Block-Coordinate Frank-Wolfe for Structural SVMs

Martin Jaggi[*]
CMAP
École Polytechnique, Palaiseau, France

Simon Lacoste-Julien[*]
INRIA - SIERRA project-team
École Normale Supérieure, Paris, France

Mark Schmidt
INRIA - SIERRA project-team
École Normale Supérieure, Paris, France

Patrick Pletscher
Machine Learning Laboratory
ETH Zurich, Switzerland

Abstract

We propose a randomized block-coordinate variant of the classic Frank-Wolfe algorithm for convex optimization with block-separable constraints. Despite its lower iteration cost, we show that it achieves the same convergence rate as the full Frank-Wolfe algorithm. We also show that, when applied to the dual structural support vector machine (SVM) objective, this algorithm has the same low iteration complexity as primal stochastic subgradient methods. However, unlike stochastic subgradient methods, the stochastic Frank-Wolfe algorithm allows us to compute the optimal step-size and yields a computable duality gap guarantee. Our experiments indicate that this simple algorithm outperforms competing structural SVM solvers.

1 Introduction

Binary SVMs are an immensely popular classification method, and this has motivated substantial interest in optimization solvers that are tailored to their specific problem structure. However, despite its wider applicability, there has been much less work on solving the optimization problem associated with *structural* SVMs, which is the generalization of SVMs to structured outputs like graphs and other combinatorial objects [1, 2]. This seems to be due to the difficulty of dealing with the exponential number of constraints in the primal problem, or the exponential number of variables in the dual problem. Indeed, because they achieve an $\tilde{O}(1/\varepsilon)$ convergence rate while only requiring a single call to the so-called *maximization oracle* on each iteration, basic stochastic subgradient methods are widely-used for training structural SVMs [3, 4]. However, these methods are often frustrating to use for practitioners, because their performance is very sensitive to the sequence of step sizes, and because it is difficult to decide when to terminate the iterations.

To solve the dual structural SVM problem, in this paper we consider the Frank-Wolfe [5] algorithm, which has seen a recent surge of interest in machine learning and signal processing [6, 7, 8, 9], including in the context of binary SVMs [10, 11]. A key advantage of this algorithm is that the iterates are *sparse*, and we show that this allows us to efficiently apply it to the dual structural SVM objective even though there are an exponential number of variables. A second key advantage of this algorithm is that the iterations only require optimizing linear functions over the constrained domain, and we show that *this is equivalent to the maximization oracle* used by subgradient and cutting-plane [12, 13] methods. Thus, the Frank-Wolfe algorithm has the same wide applicability as subgradient methods, and can be applied to problems such as low-treewidth graphical models [1], graph matchings [14], and associative Markov networks [15]. In contrast, other approaches must use more expensive oracles, such as doing a Bregman projection onto the space of structures [16], or computing marginals over labels [17, 18] which is intractable in some of these cases. Interestingly, for structural SVMs we also show that existing deterministic-subgradient and cutting-plane

---

[*]Both authors contributed equally.

methods are *special cases* of Frank-Wolfe algorithms, and this leads to stronger and simpler $O(1/\varepsilon)$ convergence rate guarantees for these existing algorithms.

As in other structural SVM solvers like cutting-plane methods [12, 13] and the excessive gap technique [18], each Frank-Wolfe iteration unfortunately requires calling the appropriate oracle once for *all* training examples, unlike the single oracle call needed by stochastic subgradient methods. This can be prohibitive for data sets with a large number of training examples. To reduce this cost, we propose a novel randomized block-coordinate version of the Frank-Wolfe algorithm for problems with block-separable constraints. We show that this algorithm still achieves the $O(1/\varepsilon)$ convergence rate of the full Frank-Wolfe algorithm, and in the context of structural SVMs it only requires a single call to the maximization oracle. Although the stochastic subgradient and the novel block-coordinate Frank-Wolfe algorithms have a similar iteration cost and theoretical convergence rate for solving the structural SVM problem, the new algorithm has several important advantages for practitioners:

- The *optimal* step-size can be efficiently computed in closed-form, so no step-size needs to be selected.
- The algorithm yields a *duality gap* guarantee, and (at the cost of computing the primal objective) we can compute the duality gap as a proper stopping criterion.
- The convergence rate holds even when using *approximate* maximization oracles.

