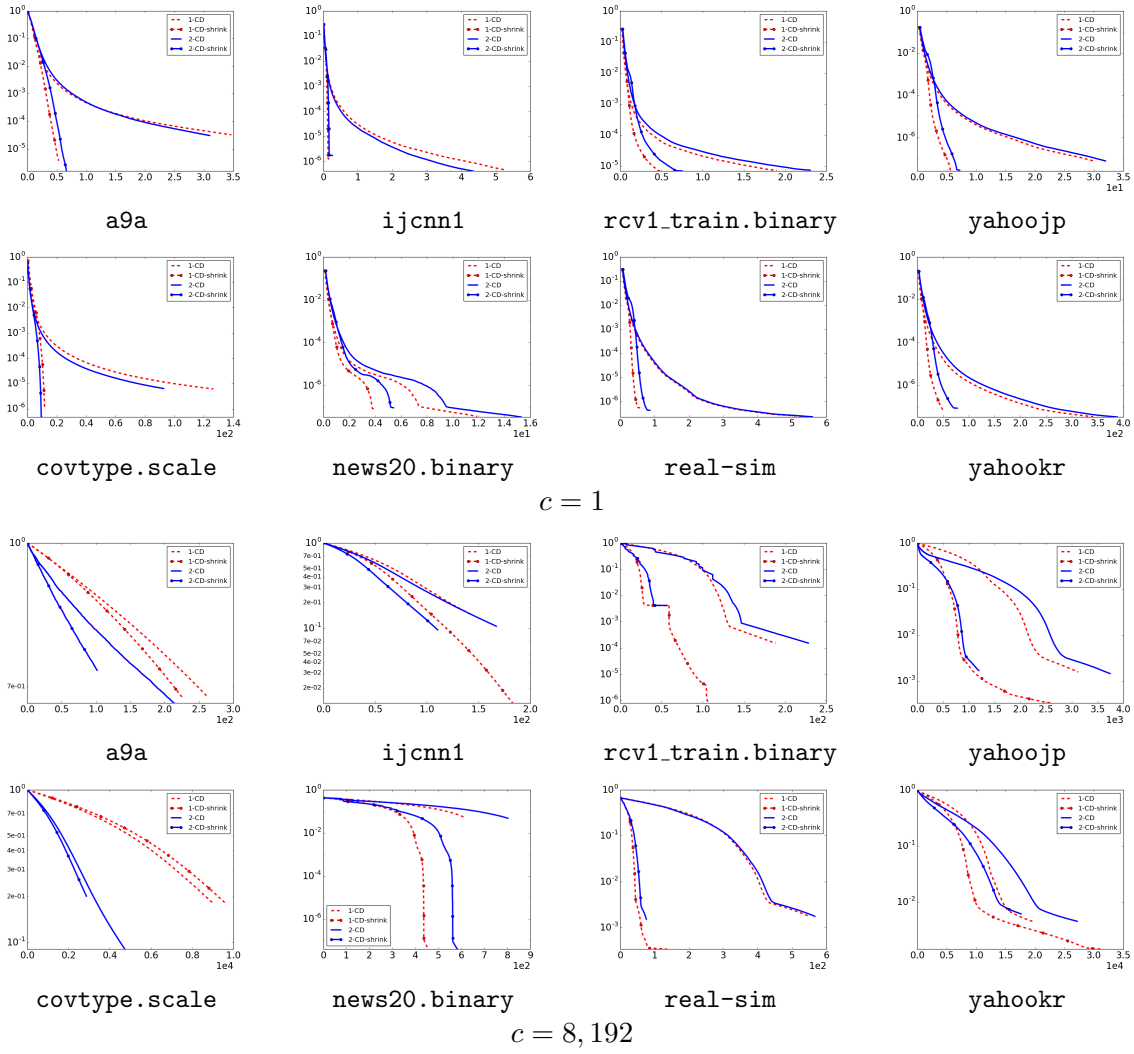


Figure IX: A timing comparison between one-variable and two-variable CD with/without shrinking for $l1$-loss SVM with $C = 1$ and $8,192$. The x-axis is running time in seconds. Note that the shrinking implementation is stopping-tolerance dependent (see line 34 of Algorithm III). Thus the curve is generated by several runs of using different tolerances. It may not be strictly decreasing because of timing fluctuation.

