Train and Test Tightness of LP Relaxations in Structured Prediction

Ofer Meshi   MESHI@TTIC.EDU
Mehrdad Mahdavi   MAHDAVI@TTIC.EDU
Toyota Technological Institute at Chicago

Adrian Weller   AW665@CAM.AC.UK
University of Cambridge

David Sontag   DSONTAG@CS.NYU.EDU
New York University

Abstract

Structured prediction is used in areas such as computer vision and natural language processing to predict structured outputs such as segmentations or parse trees. In these settings, prediction is performed by MAP inference or, equivalently, by solving an integer linear program. Because of the complex scoring functions required to obtain accurate predictions, both learning and inference typically require the use of approximate solvers. We propose a theoretical explanation to the striking observation that approximations based on linear programming (LP) relaxations are often tight on real-world instances. In particular, we show that learning with LP relaxed inference encourages integrality of training instances, and that tightness generalizes from train to test data.

1. Introduction

Many applications of machine learning can be formulated as prediction problems over structured output spaces (Bakir et al., 2007; Nowozin et al., 2014). In such problems output variables are predicted *jointly* in order to take into account mutual dependencies between them, such as high-order correlations or structural constraints (e.g., matchings or spanning trees). Unfortunately, the improved expressive power of these models comes at a computational cost, and indeed, exact prediction and learning become NP-hard in general. Despite this worst-case intractability, efficient approximations often achieve very good performance in practice. In particular, one type of approximation which has

proved effective in many applications is based on *linear programming (LP) relaxation*. In this approach the prediction problem is first cast as an integer LP (ILP), and then the integrality constraints are relaxed to obtain a tractable program. In addition to achieving high prediction accuracy, it has been observed that LP relaxations are often *tight* in practice. That is, the solution to the relaxed program happens to be optimal for the original hard problem (i.e., an *integral* solution is found). This is particularly surprising since the LPs have complex scoring functions that are not constrained to be from any tractable family. A major open question is to understand why these real-world instances behave so differently from the theoretical worst case.

This paper aims to address this question and to provide a theoretical explanation for the tightness of LP relaxations in the context of structured prediction. In particular, we show that the approximate training objective, although designed to produce accurate predictors, also induces tightness of the LP relaxation as a byproduct. Our analysis also suggests that exact training may have the opposite effect. To explain tightness of *test* instances, we prove a generalization bound for tightness. Our bound implies that if many training instances are integral, then test instances are also likely to be integral. Our results are consistent with previous empirical findings, and to our knowledge provide the first theoretical justification for the wide-spread success of LP relaxations for structured prediction in settings where the training data is not linearly separable.

2. Related Work

Many structured prediction problems can be represented as ILPs (Roth & Yih, 2005; Martins et al., 2009a; Rush et al., 2010). Despite being NP-hard in general (Roth, 1996; Shimony, 1994), various effective approximations have been proposed. Those include both search-based
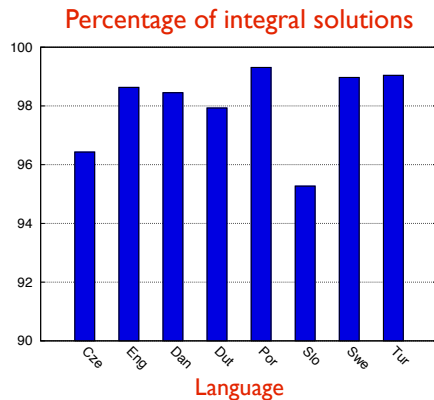
Percentage of integral solutions

Figure 1. Percentage of integral solutions for dependency parsing from Koo et al. (2010).

methods (Daumé III et al., 2009; Zhang et al., 2014), and natural LP relaxations to the hard ILP (Schlesinger, 1976; Koster et al., 1998; Chekuri et al., 2004; Wainwright et al., 2005). Tightness of LP relaxations for special classes of problems has been studied extensively in recent years and include restricting either the structure of the model or its score function. For example, the pairwise LP relaxation is known to be tight for tree-structured models and for supermodular scores (see, e.g., Wainwright & Jordan, 2008; Thapper & Živný, 2012), and the cycle relaxation (equivalently, the second-level of the Sherali-Adams hierarchy) is known to be tight both for planar Ising models with no external field (Barahona, 1993) and for almost balanced models (Weller et al., 2016). To facilitate efficient prediction, one could restrict the model class to be tractable. For example, Taskar et al. (2004) learn supermodular scores, and Meshi et al. (2013) learn tree structures.

However, the sufficient conditions mentioned above are by no means necessary, and indeed, many score functions that are useful in practice do not satisfy them but still produce integral solutions (Roth & Yih, 2004; Sontag et al., 2008; Finley & Joachims, 2008; Martins et al., 2009b; Koo et al., 2010). For example, Martins et al. (2009b) showed that predictors that are learned with LP relaxation yield tight LPs for 92.88% of the test data on a dependency parsing problem (see Table 2 therein). Koo et al. (2010) observed a similar behavior for dependency parsing on a number of languages, as can be seen in Fig. 1 (kindly provided by the authors). The same phenomenon has been observed for a multi-label classification task, where test integrality reached 100% (Finley & Joachims, 2008, Table 3).

Learning structured output predictors from labeled data was proposed in various forms by Collins (2002); Taskar et al. (2003); Tsochantaridis et al. (2004). These formulations generalize training methods for binary classifiers, such as the Perceptron algorithm and support vector machines (SVMs), to the case of structured outputs. The learn-

ing algorithms repeatedly perform prediction, necessitating the use of approximate inference within training as well as at test time. A common approach, introduced right at the inception of structured SVMs by Taskar et al. (2003), is to use LP relaxations for this purpose.