Further, our experimental results show that the optimal step-size leads to a significant advantage during the first few passes through the data, and a systematic (but smaller) advantage in later passes.

2   Structural Support Vector Machines

In structured prediction, the goal is to predict a structured object $y \in \mathcal{Y}(x)$ (such as a sequence of tags) for a given input $x \in \mathcal{X}$. In the standard approach [1, 2], a structured feature map $\phi : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^d$ encodes the relevant information for input/output pairs, and a linear classifier with parameter w is defined by $h_w(x) = \text{argmax}_{y \in \mathcal{Y}(x)} \langle w, \phi(x, y) \rangle$. Given a labelled training set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$, w is estimated by solving

$$\min_{w} \quad p(w) := \frac{\lambda}{2} \|w\|^2 + \frac{1}{n} \sum_{i=1}^n \max_{y \in \mathcal{Y}_i} \{ \underbrace{L(y_i, y)}_{=: L_i(y)} - \langle w, \underbrace{\phi(x_i, y_i) - \phi(x_i, y)}_{=: \psi_i(y)} \rangle \}. \tag{1}$$

Here $L_i(y)$ denotes the task-dependent structured error of predicting y instead of the observed output y_i, and this is typically a Hamming distance between the two labels. Despite the combinatorial nature of $\mathcal{Y}$, as discussed in the introduction solving the inner maximization in (1) for an individual example i can be done efficiently in many settings. A sub-routine that solves this problem is called a *maximization oracle*, and having such a sub-routine allows us to apply subgradient methods to the problem. We can also write this non-smooth optimization problem as a quadratic program with an exponential number of constraints, and by taking the Lagrange dual of this problem, we obtain

$$\max_{\boldsymbol{\alpha} \in \mathcal{M}} \quad d(\boldsymbol{\alpha}) := b^T \boldsymbol{\alpha} - \frac{\lambda}{2} \|A\boldsymbol{\alpha}\|^2. \tag{2}$$

The number of elements m of $\boldsymbol{\alpha}$ is $\sum_i |\mathcal{Y}_i|$, so this is a quadratic program with an exponential number of variables or potential "support vectors". The matrix $A \in \mathbb{R}^{d \times m}$ consists of the m columns $A := \left\{ \frac{1}{\lambda n} \psi_i(y) \in \mathbb{R}^d \mid i \in [n], y \in \mathcal{Y}_i \right\}$, we define $b := \left(\frac{1}{n} L_i(y) \right)_{i \in [n], y \in \mathcal{Y}_i}$, and the domain $\mathcal{M} \subset \mathbb{R}^m$ is the product of n probability simplices, $\mathcal{M} := \Delta_{|\mathcal{Y}_1|} \times \ldots \times \Delta_{|\mathcal{Y}_n|}$. The primal-dual correspondence between w and $\boldsymbol{\alpha}$ obtained from the KKT conditions is simply $w = A\boldsymbol{\alpha}$.

3   Frank-Wolfe Algorithms

Projected-gradient methods are a natural approach for optimizing over the product of probability simplices. However, in each iteration these methods must optimize a quadratic function over the constraint set, and the exponential number of variables makes this intractable for (2). In contrast, the Frank-Wolfe algorithm [5] (also known as the *conditional-gradient* method) only requires optimizing *linear* functions over $\mathcal{M}$, and in particular computing $\text{argmin}_{s \in \mathcal{M}} \langle s, \nabla f(\boldsymbol{\alpha}) \rangle$. In the extended version of this paper [19], we show that a maximization oracle yields a solution to this problem that has only a single non-zero element, allowing us to efficiently apply the Frank-Wolfe algorithm to the structural SVM dual problem. We also show that this is *equivalent* to applying

the deterministic subgradient method (though the Frank-Wolfe perspective allows us to use a more clever choice of step-size within the method), and that existing cutting-plane methods are equivalent to the "fully-corrective" variant of the Frank-Wolfe algorithm [7]. By using existing results from the Frank-Wolfe literature [7, 8], this allows us to show that the Frank-Wolfe algorithm, as well as deterministic subgradient and cutting-plane methods, achieve a *duality gap* less than ε after at most $O(1/\varepsilon)$ iterations for training structural SVMs.