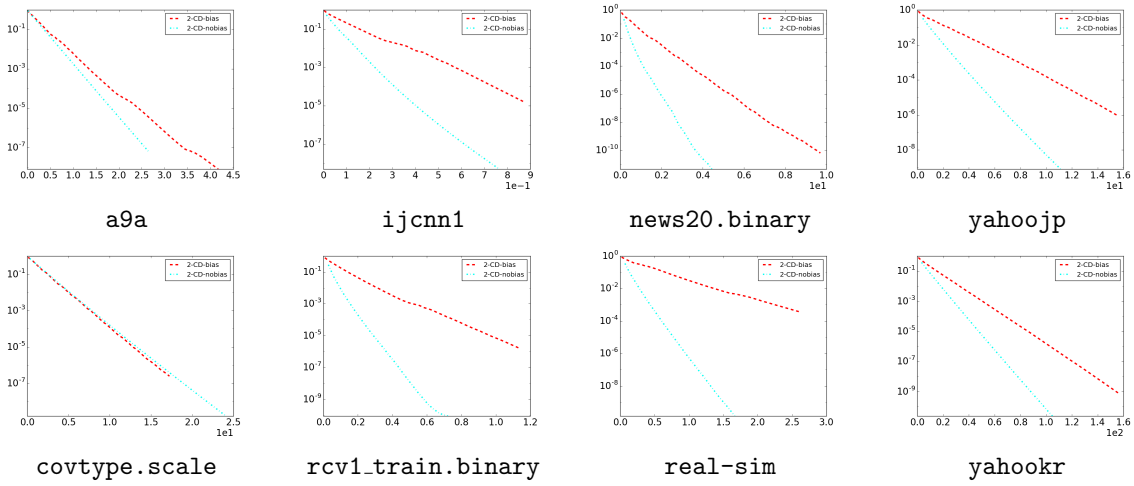


Figure X: Comparison of applying two-variable block CD to solve the SVM problem with/without a bias term. We consider the $l2$ loss and set $C = 1$. The x-axis is the running time in seconds. Shrinking is disabled.

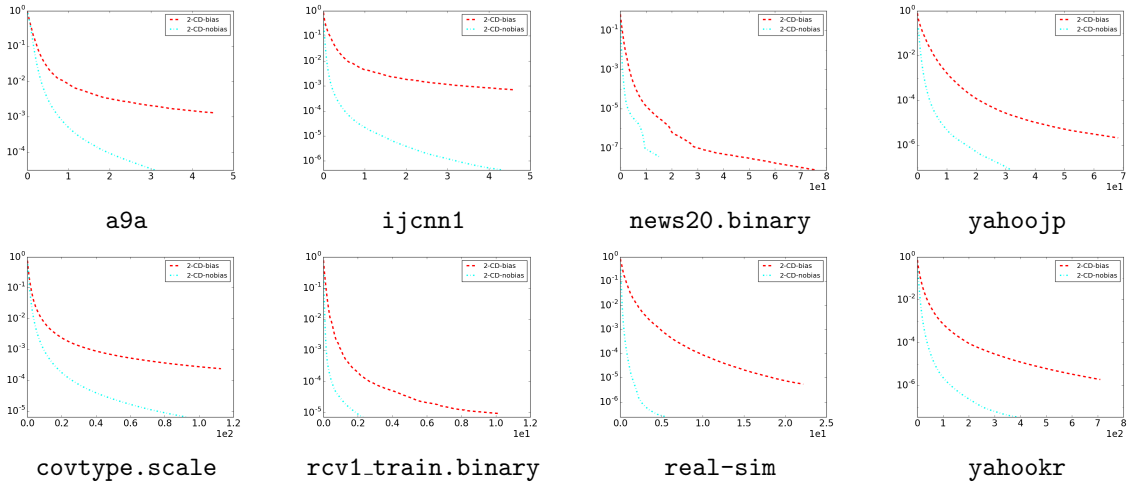


Figure XI: Comparison of applying two-variable block CD to solve the SVM problem with/without a bias term. We consider the $l1$ loss and set $C = 1$. The x-axis is the running time in seconds. Shrinking is disabled.