The closest work to ours is Kulesza & Pereira (2007), which showed that not all approximations are equally good, and that it is important to match the inference algorithms used at train and test time. The authors defined the concept of *algorithmic separability* which refers to the case when an approximate inference algorithm achieves zero loss on a data set. The authors studied the use of LP relaxations for structured learning, giving generalization bounds for the true risk of LP-based prediction. However, since the generalization bounds in Kulesza & Pereira (2007) are focused on prediction *accuracy*, the only settings in which tightness on test instances can be guaranteed are when the training data is algorithmically separable, which is seldom the case in real-world structured prediction tasks (the models are far from perfect). Our paper's main result (Theorem 4.1), on the other hand, guarantees that the expected fraction of test instances for which a LP relaxation is *integral* is close to that which was estimated on training data. This then allows us to talk about the generalization of *computation*. For example, suppose one uses LP relaxation-based algorithms that iteratively tighten the relaxation, such as Sontag & Jaakkola (2008); Sontag et al. (2008), and observes that 20% of the instances in the training data are integral using the basic relaxation and that after tightening the remaining 80% are now integral too. Our generalization bound then guarantees that approximately the same ratio will hold at test time (assuming sufficient training data).

Finley & Joachims (2008) also studied the effect of various approximate inference methods in the context of structured prediction. Their theoretical and empirical results also support the superiority of LP relaxations in this setting. Martins et al. (2009b) established conditions which guarantee algorithmic separability for LP relaxed training, and derived risk bounds for a learning algorithm which uses a combination of exact and relaxed inference.

Finally, recently Globerson et al. (2015) studied the performance of structured predictors for 2D grid graphs with binary labels from an information-theoretic point of view. They proved lower bounds on the minimum achievable expected Hamming error in this setting, and proposed a polynomial-time algorithm that achieves this error. Our work is different since we focus on LP relaxations as an approximation algorithm, we handle the most general form without making any assumptions on the model or error measure (except score decomposition), and we concentrate solely on the computational aspects while ignoring any accuracy concerns.

3. Background

In this section we review the formulation of the structured prediction problem, its LP relaxation, and the associated learning problem. Consider a prediction task where the goal is to map a real-valued input vector x to a discrete output vector $y = (y_1, \ldots, y_n)$. A popular model class for this task is based on linear classifiers. In this setting prediction is performed via a linear discriminant rule: $y(x; w) = \text{argmax}_{y'} w^\top \phi(x, y')$, where $\phi(x, y) \in \mathbb{R}^d$ is a function mapping input-output pairs to feature vectors, and $w \in \mathbb{R}^d$ is the corresponding weight vector. Since the output space is often huge (exponential in n), it will generally be intractable to maximize over all possible outputs.

In many applications the score function has a particular structure. Specifically, we will assume that the score decomposes as a sum of simpler score functions: $w^\top \phi(x, y) = \sum_c w_c^\top \phi_c(x, y_c)$, where y_c is an assignment to a (non-exclusive) subset of the variables c. For example, it is common to use such a decomposition that assigns scores to single and pairs of output variables corresponding to nodes and edges of a graph G: $w^\top \phi(x, y) = \sum_{i \in V(G)} w_i^\top \phi_i(x, y_i) + \sum_{ij \in E(G)} w_{ij}^\top \phi_{ij}(x, y_i, y_j)$. Viewing this as a function of y, we can write the prediction problem as: $\max_y \sum_c \theta_c(y_c; x, w)$ (we will sometimes omit the dependence on x and w in the sequel).

Due to its combinatorial nature, the prediction problem is generally NP-hard. Fortunately, efficient approximations have been proposed. Here we will be particularly interested in approximations based on LP relaxations. We begin by formulating prediction as the following ILP:[1]

$$\max_{\substack{\mu \in \mathcal{M}_L \\ \mu \in \{0,1\}^q}} \sum_c \sum_{y_c} \mu_c(y_c)\theta_c(y_c) + \sum_i \sum_{y_i} \mu_i(y_i)\theta_i(y_i) \quad = \theta^\top \mu$$

$$\text{where } \mathcal{M}_L = \left\{ \mu \geq 0 : \begin{array}{ll} \sum_{y_{c \setminus i}} \mu_c(y_c) = \mu_i(y_i) & \forall c, i \in c, y_i \\ \sum_{y_i} \mu_i(y_i) = 1 & \forall i \end{array} \right\}.$$

Here, $\mu_c(y_c)$ is an indicator variable for a factor c and local assignment y_c, and q is the total number of factor assignments (dimension of μ). The set $\mathcal{M}_L$ is known as the local marginal polytope (Wainwright & Jordan, 2008). First, notice that there is a one-to-one correspondence between feasible μ's and assignments y's, which is obtained by setting μ to indicators over local assignments (y_c and y_i) consistent with y. Second, while solving ILPs is NP-hard in general, it is easy to obtain a tractable program by relaxing the integrality constraints ($\mu \in \{0,1\}^q$), which may introduce fractional solutions to the LP. This relaxation is the first level of the Sherali-Adams hierarchy (Sherali & Adams, 1990), which provides successively tighter LP relaxations of an ILP. Notice that since the relaxed program is

obtained by removing constraints, its optimal value upper bounds the ILP optimum.

In order to achieve high prediction accuracy, the parameters w are learned from training data. In this supervised learning setting, the model is fit to labeled examples $\{(x^{(m)}, y^{(m)})\}_{m=1}^M$, where the goodness of fit is measured by a task-specific loss $\Delta(y(x^{(m)}; w), y^{(m)})$. In the *structured SVM* (SSVM) framework (Taskar et al., 2003; Tsochantaridis et al., 2004), the empirical risk is upper bounded by a convex surrogate called the structured hinge loss, which yields the training objective:[2]

$$\min_w \sum_m \max_y \left[w^\top \left(\phi(x^{(m)}, y) - \phi(x^{(m)}, y^{(m)}) \right) + \Delta(y, y^{(m)}) \right].$$
(1)

This is a convex function in w and hence can be optimized in various ways. But, notice that the objective includes a maximization over outputs y for each training example. This loss-augmented prediction task needs to be solved repeatedly during training (e.g., to evaluate subgradients), which makes training intractable in general. Fortunately, as in prediction, LP relaxation can be applied to the structured loss (Taskar et al., 2003; Kulesza & Pereira, 2007), which yields the relaxed training objective:

$$\min_w \sum_m \max_{\mu \in \mathcal{M}_L} \left[\theta_m^\top(\mu - \mu_m) + \ell_m^\top \mu \right], \quad (2)$$

where $\theta_m \in \mathbb{R}^q$ is a score vector in which each entry represents $w_c^\top \phi_c(x^{(m)}, y_c)$ for some c and y_c, similarly $\ell_m \in \mathbb{R}^q$ is a vector with entries[3] $\Delta_c(y_c, y_c^{(m)})$, and μ_m is the integral vector corresponding to $y^{(m)}$.

