Learning Submodular Losses with the Lovász Hinge

Jiaqian Yu JIAQIAN.YU@CENTRALESUPELEC.FR
Matthew B. Blaschko MATTHEW.BLASCHKO@INRIA.FR
CentraleSupélec & Inria, Grande Voie des Vignes, 92295 Châtenay-Malabry, France

Abstract

Learning with non-modular losses is an important problem when sets of predictions are made simultaneously. The main tools for constructing convex surrogate loss functions for set prediction are margin rescaling and slack rescaling. In this work, we show that these strategies lead to tight convex surrogates iff the underlying loss function is increasing in the number of incorrect predictions. However, gradient or cutting-plane computation for these functions is NP-hard for non-supermodular loss functions. We propose instead a novel convex surrogate loss function for submodular losses, the Lovász hinge, which leads to $\mathcal{O}(p \log p)$ complexity with $\mathcal{O}(p)$ oracle accesses to the loss function to compute a gradient or cutting-plane. As a result, we have developed the first tractable convex surrogates in the literature for submodular losses. We demonstrate the utility of this novel convex surrogate through a real world image labeling task.

1. Introduction

Statistical learning has largely addressed problems in which a loss function decomposes over individual training samples. However, there are many circumstances in which non-modular losses can lead to better end results. This is the case when multiple outputs of a prediction system are used as the basis of a decision making process that leads to a single real-world outcome. These dependencies in the effect of the predictions (rather than a statistical dependency between predictions themselves) are therefore properly incorporated into a learning system through a non-modular loss function. In this paper, we aim to provide a theoretical and algorithmic foundation for a novel class of learning algorithms that make feasible learning with submodular losses, an important subclass of non-modular losses that is currently infeasible with existing algorithms.

Convex surrogate loss functions are central to the practical application of empirical risk minimization. Straightforward principles have been developed for the design of convex surrogates for binary classification and regression (Bartlett et al., 2006), and in the structured output setting margin and slack rescaling are two principles for defining convex surrogates for more general output spaces (Tsochantaridis et al., 2005). Despite the apparent flexibility of margin and slack rescaling in their ability to bound arbitrary loss functions, there are fundamental limitations to our ability to apply these methods in practice: (i) they provide only loose upper bounds to certain loss functions, (ii) computing a gradient or cutting plane is NP-hard for submodular loss functions, and (iii) consistency results are lacking in general (McAllester, 2007; Tewari & Bartlett, 2007). In practice, modular losses, such as Hamming loss, are often applied to maintain tractability, although non-modular losses, such as the intersection over union loss have been applied in the structured prediction setting (Blaschko & Lampert, 2008; Nowozin, 2014).

Non-modular losses have been (implicitly) considered in the context of multilabel classification problems. (Cheng et al., 2010) uses the Hamming loss and subset 0-1 loss which are modular, and a rank loss which is supermodular; (Petterson & Caetano, 2011) introduces submodular pairwise potentials, not submodular loss functions, while using a non-submodular loss based on F-score. (Li & Lin, 2014) uses (weighted) Hamming loss which is modular, but also proposes a new tree-based algorithm for training; (Doppa et al., 2014) uses modular losses e.g. Hamming loss and F1 loss which is non-submodular. However, non-supermodular loss functions are substantially more rare in the literature.

In this work, we introduce an alternate principle to construct convex surrogate loss functions for submodular losses based on the Lovász extension of a set function. The Lovász extension of a submodular function is its convex closure, and has been used in other machine learning contexts e.g. (Bach, 2010; Iyer & Bilmes, 2013). We analyze the settings in which margin and slack rescaling are tight

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

convex surrogates by finding necessary and sufficient conditions for the surrogate function to be an extension of a set function. Although margin and slack rescaling generate extensions of *some* submodular set functions, their optimization is NP-hard. We therefore propose a novel convex surrogate for submodular functions based on the Lovász extension, which we call the Lovász hinge. In contrast to margin and slack rescaling, the Lovász hinge provides a tight convex surrogate to *all* submodular loss functions, and computation of a gradient or cutting plane can be achieved in $\mathcal{O}(p \log p)$ time with a linear number of oracle accesses to the loss function. We demonstrate empirically fast convergence of a cutting plane optimization strategy applied to the Lovász hinge, and show that optimization of a submodular loss results in lower average loss on the test set.

In Section 2 we introduce the notion of a submodular loss function in the context of empirical risk minimization. The Structured Output SVM is one of the most popular objectives for empirical risk minimization of interdependent outputs, and we demonstrate its properties on non-modular loss functions in Section 3. In Section 4 we introduce the Lovász hinge, and we empirically demonstrate its performance on an image labeling task on the Microsoft COCO dataset in Section 5.