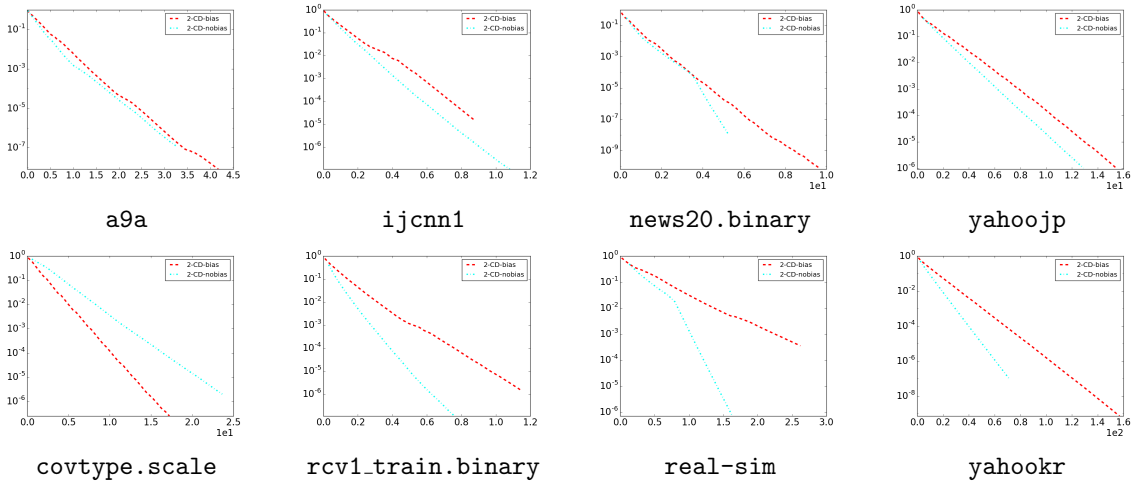


Figure XII: Comparison of applying two-variable block CD to solve dual of SVM with/without a linear constraint. All settings are the same as Figure 5, though for the approach 2-CD-nobias we apply the trick in (2.2) to embed the bias term in the model and still avoid a linear constraint in the dual.

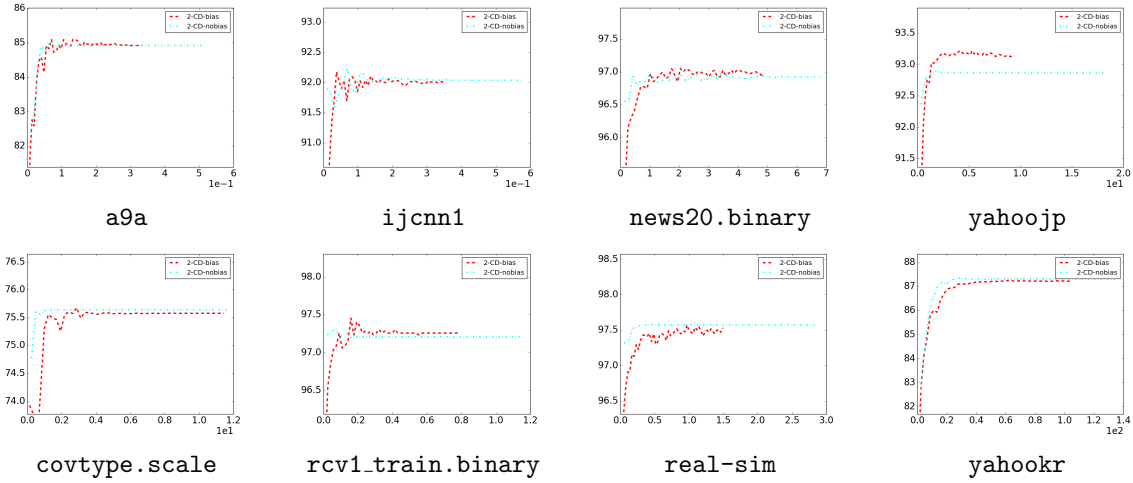


Figure XIII: Comparison of applying two-variable block CD to solve dual of SVM with/without a linear constraint. All settings are the same as Figure 5 but we use cross-validation to find their C parameters, and the y-axis is changed to the test accuracy

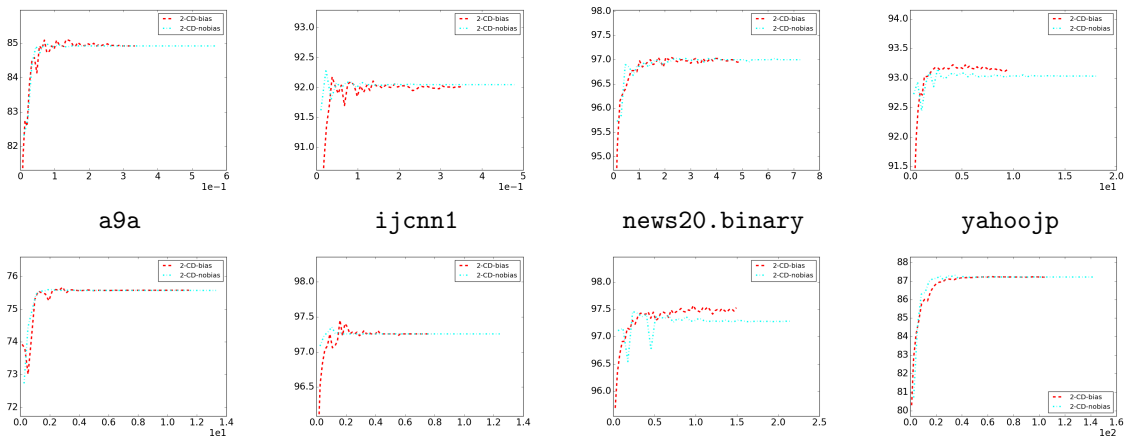


Figure XIV: Comparison of applying two-variable block CD to solve dual of SVM with/without a linear constraint. All settings are the same as Figure 5, though we use cross-validation to find their C parameters, and for the approach 2-CD-nobias we apply the trick in (2.2) to embed the bias term in the model and still avoid a linear constraint in the dual. The y-axis is changed to the test accuracy.