4. Analysis

In this section we present our main results, proposing a theoretical justification for the observed tightness of LP relaxations for structured prediction. To this end, we make two complementary arguments: in Section 4.1 we argue that optimizing the relaxed training objective of Eq. (2) also has the effect of encouraging tightness of training instances; in Section 4.2 we show that tightness generalizes from train to test data.

4.1. Tightness at Training

We first show that the *relaxed* training objective in Eq. (2), although designed to achieve high accuracy, also induces tightness of the LP relaxation. In order to simplify notation we focus on a single training instance and drop the index

---

[1]For convenience we introduce singleton factors θ_i, which can be set to 0 if needed.

[2]For brevity, we omit the regularization term, however, all of our results below still hold with regularization.

[3]We assume that the task-loss Δ decomposes as the model score.

m. Denote the solutions to the relaxed and integer LPs as:

$$\mu_L \in \underset{\mu \in \mathcal{M}_L}{\operatorname{argmax}} \theta^\top \mu \qquad\qquad \mu_I \in \underset{\substack{\mu \in \mathcal{M}_L \\ \mu \in \{0,1\}^q}}{\operatorname{argmax}} \theta^\top \mu$$

Also, let μ_T be the integral vector corresponding to the ground-truth output $y^{(m)}$. Now consider the following decomposition:

$$\underbrace{\theta^\top (\mu_L - \mu_T)}_{\text{relaxed-hinge}} = \underbrace{\theta^\top (\mu_L - \mu_I)}_{\text{integrality gap}} + \underbrace{\theta^\top (\mu_I - \mu_T)}_{\text{exact-hinge}} \quad (3)$$

This equality states that the difference in scores between the relaxed optimum and ground-truth (*relaxed-hinge*) can be written as a sum of the *integrality gap* and the difference in scores between the exact optimum and the ground-truth (*exact-hinge*) (notice that all terms are non-negative). This simple decomposition has several interesting implications.

First, we can immediately derive the following bound on the integrality gap:

$$\theta^\top (\mu_L - \mu_I) = \theta^\top (\mu_L - \mu_T) - \theta^\top (\mu_I - \mu_T) \quad (4)$$
$$\leq \theta^\top (\mu_L - \mu_T) \quad (5)$$
$$\leq \theta^\top (\mu_L - \mu_T) + \ell^\top \mu_L \quad (6)$$
$$\leq \max_{\mu \in \mathcal{M}_L} \left(\theta^\top (\mu - \mu_T) + \ell^\top \mu \right), \quad (7)$$

where Eq. (7) is precisely the relaxed training objective from Eq. (2). Therefore, optimizing the approximate training objective of Eq. (2) *minimizes an upper bound on the integrality gap*. Hence, driving down the approximate objective also reduces the integrality gap of training instances. One case where the integrality gap becomes zero is when the data is algorithmically separable. In this case the relaxed-hinge term vanishes (the exact-hinge must also vanish), and integrality is assured.

However, the bound above might sometimes be loose. Indeed, to get the bound we have discarded the exact-hinge term (Eq. (5)), added the task-loss (Eq. (6)), and maximized the loss-augmented objective (Eq. (7)). At the same time, Eq. (4) provides a precise characterization of the integrality gap. Specifically, the gap is determined by the difference between the relaxed-hinge and the exact-hinge terms. This implies that even when the relaxed-hinge is not zero, a small integrality gap can still be obtained if the exact-hinge is also large. In fact, the *only way* to get a large integrality gap is by setting the exact-hinge much smaller than the relaxed-hinge. But when can this happen?

A key point is that the relaxed and exact hinge terms are upper bounded by the relaxed and exact *training objectives*, respectively (the latter additionally depend on the task loss Δ). Therefore, minimizing the training objective will also reduce the corresponding hinge term (see also Section 5). Using this insight, we observe that relaxed training reduces

the relaxed-hinge term without directly reducing the exact-hinge term, and thereby induces a small integrality gap. On the other hand, this also suggests that *exact training may actually increase the integrality gap*, since it reduces the exact-hinge without also reducing directly the relaxed-hinge term. This finding is consistent with previous empirical evidence. Specifically, Martins et al. (2009b, Table 2) showed that on a dependency parsing problem, training with the relaxed objective achieved 92.88% integral solutions, while exact training achieved only 83.47% integral solutions. An even stronger effect was observed by Finley & Joachims (2008, Table 3) for multi-label classification, where relaxed training resulted in 99.57% integral instances, with exact training attaining only 17.7% ('Yeast' dataset).

In Section 5 we provide further empirical support for our explanation, however, we next also show its possible limitations by providing a counter-example. The counter-example demonstrates that despite training with a relaxed objective, the exact-hinge can in some cases actually be *smaller* than the relaxed-hinge, leading to a loose relaxation. Although this illustrates the limitations of the explanation above, we point out that the corresponding learning task is far from natural; we believe it is unlikely to arise in real-world applications.

Specifically, we construct a learning scenario where relaxed training obtains zero exact-hinge and non-zero relaxed-hinge, so the relaxation is not tight. Consider a model where $x \in \mathbb{R}^3$, $y \in \{0,1\}^3$, and the prediction is given by:

$$y(x; w) = \underset{y}{\operatorname{argmax}} \Big(x_1 y_1 + x_2 y_2 + x_3 y_3$$
$$+ w \left[\mathbb{1}\{y_1 \neq y_2\} + \mathbb{1}\{y_1 \neq y_3\} + \mathbb{1}\{y_2 \neq y_3\} \right] \Big).$$

The corresponding LP relaxation is then:

$$\max_{\mu \in \mathcal{M}_L} \Big(x_1 \mu_1(1) + x_2 \mu_2(1) + x_3 \mu_3(1) + w[\mu_{12}(01) + \mu_{12}(10)$$
$$+ \mu_{13}(01) + \mu_{13}(10) + \mu_{23}(01) + \mu_{23}(10)] \Big).$$

Next, we construct a trainset where the first instance is: $x^{(1)} = (2,2,2), y^{(1)} = (1,1,0)$, and the second is: $x^{(2)} = (0,0,0), y^{(2)} = (1,1,0)$. It can be verified that $w = 1$ minimizes the relaxed objective (Eq. (2)). However, with this weight vector the relaxed-hinge for the second instance is equal to 1, while the exact-hinge for both instances is 0 (the data is separable w.r.t. $w = 1$). Consequently, there is an integrality gap of 1 for the second instance, and the relaxation is loose (the first instance is actually tight).