4 Block-Coordinate Frank-Wolfe Algorithm

When applied to the structural SVM problem, the Frank-Wolfe algorithm requires n calls to the maximization oracle on each iteration. This can make the algorithm impractical in cases where the number of training examples n is large. In this section we consider a *randomized block-coordinate* generalization of the Frank-Wolfe algorithm, where at each iteration the algorithm chooses a block i uniformly at random and applies a Frank-Wolfe update to the block. Algorithm 1 gives the new method, which can be applied to any convex optimization problem of the form

$$\min_{\boldsymbol{\alpha}\in\mathcal{M}^{(1)}\times\ldots\times\mathcal{M}^{(n)}} f(\boldsymbol{\alpha}), \tag{3}$$

where the domain has the structure of a Cartesian product $\mathcal{M} = \mathcal{M}^{(1)} \times \ldots \times \mathcal{M}^{(n)} \subseteq \mathbb{R}^m$ over $n \geq 1$ blocks (with each block convex and compact). The algorithm can be interpreted as a variant of Nesterov's "huge-scale" coordinate descent method [20, Section 4], where the projected-gradient step is replaced by a Frank-Wolfe step.

The block-coordinate Frank-Wolfe method can have much cheaper iterations than the classical Frank-Wolfe method, since each update only affects a single variable block. In the context of structural SVMs, this method only requires a single call to the maximization oracle on each iteration. Despite this, our convergence results show that after $O(1/\varepsilon)$ iterations, Algorithm 1 still obtains an ε-small duality gap (as in the full Frank-Wolfe algorithm). Below, we present a specialization of our result to the case of structural SVMs, where we use $R := \max_{i,\boldsymbol{y}}\{\|\boldsymbol{\psi}_i(\boldsymbol{y})\|\}$ and $L_{\max} := \max_{i,\boldsymbol{y}}\{L_i(\boldsymbol{y})\}$.

Algorithm 1: Block-Coordinate Frank-Wolfe

Let $\boldsymbol{\alpha}^{(0)} \in \mathcal{M} = \mathcal{M}^{(1)} \times \ldots \times \mathcal{M}^{(n)}$
for $k = 0 \ldots K$ **do**
 Pick $i \in_{u.a.r.} [n]$
 Find $\boldsymbol{s}_{(i)} := \underset{\boldsymbol{s}'_{(i)}\in\mathcal{M}^{(i)}}{\operatorname{argmin}} \left\langle \boldsymbol{s}'_{(i)}, \nabla_{(i)}f(\boldsymbol{\alpha}^k) \right\rangle$
 Set $\gamma := \frac{2n}{k+2n}$, or find the optimal γ
 Update $\boldsymbol{\alpha}^{k+1}_{(i)} := \boldsymbol{\alpha}^k_{(i)} + \gamma\big(\boldsymbol{s}_{(i)} - \boldsymbol{\alpha}^k_{(i)}\big)$
end

Theorem 1. *If $\lambda n \leq \frac{2R^2}{L_{max}}$, then Algorithm 1 obtains an expected duality gap* $\mathrm{E}[p(\boldsymbol{w}^k)-d(\boldsymbol{\alpha}^k)] \leq \varepsilon$ *after at most $O\left(\frac{R^2}{\lambda\varepsilon}\right)$ iterations for the primal-dual pair* (1) *and* (2).

If $\lambda n > \frac{2R^2}{L_{max}}$, then using $\gamma = \frac{2n}{k+2n}$ requires an additional $O\left(\frac{nL_{max}}{\varepsilon}\right)$ steps to obtain this guarantee, and optimally setting γ requires an additional $2n\log\left(\frac{\lambda n L_{max}}{2R^2}\right)$ steps.