VI. Proofs of Theorems in Section II

VI.I Proof of Theorem II.1

Proof Because $\bar{d}_2$ is an optimal solution of (vii), it satisfies the following optimality condition

$$Q_{12}\bar{d}_1 + Q_{22}\bar{d}_2 + p_2 \begin{cases} \geq 0 & \text{if } \bar{d}_2 < U_2, \\ \leq 0 & \text{if } \bar{d}_2 > L_2. \end{cases} \tag{xxix}$$

This is already the optimality condition corresponding to d_2, so our remaining task of proving that $(\bar{d}_1, \bar{d}_2)$ is optimal is to show that

$$Q_{11}\bar{d}_1 + Q_{12}\bar{d}_2 + p_1 \begin{cases} \geq 0 & \text{if } \bar{d}_1 < U_1, \\ \leq 0 & \text{if } \bar{d}_1 > L_1. \end{cases} \tag{xxx}$$

We prove

$$Q_{11}\bar{d}_1 + Q_{12}\bar{d}_2 + p_1 \leq 0$$

under the situation of

$$d_1^* \geq U_1 > L_1, \ \bar{d}_1 = U_1. \tag{xxxi}$$

The proof of the other situation

$$d_1^* \leq L_1 < U_1, \ \bar{d}_1 = L_1$$

is the same. Note that if

$$U_1 = L_1,$$

then (xxx) directly holds because no $\bar{d}_1 \in (L_1, U_1)$. Now we consider two cases.
Case 1: $Q_{12} \geq 0$

From (xlv),

$$\begin{aligned} \frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} - d_2^* =& \frac{-Q_{12}\bar{d}_1 - p_2 - Q_{22}d_2^*}{Q_{22}} \\ =& \frac{-Q_{12}(\bar{d}_1 - d_1^*)}{Q_{22}} \geq 0, \end{aligned} \tag{xxxii}$$

where (xxxii) is from $Q_{12} \geq 0$ and (xxxi). Thus in finding $\bar{d}_2$ in (vii) we intend to increase d_2^* to $\bar{d}_2$.

We further consider two situations. First,

$$\frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} \leq L_2.$$

Then (xxxii) implies

$$d_2^* \leq \frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} \leq L_2$$

and with (vii),

$$\bar{d}_2 = P[d_2^*] = L_2. \tag{xxxiii}$$

By (xxxi), (xxxiii), and the assumption in (vi), we have the optimality condition

$$Q_{11}\bar{d}_1 + Q_{12}\bar{d}_2 + p_1 = Q_{11}P[d_1^*] + Q_{12}P[d_2^*] + p_1 \leq 0. \tag{xxxiv}$$

If on the other hand

$$\frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} > L_2, \tag{xxxv}$$

then (vii) and (xxxv) imply

$$\bar{d}_2 = \min(U_2, \frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}}) \leq \frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}}. \tag{xxxvi}$$

Therefore,

$$\begin{aligned} \bar{d}_2 - d_2^* \leq & \frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} - d_2^* \\ = & \frac{-Q_{12}(\bar{d}_1 - d_1^*)}{Q_{22}}, \end{aligned} \tag{xxxvii}$$

where (xxxvii) is from (xxxii). Then

$$\begin{aligned} Q_{11}\bar{d}_1 + Q_{12}\bar{d}_2 + p_1 = & Q_{11}(\bar{d}_1 - d_1^*) + Q_{12}(\bar{d}_2 - d_2^*) && \text{(xxxviii)} \\ \leq & (\bar{d}_1 - d_1^*)\frac{Q_{11}Q_{22} - Q_{12}^2}{Q_{22}} && \text{(xxxix)} \\ \leq & 0, && \text{(xl)} \end{aligned}$$

where (xxxviii) is from (xlv), (xxxix) is from (xxxvii), and (xl) is from the positive semi-definiteness of $\left[\begin{smallmatrix} Q_{11} & Q_{12} \\ Q_{12} & Q_{22} \end{smallmatrix}\right]$. This leads to the optimality condition in (xxx).

Case 2: $Q_{12} < 0$

By a similar derivation to (xxxii),

$$\frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} - d_2^* = \frac{-Q_{12}(\bar{d}_1 - d_1^*)}{Q_{22}} \leq 0. \tag{xli}$$

Thus we intend to decrease d_2^* to $\bar{d}_2$.