Finally, note that Eq. (3) holds for *any integral* μ, and not just the ground-truth μ_T. In other words, the only property of μ_T we are using here is its integrality. Indeed, in Section 5 we verify empirically that training a model using *random*

labels still attains the same level of tightness as training with the ground-truth labels. On the other hand, accuracy drops dramatically, as expected. This analysis suggests that *tightness is not related to accuracy* of the predictor. Finley & Joachims (2008) explained tightness of LP relaxations by noting that fractional solutions always incur a loss during training. Our analysis suggests an alternative explanation, emphasizing the difference in scores (Eq. (4)) rather than the loss, and decoupling tightness from accuracy.

4.2. Generalization of Tightness

Our argument in Section 4.1 concerns only the tightness of train instances. However, the empirical evidence discussed above pertains to test data. To bridge this gap, in this section we show that train tightness implies test tightness. We do so by proving a generalization bound for tightness based on Rademacher complexity.

We first define a loss function which measures the lack of integrality (or, fractionality) for a given instance. To this end, we consider the discrete set of *vertices* of the local polytope $\mathcal{M}_L$ (excluding its convex hull), denoting by $\mathcal{M}^I$ and $\mathcal{M}^F$ the sets of fully-integral and non-integral (i.e., fractional) vertices, respectively (so $\mathcal{M}^I \cap \mathcal{M}^F = \emptyset$, and $\mathcal{M}^I \cup \mathcal{M}^F$ consists of all vertices of $\mathcal{M}_L$). Considering vertices is without loss of generality, since linear programs always have a vertex that is optimal. Next, let $\theta_x \in \mathbb{R}^q$ be the mapping from weights w and inputs x to scores (as used in Eq. (2)), and let $I^*(\theta) = \max_{\mu \in \mathcal{M}^I} \theta^\top \mu$ and $F^*(\theta) = \max_{\mu \in \mathcal{M}^F} \theta^\top \mu$ be the best integral and fractional scores attainable, respectively. By convention, we set $F^*(\theta) = -\infty$ whenever $\mathcal{M}^F = \emptyset$. The fractionality of θ can be measured by the quantity $D(\theta) = F^*(\theta) - I^*(\theta)$. If this quantity is large then the LP has a fractional solution with a much better score than any integral solution. We can now define the loss:

$$\mathcal{L}(\theta) = \begin{cases} 1 & D(\theta) > 0 \\ 0 & \text{otherwise} \end{cases}. \tag{8}$$

That is, the loss equals 1 if and only if the optimal fractional solution has a (strictly) higher score than the optimal integral solution.[4] This loss ignores the ground-truth y, as expected. In addition, we define a ramp loss parameterized by $\gamma > 0$ which upper bounds the fractionality loss:

$$\varphi_\gamma(\theta) = \begin{cases} 0 & D(\theta) \leq -\gamma \\ 1 + D(\theta)/\gamma & -\gamma < D(\theta) \leq 0 \\ 1 & D(\theta) > 0 \end{cases}, \tag{9}$$

For this loss to be zero, the best integral solution has to be better than the best fractional solution by at least γ,

---

[4]Notice that the loss will be 0 whenever the non-integral and integral optima are equal, but this is fine for our purpose, since we consider the relaxation to be tight in this case.

which is a stronger requirement than mere tightness. In Section 4.2.1 we give examples of models that are guaranteed to satisfy this stronger requirement, and in Section 5 we also show this often happens in practice. We point out that $\varphi_\gamma(\theta)$ is generally hard to compute, as is $\mathcal{L}(\theta)$ (due to the discrete optimization involved in computing $I^*(\theta)$ and $F^*(\theta)$). However, here we are only interested in proving that tightness is a generalizing property, so we will not worry about computational efficiency for now. We are now ready to state the main theorem of this section.

Theorem 4.1. *Let inputs be independently selected according to a probability measure $P(X)$, and let Θ be the class of all scoring functions θ_X with $\|w\|_2 \leq B$. Let $\|\phi(x, y_c)\|_2 \leq \hat{R}$ for all x, c, y_c, and q is the total number of factor assignments (dimension of μ). Then for any number of samples M and any $0 < \delta < 1$, with probability at least $1 - \delta$, every $\theta_X \in \Theta$ satisfies:*

$$\mathbb{E}_P[\mathcal{L}(\theta_X)] \leq \hat{\mathbb{E}}_M[\varphi_\gamma(\theta_X)] + O\left(\frac{q^{1.5}B\hat{R}}{\gamma\sqrt{M}}\right) + \sqrt{\frac{8\ln(2/\delta)}{M}}$$
$$(10)$$

where $\hat{\mathbb{E}}_M$ is the empirical expectation.

Proof. Our proof relies on the following general result from Bartlett & Mendelson (2002).

Theorem 4.2 (Bartlett & Mendelson (2002), Theorem 8)**.** *Consider a loss function $\mathcal{L} : \mathcal{Y} \times \Theta \mapsto [0, 1]$ and a dominating function $\varphi : \mathcal{Y} \times \Theta \mapsto [0, 1]$ (i.e., $\mathcal{L}(y, \theta) \leq \varphi(y, \theta)$ for all y, θ). Let $\mathcal{F}$ be a class of functions mapping $\mathcal{X}$ to Θ, and let $\{(x^{(m)}, y^{(m)})\}_{m=1}^M$ be independently selected according to a probability measure $P(x, y)$. Then for any number of samples M and any $0 < \delta < 1$, with probability at least $1 - \delta$, every $f \in \mathcal{F}$ satisfies:*

$$\mathbb{E}[\mathcal{L}(y, f(x))] \leq \hat{\mathbb{E}}_M[\varphi(y, f(x))] + \mathcal{R}_M(\tilde{\varphi} \circ f) + \sqrt{\frac{8\ln(2/\delta)}{M}},$$

where $\hat{\mathbb{E}}_M$ is the empirical expectation, $\tilde{\varphi} \circ f = \{(x, y) \mapsto \varphi(y, f(x)) - \varphi(y, 0) : f \in \mathcal{F}\}$, and $\mathcal{R}_M(\mathcal{F})$ is the Rademacher complexity of the class $\mathcal{F}$.