2. Submodular Loss Functions

In empirical risk minimization (ERM), we approximate the risk, $\mathcal{R}$ of a prediction function $f : \mathcal{X} \mapsto \mathcal{Y}$ by an empirical sum over losses incurred on a finite sample, using e.g. an i.i.d. sampling assumption (Vapnik, 1995):

$$\hat{\mathcal{R}}(f) := \frac{1}{n} \sum_{i=1}^{n} \Delta(y_i, f(x_i)) \tag{1}$$

Central to the practical application of the ERM principle, one must approximate, or upper bound the discrete loss function Δ with a convex surrogate. We will identify the creation of a convex surrogate for a specific loss function with an operator that maps a function with a discrete domain to one with a continuous domain. In particular, we will study the case that the discrete domain is a set of p binary predictions. In this case we denote

$$\mathcal{Y} = \{-1, +1\}^p \tag{2}$$
$$f(x) = \text{sign}(g(x)) \tag{3}$$
$$\Delta : \{-1, +1\}^p \times \{-1, +1\}^p \mapsto \mathbb{R}_+ \tag{4}$$
$$\mathbf{B}\Delta : \{-1, +1\}^p \times \mathbb{R}^p \mapsto \mathbb{R} \tag{5}$$

where $\mathbf{B}$ is an operator that constructs the surrogate loss function from Δ, and $g : \mathcal{X} \mapsto \mathbb{R}^p$ is a parametrized prediction function to be optimized by ERM.

A key property is the relationship between $\mathbf{B}\Delta$ and Δ. In particular, we are interested in when a given surrogate strategy $\mathbf{B}\Delta$ yields an *extension* of Δ. We make this notion formal by identifying $\{-1, +1\}^p$ with a given p-dimensional unit hypercube of $\mathbb{R}^p$ (cf. Definition 3). We say that $\mathbf{B}\Delta(y, \cdot)$ is an extension of $\Delta(y, \cdot)$ iff the functions are equal over the vertices of this unit hypercube. We focus on function extensions as they ensure a tight relationship between the discrete loss and the convex surrogate.

2.1. Set Functions and Submodularity

For many optimization problems, a function defined on the power set of a given base set V, a *set function* is often taken into consideration to be minimized (or maximized). Submodular functions play an important role among these set functions, similar to convex functions on vector spaces.

Submodular functions may be defined through several equivalent properties. We use the following definition (Fujishige, 2005):

Definition 1. *A set function $l : \mathcal{P}(V) \mapsto \mathbb{R}$ is **submodular** if and only if for all subsets $A, B \subseteq V$, $l(A) + l(B) \geq l(A \cup B) + l(A \cap B)$.*

A function is *supermodular* iff its negative is submodular, and a function is modular iff it is both submodular and supermodular. A modular function can be written as a dot product between a binary vector in $\{0, 1\}^p$ (where $p = |V|$) encoding a subset of V and a coefficient vector in $\mathbb{R}^p$ which uniquely identifies the modular function. By example, Hamming loss is a modular function with a coefficient vector of all ones, and a subset defined by the entries that differ between two vectors.

Necessary to the sequel is the notion of monotone set functions

Definition 2. *A set function $l : \mathcal{P}(V) \mapsto \mathbb{R}$ is **increasing** if and only if for all subsets $A \subset V$ and elements $x \in V \setminus A$, $l(A) \leq l(A \cup \{x\})$.*

In this paper, we consider loss functions for multiple outputs that are set functions where inclusion in a set is defined by a corresponding prediction being incorrect:

$$\Delta(y, \tilde{y}) = l(\{i | y^i \neq \tilde{y}^i\}) \tag{6}$$

for some set function l. Such functions are typically increasing, though it is possible to conceive of a sensible loss function that may be non-increasing.

With these notions, we now turn to an analysis of margin and slack rescaling, and show necessary and sufficient conditions for these operators to yield an extension to the underlying discrete loss function.

3. Existing Convex Surrogates

A general problem is to learn a mapping f from inputs $x \in \mathcal{X}$ to discrete outputs (labels) $y \in \mathcal{Y}$. The Structured Output SVM (SOSVM) is a popular framework for doing so in the regularized risk minimization framework. The approach that SOSVM pursues is to learn a function $h : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}$ over input/output pairs from which a prediction can be derived by maximizing $f(x) = \arg\max_y h(x, y)$ over the response from a given input x. The SOSVM framework assumes h to be represented by an inner product between an element of a reproducing kernel Hilbert space (RKHS) and some combined feature representation of inputs and outputs $\phi(x, y)$,

$$h(x, y; w) = \langle w, \phi(x, y) \rangle \tag{7}$$

although the notions of margin and slack rescaling may be applied to other function spaces, including random forests and deep networks.