We also consider two cases. First,

$$\frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} \geq U_2.$$

Then (xli) and (vii) imply

$$\bar{d}_2 = P[d_2^*] = U_2.$$

By (xxxi) and the assumption in (vi), we have the optimality condition in (xxxiv). If on the other hand,

$$\frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} < U_2, \tag{xlii}$$

then (vii) and (xlii) imply

$$\bar{d}_2 = \max(L_2, \frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}}) \geq \frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}}. \tag{xliii}$$

Therefore,

$$\begin{aligned} \bar{d}_2 - d_2^* \geq & \frac{-(Q_{12}\bar{d}_1 + P_2)}{Q_{22}} - d_2^* \\ = & \frac{-Q_{12}(\bar{d}_1 - d_1^*)}{Q_{22}}, \end{aligned} \tag{xliv}$$

where (xliv) is from (xli). With $Q_{12} < 0$, we obtain the same inequalities in (xxxviii)-(xl) and the optimality condition in (xxx). ■

VI.II Proof of Theorem II.2

Proof Consider

$$d_1^* \geq U_1.$$

From (3.16), the optimal solution satisfies

$$\begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12} & Q_{22} \end{bmatrix} \begin{bmatrix} d_1^* \\ d_2^* \end{bmatrix} + \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{xlv}$$

and then

$$\begin{aligned} & Q_{11}U_1 + Q_{12}P[d_2^*] + p_1 \\ = & \, Q_{11}d_1^* + Q_{12}d_2^* + p_1 + Q_{11}(U_1 - d_1^*) \\ = & \, Q_{11}(U_1 - d_1^*) \leq 0, \end{aligned}$$

so the optimality condition is satisfied. The situation for

$$d_1^* \leq L_1$$

is similar. ■

VI.III Proof of Theorem II.3

Proof We consider

$$L_i < U_i, i = 1, 2, \tag{xlvi}$$

because if $L_i = U_i$, the optimality condition directly holds.

We assume $P[d_1^*] = U_1$. The situations of $P[d_1^*] = L_1$ is similar because of the symmetry. Assume the result is wrong. Then $(P[d_1^*], P[d_2^*])$ satisfies neither the optimality condition of d_1 nor that of d_2. We further consider two cases $P[d_2^*] = L_2$ and U_2.

Case 1: $P[d_2^*] = L_2$

Define Δ_1 and Δ_2 as

$$\begin{aligned} \Delta_1 &= P[d_1^*] - d_1^* = U_1 - d_1^* \leq 0, \\ \Delta_2 &= P[d_2^*] - d_2^* = L_2 - d_2^* \geq 0. \end{aligned} \tag{xlvii}$$

Because optimality conditions are violated, with (xlvi),

$$\begin{aligned} Q_{11}P[d_1^*] + Q_{12}P[d_2^*] + p_1 > 0, \\ Q_{12}P[d_1^*] + Q_{22}P[d_2^*] + p_2 < 0. \end{aligned} \tag{xlviii}$$

With (xlv) and (xlvii), (xlviii) becomes

$$\begin{aligned} Q_{11}\Delta_1 + Q_{12}\Delta_2 > 0, \\ Q_{12}\Delta_1 + Q_{22}\Delta_2 < 0. \end{aligned} \tag{xlix}$$

We then have

$$\Delta_1 \neq 0 \text{ or } \Delta_2 \neq 0. \tag{l}$$

Otherwise, (xlix) cannot hold. From (xlix) and (l),

$$\begin{bmatrix} \Delta_1 & \Delta_2 \end{bmatrix} \begin{bmatrix} Q_{11}\Delta_1 + Q_{12}\Delta_2 \\ Q_{12}\Delta_1 + Q_{22}\Delta_2 \end{bmatrix} = \begin{bmatrix} \Delta_1 & \Delta_2 \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12} & Q_{22} \end{bmatrix} \begin{bmatrix} \Delta_1 \\ \Delta_2 \end{bmatrix} < 0.$$

However,

$$\begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12} & Q_{22} \end{bmatrix}$$

is positive semi-definite, so there is a contradiction.

case 2: $P[d_2^*] = U_2$

We have $\Delta_1 \leq 0,\ \Delta_2 \leq 0$. The violation of the result implies

$$\begin{aligned} Q_{11}\Delta_1 + Q_{12}\Delta_2 > 0, \\ Q_{12}\Delta_1 + Q_{22}\Delta_2 > 0. \end{aligned}$$