In terms of ε, the $O(1/\varepsilon)$ convergence rate above is the same as existing stochastic subgradient and cutting-plane methods. However, unlike cutting-plane methods which require $O(n)$ oracle calls per iteration, this rate is achieved "online" using only a single oracle call per iteration. Further, unlike stochastic subgradient methods, the stochastic Frank-Wolfe method does not require setting a step-size (we can use the structure of (2) to efficiently solve for the optimal γ in closed-form), obtains duality gap guarantees, and allows us to compute the current duality gap in order to aid in deciding when to terminate the algorithm (at the cost of an extra pass through the data).

5 Experiments

We compare our novel Frank-Wolfe approach to existing algorithms for training structural SVMs on the OCR dataset ($n = 6251, d = 4028$) from [1] and the CoNLL dataset ($n = 8936, d = 1643026$) from [21]. Both datasets are sequence labeling tasks, where the loss-augmented decoding problem can be solved exactly by the Viterbi algorithm. Our third application is a word alignment problem between sentences in different languages in the setting of [16] ($n = 5000, d = 82$).

The methods in our comparison are the batch Frank-Wolfe algorithm with line-search (*FW*) and our novel block-coordinate Frank-Wolfe (*BCFW*) method with line-search, the *cutting plane* al-

Figure 1: The shaded areas for the stochastic methods (*BCFW*, *SSG* and *online-EG*) indicate the worst and best objective achieved in 10 randomized runs. The top row compares the suboptimality achieved by different solvers for different regularization parameters λ. For large λ (a), the stochastic algorithms (*BCFW* and *SSG*) perform considerably better than the batch solvers (*cutting plane* and *FW*). For a small λ (c), even the batch solvers achieve a lower objective earlier on than *SSG*. Our proposed *BCFW* algorithm achieves a low objective in both settings. (d) shows the convergence for CoNLL with the first passes in more details. Here *BCFW* already results in a low objective even after seeing only few datapoints. The same can be observed for the test error in (e). Finally, (f) compares the stochastic methods for the matching prediction task.

gorithm implemented in SVMstruct [12] with its default options, the online exponentiated gradient (*online-EG*) method of [17], the stochastic subgradient method (*SSG*) with step-size chosen as in the "pegasos" version of [4], and the optimal stochastic subgradient (*SSG-tavg*) method of [22] which is the same as *SSG* but the second half of the iterates are averaged (yielding a faster convergence rate of $O(1/k)$ instead of $O(\log k/k)$). Analogously, *BCFW-tavg* uses averaging in the second half. The performance of the different algorithms according to several criteria is visualized in Figure 1. Additional experiments and a more detailed discussion can be found in the full version of this paper [19]. In most of the experiments, the randomized block-coordinate Frank-Wolfe dominates all competitors. The superiority is especially striking for the first few iterations, and when using a small regularization strength λ, which is often needed in practice.

6   Discussion

Related Work   There has been substantial work on dual coordinate ascent for SVMs, including the original SMO algorithm, but few of these lead to rate guarantees in the structured case. The SMO algorithm was generalized to structural SVMs [15, Chapter 6], but this requires something equivalent to an expectation oracle and its convergence rate seems to scale badly with the size of the output space. [23] consider optimizing one training example at a time using multiple Frank-Wolfe updates, but do not obtain any rate guarantees. Our stochastic Frank-Wolfe algorithm is equivalent to the method of [24] in the degenerate *binary* SVM case. [24] shows a local linear convergence rate in the dual, and our result complements this result by providing duality gap guarantees for their algorithm. Another generalization of [24] to the structured case is [25], but without rate guarantees.

Approximate Maximization Oracles   Interestingly, our convergence rates still hold for appropriately defined *approximate* maximization oracles. For structural SVMs, this significantly improves the applicability to large-scale problems, where in some cases exact maximization may be too costly but approximate maximization is possible.