To use this result, we define $\Theta = \mathbb{R}^q$, $f(x) = \theta_x$, and $\mathcal{F}$ to be the class of all such functions satisfying $\|w\|_2 \leq B$ and $\|\phi(x, y_c)\|_2 \leq \hat{R}$. In order to obtain a meaningful bound, we would like to bound the Rademacher term $\mathcal{R}_M(\tilde{\varphi} \circ f)$. Theorem 12 in Bartlett & Mendelson (2002) states that if $\tilde{\varphi}$ is Lipschitz with constant L and satisfies $\tilde{\varphi}(0) = 0$, then $\mathcal{R}_M(\tilde{\varphi} \circ f) \leq 2L\mathcal{R}_M(\mathcal{F})$. In addition, Weiss & Taskar (2010) show that $\mathcal{R}_M(\mathcal{F}) = O(\frac{qB\hat{R}}{\sqrt{M}})$. Therefore, it remains to compute the Lipschitz constant of $\tilde{\varphi}$, which is equal to the Lipschitz constant of φ. For this purpose, we will bound the Lipschitz constant of $D(\theta)$, and then use $L(\varphi_\gamma(\theta)) \leq L(D(\theta))/\gamma$ (from Eq. (9)).

Let $\mu_I \in \text{argmax}_{\mu \in \mathcal{M}^I} \theta^\top \mu$ and $\mu_F \in$
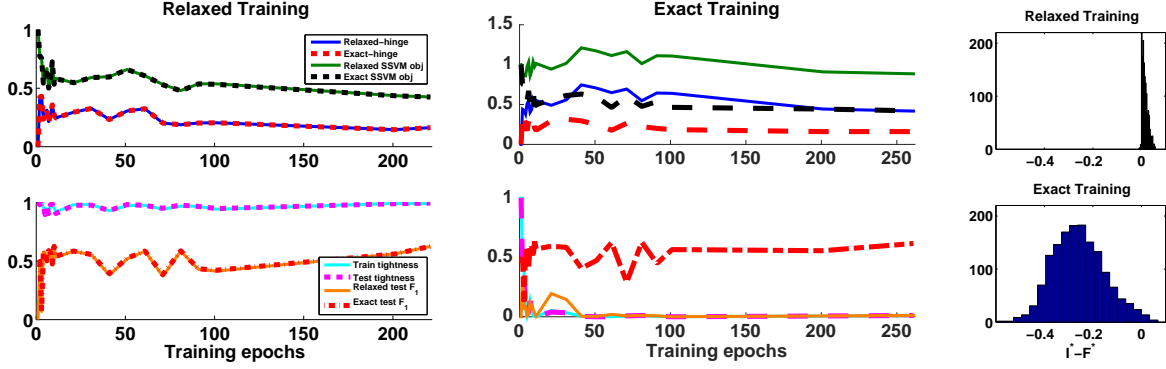
Figure 2. Training with the 'Yeast' dataset. Various quantities of interest are shown as a function of training iterations. (Left) Training with LP relaxation. (Middle) Training with ILP. (Right) Integrality margin (bin widths are scaled differently).

$\operatorname{argmax}_{\mu \in \mathcal{M}^F} \theta^\top \mu$, then:

$$D(\theta^1) - D(\theta^2)$$
$$= (\mu_F^1 - \mu_I^1) \cdot \theta^1 - (\mu_F^2 - \mu_I^2) \cdot \theta^2$$
$$= (\mu_F^1 \cdot \theta^1 - \mu_F^2 \cdot \theta^2) + (\mu_F^1 \cdot \theta^2 - \mu_F^1 \cdot \theta^2)$$
$$\quad + (\mu_I^2 \cdot \theta^2 - \mu_I^1 \cdot \theta^1) + (\mu_I^2 \cdot \theta^1 - \mu_I^2 \cdot \theta^1)$$
$$= \mu_F^1 \cdot (\theta^1 - \theta^2) + (\mu_F^1 - \mu_F^2) \cdot \theta^2$$
$$\quad + \mu_I^2 \cdot (\theta^2 - \theta^1) + (\mu_I^2 - \mu_I^1) \cdot \theta^1$$
$$\leq (\mu_F^1 - \mu_I^2) \cdot (\theta^1 - \theta^2) \qquad \text{[optimality of } \mu_F^2 \text{ and } \mu_I^1]$$
$$\leq \|\mu_F^1 - \mu_I^2\|_2 \|\theta^1 - \theta^2\|_2 \qquad \text{[Cauchy-Schwarz]}$$
$$\leq \sqrt{q} \|\theta^1 - \theta^2\|_2$$

Therefore, $L = \sqrt{q}/\gamma$.

Combining everything together, and dropping the spurious dependence on y, we obtain the bound in Eq. (10). Finally, we point out that when using an L_2 regularizer at training, we can actually drop the assumption $\|w\|_2 \leq B$ and instead use a bound on the norm of the optimal solution (as in the analysis of Shalev-Shwartz et al. (2011)). □

Theorem 4.1 shows that if we observe high integrality (equivalently, low fractionality) on a finite sample of training data, then it is likely that integrality of test data will not be much lower, provided sufficient number of samples.

Our result actually applies more generally to any two disjoint sets of vertices, and is not limited to $\mathcal{M}^I$ and $\mathcal{M}^F$. For example, we can replace $\mathcal{M}^I$ by the set of vertices with at most 10% fractional values, and $\mathcal{M}^F$ by the rest of the vertices of $\mathcal{M}_L$. This gives a different meaning to the loss $\mathcal{L}(\theta)$, and the rest of our analysis holds unchanged. Consequently, our generalization result implies that it is likely to observe a similar portion of instances with at most 10% fractional values at test time as we did at training.