With (l),

$$\begin{bmatrix} \Delta_1 & \Delta_2 \end{bmatrix} \begin{bmatrix} Q_{11}\Delta_1 + Q_{12}\Delta_2 \\ Q_{12}\Delta_1 + Q_{22}\Delta_2 \end{bmatrix} = \begin{bmatrix} \Delta_1 & \Delta_2 \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{12} & Q_{22} \end{bmatrix} \begin{bmatrix} \Delta_1 \\ \Delta_2 \end{bmatrix} < 0,$$

a contradiction to the positive semi-definiteness of $\left[\begin{smallmatrix} Q_{11} & Q_{12} \\ Q_{12} & Q_{22} \end{smallmatrix}\right]$. ■

VII. Solving the Two-variable Sub-problem (3.17) Without the Proximal Term

It is possible to consider a sub-problem without the proximal term, though the procedure becomes more complicated. Here we show details.

VII.I The Situation of $Q_{ii} = 0$ or $Q_{jj} = 0$

For $l1$-loss SVM, without the proximal term it is possible that $Q_{ii} = 0$ or $Q_{jj} = 0$. Assume $Q_{ii} = 0$. Then

$$Q_{ii} = \|\boldsymbol{x}_i\|^2 = 0$$

implies that $\boldsymbol{x}_i = \mathbf{0}$. From

$$\begin{aligned}\nabla_i f(\boldsymbol{\alpha}) &= \sum_{j=1}^{l} y_i y_j \boldsymbol{x}_i^T \boldsymbol{x}_j \alpha_j - 1 \\ &= -1 \le 0, \ \forall \boldsymbol{\alpha},\end{aligned}$$

by the optimality condition, $\alpha_i = C_i$ is optimal for the dual problem (2.3). We can identify these zero instances before running the CD algorithm.

VII.II Hessian is Positive Semi-definite

For $l2$-loss SVM, from (2.4), Hessian is always positive definite. However, for $l1$-loss SVM, Hessian may be only positive semi-definite and it is possible that

$$Q_{11}Q_{22} - Q_{12}^2 = 0. \tag{li}$$

If Hessian is positive definite, then the solution procedure in Section II can be used. Here we address the situation if (li) occurs. From Section VII.I, after removing zero instances in the beginning, we have

$$Q_{11} > 0 \text{ and } Q_{22} > 0,$$

and therefore (li) implies

$$Q_{12} \neq 0. \tag{lii}$$

When the Hessian is positive definite, in Section II we calculate d_1^* and d_2^*, which play an important role in Algorithm II. With (li), they are not well defined because a division by zero occurs. However, we will show that an extension of (ii) to define d_1^* and d_2^* is possible. We begin with checking the numerator of d_1^* and d_2^* in (ii). From

$$Q_{11}(-Q_{22}p_1 + Q_{12}p_2) = -Q_{12}(-Q_{11}p_2 + Q_{12}p_1),$$

the two numerators have the following relationship.

$$-Q_{22}p_1 + Q_{12}p_2 = \frac{-Q_{12}}{Q_{11}}(-Q_{11}p_2 + Q_{12}p_1). \tag{liii}$$

Now assume that

$$-Q_{22}p_1 + Q_{12}p_2 \neq 0. \tag{liv}$$

We will discuss later how to handle the situation if this value is zero.

From (li), (liii), and (liv), we extend (ii) to define

$$\begin{aligned} d_1^* &= \begin{cases} \infty & \text{if} - Q_{22}p_1 + Q_{12}p_2 > 0, \\ -\infty & \text{if} - Q_{22}p_1 + Q_{12}p_2 < 0, \end{cases} \\ d_2^* &= \begin{cases} \infty & \text{if} - Q_{11}p_2 + Q_{12}p_1 > 0, \\ -\infty & \text{if} - Q_{11}p_2 + Q_{12}p_1 < 0. \end{cases} \end{aligned} \tag{lv}$$

These values can be projected to lower or upper bounds if we make the following assumption.

Assumption VII.1 *We have*

$$-\infty < L_i \leq U_i < \infty, \ i = 1, 2. \tag{lvi}$$

For $l1$-loss SVM, whose Hessian may be only positive semi-definite, this assumption holds because in (2.1) we choose $C < \infty$.

We show in the following theorem that Theorem II.1 and Theorem II.3 can be extended here, so the same Algorithm II can be used without modifications. Note that Theorem II.2 is no longer needed because from Assumption VII.1 and (lv), the condition $d_2^* \in [L_2, U_2]$ never holds.