Kernelized Algorithms   Our Algorithm 1 can directly be used with kernels by maintaining the *sequence* of sparse dual variables $\boldsymbol{\alpha}^k$. We note that this leads to the currently best known upper bound on the *number of support vectors*, since we are guaranteed an ε accuracy using only $O(\frac{R^2}{\lambda \varepsilon})$ support vectors.

References

[1] B. Taskar, C. Guestrin, and D. Koller. Max-Margin Markov Networks. In *NIPS*, 2003.

[2] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun. Large margin methods for structured and interdependent output variables. *JMLR*, 6:1453–1484, 2005.

[3] N. Ratliff, J. A. Bagnell, and M. Zinkevich. (online) subgradient methods for structured prediction. In *AISTATS*, 2007.

[4] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter. Pegasos: primal estimated sub-gradient solver for SVM. *Mathematical Programming*, 127(1), 2010.

[5] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

[6] O.L. Mangasarian. Machine learning via polyhedral concave minimization. Technical report, 1995.

[7] K. Clarkson. Coresets, sparse greedy approximation, and the Frank-Wolfe algorithm. *ACM Transactions on Algorithms*, 6(4):1–30, 2010.

[8] M. Jaggi. *Sparse convex optimization methods for machine learning*. PhD thesis, ETH Zürich, 2011.

[9] F. Bach, S. Lacoste-Julien, and G. Obozinski. On the equivalence between herding and conditional gradient algorithms. In *ICML*, 2012.

[10] B. Gärtner and M. Jaggi. Coresets for polytope distance. *ACM Symposium on Computational Geometry*, 2009.

[11] H. Ouyang and A. Gray. Fast stochastic Frank-Wolfe algorithms for nonlinear SVMs. *SDM*, 2010.

[12] T. Joachims, T. Finley, and C. Yu. Cutting-plane training of structural SVMs. *Machine Learn.*, 77(1):27–59, 2009.

[13] C.H. Teo, S.V.N. Vishwanathan, A.J. Smola, and Q.V. Le. Bundle methods for regularized risk minimization. *JMLR*, 11:311–365, 2010.

[14] T.S. Caetano, J.J. McAuley, Li Cheng, Q.V. Le, and A.J. Smola. Learning graph matching. *IEEE PAMI*, 31(6):1048 –1058, 2009.

[15] B. Taskar. *Learning structured prediction models: A large margin approach*. PhD thesis, Stanford, 2004.

[16] B. Taskar, S. Lacoste-Julien, and M. I. Jordan. Structured prediction, dual extragradient and Bregman projections. *JMLR*, 7:1627–1653, 2006.

[17] M. Collins, A. Globerson, T. Koo, X. Carreras, and P. L. Bartlett. Exponentiated gradient algorithms for conditional random fields and max-margin Markov networks. *JMLR*, 9:1775–1822, 2008.

[18] X. Zhang, A. Saha, and S. V. N. Vishwanathan. Accelerated training of max-margin markov networks with kernels. In *ALT*, pages 292–307. Springer, 2011.

[19] S. Lacoste-Julien, M. Jaggi, M. Schmidt, and P. Pletscher. Block-Coordinate Frank-Wolfe Optimization for Structural SVMs. *arXiv*, cs.LG, 2012.

[20] Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems, 2010.

[21] E.F.T.K. Sang and S. Buchholz. Introduction to the CoNLL-2000 shared task: Chunking, 2000.

[22] A. Rakhlin, O. Shamir, and K. Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. In *ICML*, 2012.

[23] J. Rousu, C. Saunders, S. Szedmak, and J. Shawe-Taylor. Kernel-based learning of hierarchical multilabel classification models. *JMLR*, 2006.

[24] C. Hsieh, K. Chang, C. Lin, S. Keerthi, and S. Sundararajan. A dual coordinate descent method for large-scale linear SVM. In *ICML*, pages 408–415, 2008.

[25] P. Balamurugan, S. Shevade, S. Sundararajan, and S. Keerthi. A sequential dual method for structural SVMs. In *SDM*, 2011.