4.2.1. γ-TIGHT RELAXATIONS

In this section we study the stronger notion of tightness required by our surrogate fractionality loss (Eq. (9)), and show examples of models that satisfy it.

Definition An LP relaxation is called γ-*tight* if $I^*(\theta) \geq F^*(\theta) + \gamma$ (so $\varphi_\gamma(\theta) = 0$). That is, the best integral value is larger than the best non-integral value by at least γ.[5]

We focus on binary pairwise models and show two cases where the model is guaranteed to be γ-tight. Proofs are provided in Appendix A. Our first example involves *balanced* models, which are binary pairwise models that have supermodular scores, or can be made supermodular by "flipping" a subset of the variables (for more details, see Appendix A).

Proposition 4.3. *A balanced model with a unique optimum is $(\alpha/2)$-tight, where α is the difference between the best and second-best (integral) solutions.*

This result is of particular interest when learning structured predictors where the edge scores depend on the input. Whereas one could learn supermodular models by enforcing linear inequalities (Taskar et al., 2004), we know of no tractable means of ensuring the model is balanced. Instead, one could learn over the full space of models using LP relaxation. If the learned models are balanced on the training data, Prop. 4.3 together with Theorem 4.1 tell us that the LP relaxation is likely to be tight on test data as well.

Our second example regards models with singleton scores that are much stronger than the pairwise scores. Consider a binary pairwise model[6] in minimal representation, where $\bar{\theta}_i$ are node scores and $\bar{\theta}_{ij}$ are edge scores in this representation (see Appendix A for full details). Further, for each variable i, define the set of neighbors with *attractive* edges $N_i^+ = \{j \in N_i | \bar{\theta}_{ij} > 0\}$, and the set of neighbors with *repulsive* edges $N_i^- = \{j \in N_i | \bar{\theta}_{ij} < 0\}$.

Proposition 4.4. *If all variables satisfy the condition:*

$$\bar{\theta}_i \geq - \sum_{j \in N_i^-} \bar{\theta}_{ij} + \beta, \quad or \quad \bar{\theta}_i \leq - \sum_{j \in N_i^+} \bar{\theta}_{ij} - \beta$$

for some $\beta > 0$, then the model is $(\beta/2)$-tight.

---

[5]Notice that scaling up θ will also increase γ, but our bound in Eq. (10) also grows with the norm of θ (via $B\hat{R}$). Therefore, we assume here that $\|\theta\|_2$ is bounded.

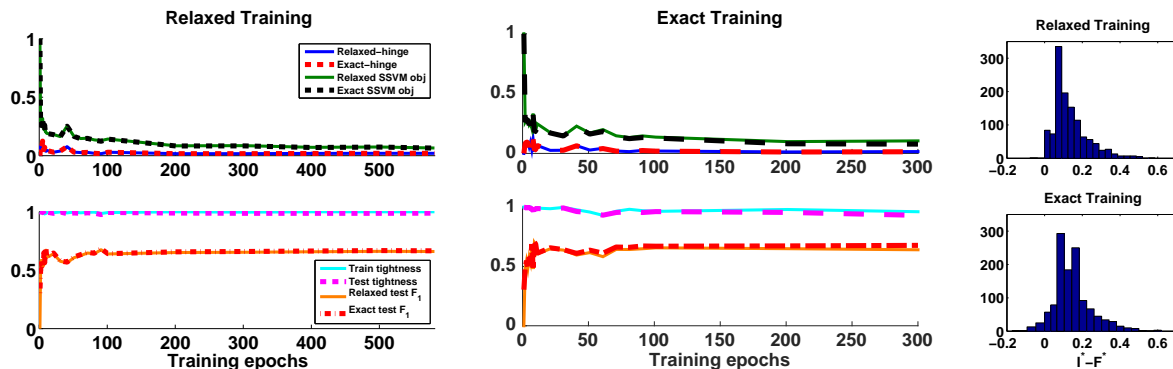[6]This case easily generalizes to non-binary variables.

Figure 3. Training with the 'Scene' dataset. Various quantities of interest are shown as a function of training iterations. (Left) Training with LP relaxation. (Middle) Training with ILP. (Right) Integrality margin.

Finally, we point out that in both of the examples above, the conditions can be verified efficiently and if they hold, the value of γ can be computed efficiently.

5. Experiments

In this section we present some numerical results to support our theoretical analysis. We run experiments for a multi-label classification task and an image segmentation task. For training we have implemented the block-coordinate Frank-Wolfe algorithm for structured SVM (Lacoste-Julien et al., 2013), using GLPK as the LP solver. We use a standard L_2 regularizer, chosen via cross-validation.

Multi-label classification For multi-label classification we adopt the experimental setting of Finley & Joachims (2008). In this setting labels are represented by binary variables, the model consists of singleton and pairwise factors forming a fully connected graph over the labels, and the task loss is the normalized Hamming distance.

Fig. 2 shows relaxed and exact training iterations for the 'Yeast' dataset (14 labels). We plot the relaxed and exact hinge terms (Eq. (3)), the exact and relaxed SSVM training objectives[7] (Eq. (1) and Eq. (2), respectively), fraction of train and test instances having integral solutions, as well as test accuracy (measured by F_1 score). We use a simple scheme to round fractional solutions found with relaxed inference. First, we note that the relaxed-hinge values are nicely correlated with the relaxed training objective, and likewise the exact-hinge is correlated with the exact objective (left and middle, top). Second, observe that with relaxed training, the relaxed-hinge and the exact-hinge are very close (left, top), so the integrality gap, given by their difference, remains small (almost 0 here). On the other hand, with exact training there is a large integrality gap (middle, top). Indeed, we can see that the percentage of

integral solutions is almost 100% for relaxed training (left, bottom), and close to 0% with exact training (middle, bottom). In Fig. 2 (right) we also show histograms of the difference between the optimal integral and fractional values, i.e., the integrality margin ($I^*(\theta) - F^*(\theta)$), under the final learned model for all training instances. It can be seen that with relaxed training this margin is positive (although small), while exact training results in larger negative values. Finally, we note that train and test integrality levels are very close to each other, almost indistinguishable (left and middle, bottom), which provides empirical support to our generalization result from Section 4.2.