Theorem VII.2 *Under Assumption VII.1, if*

$$Q_{11}Q_{22} - Q_{12}^2 = 0 \tag{lvii}$$

and d_1^*, d_2^* are defined as in (lv)*, then Theorems II.1 and II.3 hold.*

Proof We begin with checking Theorem II.1. The same proof can almost be used. We also prove only the situation

$$d_1^* \geq U_1, \bar{d}_1 = U_1.$$

From the definition in (lv), this in fact means

$$d_1^* = \infty, \ P[d_1^*] = \bar{d}_1 = U_1. \tag{lviii}$$

Further, (lviii) and (lv) imply

$$-Q_{22}p_1 + Q_{12}p_2 > 0. \tag{lix}$$

We now consider $Q_{12} > 0$, while the proof for $Q_{12} < 0$ is similar. Note that we have $Q_{12} \neq 0$ from (lii).

The same as in Theorem II.1, we further consider two situations. First,

$$\frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} \leq L_2.$$

From (lix), $Q_{12} > 0$, (liii) and (lv), we have

$$d_2^* = -\infty \leq \frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} \leq L_2. \tag{lx}$$

Then (xxxiii) and (xxxiv) follow, so we have the needed optimality condition

If on the other hand,

$$\frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} > L_2,$$

then (xliii) holds. We now check the optimality condition of d_1:

$$\begin{aligned}
& Q_{11}\bar{d}_1 + Q_{12}\bar{d}_2 + p_1 \\
\leq & Q_{11}\bar{d}_1 + Q_{12}\frac{-(Q_{12}\bar{d}_1 + p_2)}{Q_{22}} + p_1 && \text{(lxi)} \\
= & \frac{-Q_{12}p_1 + Q_{22}p_1}{Q_{22}} && \text{(lxii)} \\
\leq & 0, && \text{(lxiii)}
\end{aligned}$$

where (lxi) is from $Q_{12} \geq 0$ and (xliii), (lxii) is from (lvii), and (lxiii) is from (lix).

Next, to extend Theorem II.3 we follow the same setting to consider

$$d_1^* = +\infty,\ P[d_1^*] = U_1 \tag{lxiv}$$

and check the two cases $P[d_2^*] = L_2$ or U_2.
Case 1: $P[d_2^*] = L_2$
For this case

$$d_2^* = -\infty \text{ and } P[d_2^*] = L_2.$$

From (liii) and (lv),

$$Q_{12} > 0. \tag{lxv}$$

If the result in Theorem II.3 is wrong, both optimality conditions are violated and

$$\begin{aligned}
Q_{11}U_1 + Q_{12}L_2 + p_1 > 0, \\
Q_{12}U_1 + Q_{22}L_2 + p_2 < 0.
\end{aligned}$$

With (li) and (lxv),

$$\begin{aligned}
& Q_{22}(Q_{11}U_1 + Q_{12}L_2 + p_1) \\
> & 0 && \text{(lxvi)} \\
> & Q_{12}(Q_{12}U_1 + Q_{22}L_2 + p_2)
\end{aligned}$$

leads to

$$-Q_{22}p_1 + Q_{12}p_2 < 0. \tag{lxvii}$$

From (lv), we obtain a contradiction to $d_1^* = \infty$ in (lxiv).
Case 2: $P[d_2^*] = U_2$
For this case

$$d_2^* = \infty \text{ and } P[d_2^*] = U_2.$$

From (liii) and (lv),

$$Q_{12} < 0. \tag{lxviii}$$

The same with the last case, we assume the optimality conditions are violated and therefore

$$Q_{11}U_1 + Q_{12}L_2 + p_1 > 0,$$
$$Q_{12}U_1 + Q_{22}L_2 + p_2 > 0.$$

With (lxviii), we can have (lxvi) and (lxvii). Then (lxvii) contradicts the assumption. ■

We now show that the same procedure in Algorithm II can be used. From (lv) and Assumption VII.1,

$$d_1^* \notin [L_1, U_1] \text{ and } d_2^* \notin [L_2, U_2]. \tag{lxix}$$

Then $P[d_1^*]$ and $P[d_2^*]$ are bounded. We check if $(P[d_1^*], P[d_2^*])$ satisfies the optimality condition of d_1. If it does, then from Theorem VII.2, we can apply Theorem II.1 to use (vii) for obtaining a solution. Otherwise, from Theorem VII.2 and (lxix), we apply Theorem II.3 to have that $(P[d_1^*], P[d_2^*])$ satisfies the optimality condition of d_2. Then we apply Theorem II.1 to obtain an optimal solution as in (x). Therefore, the solution procedure is exactly the same as the procedure in Algorithm II for positive-definite Hessian.

Next we discuss the rare situation where both

$$Q_{11}Q_{22} - Q_{12}^2 = 0 \tag{lxx}$$

and

$$-Q_{22}p_1 + Q_{12}p_2 = 0. \tag{lxxi}$$

The objective function can be written as

$$\begin{aligned} &\frac{1}{2}Q_{11}d_1^2 + Q_{12}d_1d_2 + \frac{1}{2}Q_{22}d_2^2 + p_1d_1 + p_2d_2 \\ =&\frac{1}{2Q_{11}}(Q_{11}d_1 + Q_{12}d_2 + p_1)^2 + \text{constant}, \end{aligned} \tag{lxxii}$$

where for the linear term of d_2, we use (lxx)-(lxxi) and (lii) to have

$$\frac{Q_{12}p_1d_2}{Q_{11}} = \frac{Q_{12}^2p_2d_2}{Q_{11}Q_{22}} = p_2d_2. \tag{lxxiii}$$

From (lxxii), the optimization problem becomes to find a point in the feasible region

$$L_1 \le d_1 \le U_1,\ L_2 \le d_2 \le U_2, \tag{lxxiv}$$

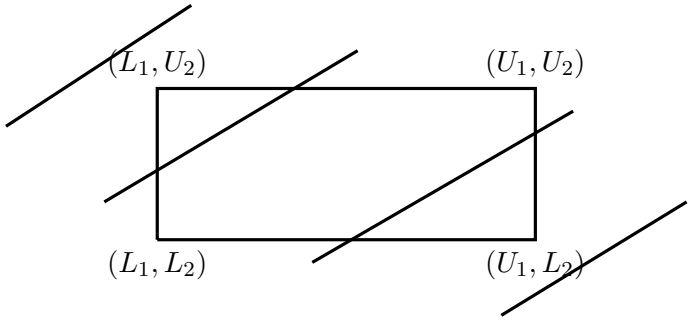
that is the closest to the plane

$$Q_{11}d_1 + Q_{12}d_2 + p_1 = 0. \tag{lxxv}$$

We then give the details in Section VII.III to design Algorithm IV for finding an optimal solution.

VII.III Solution Procedure when (lxx) and (lxxi) Both Happen

It is easy to identify an optimal solution by checking the geometric relationship between the feasible region and the straight line in (lxxv). If $Q_{12} < 0$, then the line has a positive slope. Thus we have the following four possible situations.



If

$$Q_{11}U_1 + Q_{12}L_2 + p_1 \le 0, \tag{lxxvi}$$

then the whole feasible region is on the left side of the line and (U_1, L_2) is the closest point.

If (lxxvi) dose not hold and

$$Q_{11}L_1 + Q_{12}L_2 + p_1 \le 0,$$

then from the figure, a line segment is the interaction between the line and the region. Any point on this line segment is an optimal solution. We can simply consider the interaction point on the horizontal line $d_2 = L_2$:

$$\bar{d}_1 = \frac{-Q_{12}L_2 - p_1}{Q_{11}},\ \bar{d}_2 = L_2. \tag{lxxvii}$$

Other situations are similar. A summary of the procedure is in Algorithm IV. For practical implementations we switch from $d_1,\ d_2$ back to $\alpha_i,\ \alpha_j$ by

$$U_1 \equiv C_i - \alpha_i,\ L_1 \equiv -\alpha_i \text{ and } p_1 = \nabla_i f(\boldsymbol{\alpha}).$$

Note that the plane (lxxv) becomes

$$Q_{ii}\bar{\alpha}_i + Q_{ij}\bar{\alpha}_j = \delta \equiv Q_{ii}\alpha_i + Q_{ij}\alpha_j - p_i, \tag{lxxviii}$$

where $(\bar{\alpha}_i, \bar{\alpha}_j)$ and (α_i, α_j) are the variable and the current iterate, respectively.

References

C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM TIST*, 2(3):27:1–27:27, 2011.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: a library for large linear classification. *Journal of Machine Learning Research*, 9: 1871–1874, 2008. URL `http://www.csie.ntu.edu.tw/~cjlin/papers/liblinear.pdf`.

C.-J. Hsieh, K.-W. Chang, C.-J. Lin, S. S. Keerthi, and S. Sundararajan. A dual coordinate descent method for large-scale linear SVM. In *ICML*, 2008a.

Cho-Jui Hsieh, Kai-Wei Chang, Chih-Jen Lin, S. Sathiya Keerthi, and Sellamanickam Sundararajan. A dual coordinate descent method for large-scale linear SVM. In *Proceedings of the Twenty Fifth International Conference on Machine Learning (ICML)*, 2008b. URL `http://www.csie.ntu.edu.tw/~cjlin/papers/cddual.pdf`.

Thorsten Joachims. Making large-scale SVM learning practical. In *Advances in Kernel Methods - Support Vector Learning*. MIT Press, 1998.

Ingo Steinwart, Don Hush, and Clint Scovel. Training SVMs without offset. *JMLR*, 12: 141–202, 2011.

Po-Wei Wang and Chih-Jen Lin. Iteration complexity of feasible descent methods for convex optimization. *JMLR*, 2014.