We next train a model using random labels (with similar label counts as the true data). In this setting the learned model obtains 100% tight training instances (not shown), which supports our observation that any integral point can be used in place of the ground-truth, and that accuracy is not important for tightness. Finally, in order to verify that tightness is not coincidental, we test the tightness of the relaxation induced by a random weight vector w. We find that random models are never tight (in 20 trials), which shows that tightness of the relaxation does not come by chance.

We now proceed to perform experiments on the 'Scene' dataset (6 labels). The results, in Fig. 3, differ from the 'Yeast' results in case of exact training (middle). Specifically, we observe that in this case the relaxed-hinge and exact-hinge are close in value (middle, top), as for relaxed training (left, top). As a consequence, the integrality gap is very small and the relaxation is tight for almost all train (and test) instances. These results illustrate that sometimes optimizing the exact objective can reduce the relaxed objective (and relaxed-hinge) as well. Further, in this setting we observe a larger integrality margin (right), namely the integral optimum is strictly better than the fractional one.

We conjecture that the LP instances are easy in this case

---

[7]The displayed objective values are averaged over train instances and exclude regularization.
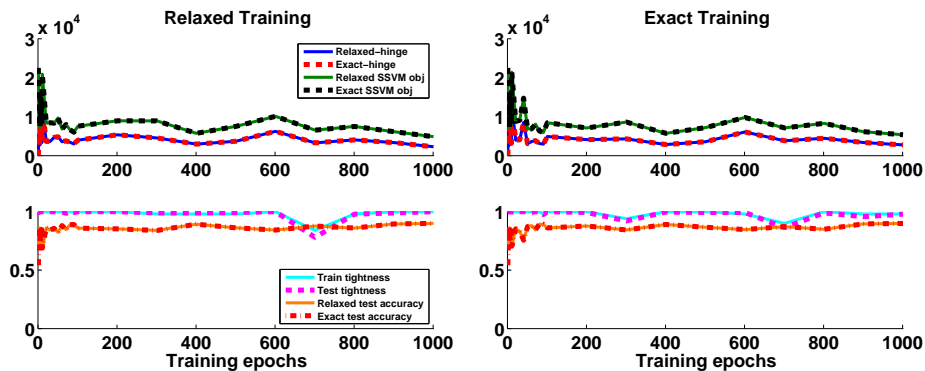
Figure 4. Training for foreground-background segmentation with the Weizmann Horse dataset. Various quantities of interest are shown as a function of training iterations. (Left) Training with LP relaxation. (Right) Training with ILP.

due to the dominance of the singleton scores.[8] Specifically, the features provide a strong signal which allows label assignment to be decided mostly based on the local score, with little influence coming from the pairwise terms. To test this conjecture we inject Gaussian noise into the input features, forcing the model to rely more on the pairwise interactions. We find that with the noisy singleton scores the results are indeed more similar to the 'Yeast' dataset, where a large integrality gap is observed and fewer instances are tight (see Appendix B in the supplement).

Image segmentation   Finally, we conduct experiments on a foreground-background segmentation problem using the Weizmann Horse dataset (Borenstein et al., 2004). The data consists of 328 images, of which we use the first 50 for training and the rest for testing. Here a binary output variable is assigned to each pixel, and there are $\sim 58K$ variables per image on average. We extract singleton and pairwise features as described in Domke (2013). Fig. 4 shows the same quantities as in the multi-label setting, except for the accuracy measure – here we compute the percentage of correctly classified pixels rather than F_1. We observe a very similar behavior to that of the 'Scene' multi-label dataset (Fig. 3). Specifically, both relaxed and exact training produce a small integrality gap and high percentage of tight instances. Unlike the 'Scene' dataset, here only 1.2% of variables satisfy the condition in Prop. 4.4 (using LP training). In all of our experiments the learned model scores were never balanced (Prop. 4.3), although for the segmentation problem we believe the models learned are close to balanced, both for relaxed and exact training.

6. Conclusion

In this paper we propose an explanation for the tightness of LP relaxations often observed in structured prediction

applications. Our analysis is based on a careful examination of the integrality gap and its relation to the training objective. It shows how training with LP relaxations, although designed with accuracy considerations in mind, also induces tightness of the relaxation. Our derivation also suggests that exact training may sometimes have the opposite effect, increasing the integrality gap.

To explain tightness of test instances, we show that tightness generalizes from train to test instances. Compared to the generalization bound of Kulesza & Pereira (2007), our bound only considers the tightness of the instance, ignoring label errors. Thus, for example, if learning happens to settle on a set of parameters in a tractable regime (e.g., supermodular potentials or stable instances (Makarychev et al., 2014)) for which the LP relaxation is tight for all training instances, our generalization bound guarantees that with high probability the LP relaxation will also be tight on test instances. In contrast, in Kulesza & Pereira (2007), tightness on test instances can only be guaranteed when the training data is algorithmically separable (i.e., LP-relaxed inference predicts perfectly).

Our work suggests many directions for further study. Our analysis in Section 4.1 focuses on the score hinge, and it would be interesting to also study the effect of various task losses Δ on tightness of the relaxation at training. Next, our bound in Section 4.2 is intractable to compute due to the hardness of the surrogate loss φ. It is therefore desirable to derive a tractable alternative which could be used to obtain a useful guarantee in practice. The upper bound on integrality shown in Section 4.1 holds for other convex relaxations which have been proposed for structured prediction, such as semi-definite programming relaxations (Kumar et al., 2009). However, it is less clear how to extend the generalization result to such non-polyhedral relaxations. Finally, we hope that our methodology will be useful for shedding light on tightness of convex relaxations in other learning problems.

---

[8]With ILP training, the condition in Prop. 4.4 is satisfied for 65% of all variables, although only 1% of the training instances satisfy it for all their variables.

References

Bakir, G. H., Hofmann, T., Schölkopf, B., Smola, A. J., Taskar, B., and Vishwanathan, S. V. N. *Predicting Structured Data.* The MIT Press, 2007.

Barahona, F. On cuts and matchings in planar graphs. *Mathematical Programming*, 60:53–68, 1993.

Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. *The Journal of Machine Learning Research*, 3:463–482, 2002.

Borenstein, E., Sharon, E., and Ullman, S. Combining top-down and bottom-up segmentation. In *CVPR*, 2004.

Chekuri, C., Khanna, S., Naor, J., and Zosin, L. A linear programming formulation and approximation algorithms for the metric labeling problem. *SIAM J. on Discrete Mathematics*, 18 (3):608–625, 2004.

Collins, M. Discriminative training methods for hidden Markov models: Theory and experiments with perceptron algorithms. In *EMNLP*, 2002.

Daumé III, Hal, Langford, John, and Marcu, Daniel. Search-based structured prediction. *Machine Learning*, 75(3):297–325, 2009.

Domke, J. Learning graphical model parameters with approximate marginal inference. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(10), 2013.

Finley, T. and Joachims, T. Training structural SVMs when exact inference is intractable. In *Proceedings of the 25th International Conference on Machine learning*, pp. 304–311, 2008.

Globerson, A., Roughgarden, T., Sontag, D., and Yildirim, C. How hard is inference for structured prediction? In *ICML*, 2015.

Koo, T., Rush, A. M., Collins, M., Jaakkola, T., and Sontag, D. Dual decomposition for parsing with non-projective head automata. In *EMNLP*, 2010.

Koster, A., van Hoesel, S.P.M., and Kolen, A.W.J. The partial constraint satisfaction problem: Facets and lifting theorems. *Operations Research Letters*, 23:89–97, 1998.

Kulesza, A. and Pereira, F. Structured learning with approximate inference. In *Advances in Neural Information Processing Systems 20*, pp. 785–792. 2007.

Kumar, M. P., Kolmogorov, V., and Torr, P. H. S. An analysis of convex relaxations for MAP estimation of discrete MRFs. *JMLR*, 10:71–106, 2009.

Lacoste-Julien, S., Jaggi, M., Schmidt, M., and Pletscher, P. Block-coordinate Frank-Wolfe optimization for structural SVMs. In *ICML*, pp. 53–61, 2013.

Makarychev, Konstantin, Makarychev, Yury, and Vijayaraghavan, Aravindan. Bilulinial stable instances of max cut and minimum multiway cut. *Proc. 22nd Symposium on Discrete Algorithms (SODA)*, 2014.

Martins, A., Smith, N., and Xing, E. P. Concise integer linear programming formulations for dependency parsing. In *ACL*, 2009a.

Martins, A., Smith, N., and Xing, E. P. Polyhedral outer approximations with application to natural language parsing. In *Proceedings of the 26th International Conference on Machine Learning*, 2009b.

Meshi, O., Eban, E., Elidan, G., and Globerson, A. Learning max-margin tree predictors. In *UAI*, 2013.

Nowozin, Sebastian, Gehler, Peter V., Jancsary, Jeremy, and Lampert, Christoph. *Advanced Structured Prediction*. MIT Press, 2014.

Roth, D. On the hardness of approximate reasoning. *Artificial Intelligence*, 82, 1996.

Roth, D. and Yih, W. A linear programming formulation for global inference in natural language tasks. In *CoNLL, The 8th Conference on Natural Language Learning*, 2004.

Roth, D. and Yih, W. Integer linear programming inference for conditional random fields. In *ICML*, pp. 736–743. ACM, 2005.

Rush, Alexander M., Sontag, David, Collins, Michael, and Jaakkola, Tommi. On dual decomposition and linear programming relaxations for natural language processing. In *EMNLP*, 2010.

Schlesinger, M. I. Syntactic analysis of two-dimensional visual signals in noisy conditions. *Kibernetika*, 4:113–130, 1976.

Shalev-Shwartz, S., Singer, Y., Srebro, N., and Cotter, A. Pegasos: Primal estimated sub-gradient solver for svm. *Mathematical programming*, 127(1):3–30, 2011.

Sherali, H. D. and Adams, W. P. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM J. on Disc. Math.*, 3(3): 411–430, 1990.

Shimony, Y. Finding the MAPs for belief networks is NP-hard. *Aritifical Intelligence*, 68(2):399–410, 1994.

Sontag, D. and Jaakkola, T. New outer bounds on the marginal polytope. In Platt, J.C., Koller, D., Singer, Y., and Roweis, S. (eds.), *Advances in Neural Information Processing Systems 20*, pp. 1393–1400. MIT Press, Cambridge, MA, 2008.

Sontag, D., Meltzer, T., Globerson, A., Jaakkola, T., and Weiss, Y. Tightening LP relaxations for MAP using message passing. In *UAI*, pp. 503–510, 2008.

Taskar, B., Guestrin, C., and Koller, D. Max-margin Markov networks. In *Advances in Neural Information Processing Systems*. MIT Press, 2003.

Taskar, B., Chatalbashev, V., and Koller, D. Learning associative Markov networks. In *Proc. ICML*. ACM Press, 2004.

Thapper, J. and Živný, S. The power of linear programming for valued CSPs. In *FOCS*, 2012.

Tsochantaridis, I., Hofmann, T., Joachims, T., and Altun, Y. Support vector machine learning for interdependent and structured output spaces. In *ICML*, pp. 104–112, 2004.

Wainwright, M. and Jordan, M. I. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers Inc., Hanover, MA, USA, 2008.

Wainwright, M., Jaakkola, T., and Willsky, A. MAP estimation via agreement on trees: message-passing and linear programming. *IEEE Transactions on Information Theory*, 51(11): 3697–3717, 2005.

Weiss, D. and Taskar, B. Structured Prediction Cascades. In *AISTATS*, 2010.

Weller, Adrian. Bethe and related pairwise entropy approximations. In *Uncertainty in Artificial Intelligence (UAI)*, 2015.

Weller, Adrian, Rowland, Mark, and Sontag, David. Tightness of LP relaxations for almost balanced models. In *AISTATS*, 2016.

Zhang, Yuan, Lei, Tao, Barzilay, Regina, and Jaakkola, Tommi. Greed is good if randomized: New inference for dependency parsing. In *EMNLP*, 2014.