A bounded loss function $\Delta : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ quantifies the loss associated with a prediction $\tilde{y}$ while the true value is y, and is used to re-scale the constraints. The margin-rescaling constraints and slack-rescaling constraints are:

$$\min_{w,\xi} \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n} \xi_i, \quad \forall i, \forall \tilde{y} \in \mathcal{Y} : \tag{8}$$

$$\langle w, \phi(x_i, y_i) \rangle - \langle w, \phi(x_i, \tilde{y}) \rangle \geq \Delta(y_i, \tilde{y}) - \xi_i \tag{9}$$

$$\text{or} \quad \langle w, \phi(x_i, y_i) \rangle - \langle w, \phi(x_i, \tilde{y}) \rangle \geq 1 - \frac{\xi_i}{\Delta(y_i, \tilde{y})} \tag{10}$$

respectively.

In the sequel, we will consider the case that each $x_i \in \mathcal{X}$ is an ordered set of p elements in a RKHS, $\mathcal{H}$, and that $y_i \in \mathcal{Y}$ is a binary vector in $\{-1, +1\}^p$. We consider feature maps such that

$$\langle w, \phi(x_i, y_i) \rangle = \sum_{j=1}^{p} \langle w^j, x_i^j \rangle y_i^j. \tag{11}$$

Given this family of joint feature maps, we may identify the jth dimension of g (cf. Equation (3)) with $\langle w^j, x_i^j \rangle$. Therefore $\arg\max_y h(x, y; w) = \text{sign}(g(x))$ and $h(x, y) = \langle g(x), y \rangle$.

3.1. Analysis of Margin and Slack Rescaling

We now turn to the problem of determining necessary and sufficient conditions for margin and slack rescaling to yield an extension of the underlying set loss function. We denote the operators for margin and slack rescaling that map the loss function to its convex surrogate $\mathbf{M}$ and $\mathbf{S}$, respectively. These operators have the same signature as $\mathbf{B}$ in Equation (5).

$$\mathbf{M}\Delta(y, g(x)) := \max_{\tilde{y} \in \mathcal{Y}} \Delta(y, \tilde{y}) + \langle g(x), \tilde{y} \rangle - \langle g(x), y \rangle \tag{12}$$

$$\mathbf{S}\Delta(y, g(x)) := \max_{\tilde{y} \in \mathcal{Y}} \Delta(y, \tilde{y})\big(1 + \langle g(x), \tilde{y} \rangle - \langle g(x), y \rangle\big) \tag{13}$$

respectively.

In order to analyse whether $\mathbf{M}\Delta$ and $\mathbf{S}\Delta$ are extensions of Δ, we construct a mapping to a p-dimensional vector space using following definition:

Definition 3. *A convex surrogate function $\mathbf{B}\Delta(y, \cdot)$ is an extension when $\mathbf{B}\Delta(y, \cdot) = \Delta(y, \cdot)$ on the vertices of the 0-1 unit cube under the mapping to $\mathbb{R}^p$ (cf. Fig. 1):*

$$i = \{1, \ldots, p\}, \quad [u]^i = 1 - \langle w, x^i \rangle y^i \tag{14}$$

In the sequel, we will use the notation for $l : \mathcal{P}(V) \mapsto \mathbb{R}$ as in Equation (6). Note that at the vertices, $\Delta(y, \cdot)$ has the following values:

$$l(\varnothing) \text{ at } \mathbf{0}_p \tag{15}$$

$$l(\mathbf{I}) \text{ at } \{v | v \in \{0, 1\}, v_i = 1 \Leftrightarrow i \in \mathbf{I}\} \tag{16}$$

We call (16) the value of l at *the vertex* $\mathbf{I}$.

3.2. Slack rescaling

Proposition 1. *$\mathbf{S}\Delta(y, \cdot)$ is an extension of a set function $\Delta(y, \cdot)$ iff $\Delta(y, \cdot)$ is an increasing function.*

Proof. First we demonstrate the necessity. Given $\mathbf{S}\Delta(y, \cdot)$ an extension of $\Delta(y, \cdot)$, we analyse whether $\Delta(y, \cdot)$ is an increasing function. Specifically, there are two cases to determine the values of $\mathbf{S}\Delta(y, \cdot)$:

1. if $u = \mathbf{0}$, $\mathbf{S}\Delta(y, g(x)) = l(\varnothing)$ according to Equation (6), where u is defined as in Equation (14).
2. if $u \in \mathbb{R}^p \setminus \{\mathbf{0}\}$, let $\mathbf{I} = \{i | u^i \neq 0\}$, then according to Equation (13), $\mathbf{S}\Delta(y, g(x))$ takes the value of the following equation:

$$\max_{\mathbf{I} \in \mathcal{P}(V)} l(\mathbf{I})\big(1 - \sum_{i \in \mathbf{I}} \langle w, x^i \rangle y^i\big) \tag{17}$$

As $\mathbf{S}\Delta(y, g(x))$ is an extension of $\Delta(y, \text{sign}(g(x)))$, by Definition 3, $\mathbf{S}\Delta(y, g(x)) = \Delta(y, \text{sign}(g(x)))$ at the vertices as in (15) and (16). Note that it is trivial when $u = \mathbf{0}$ so the first case is always true for arbitrary (including increasing) l.

Considering the second case when $\mathbf{S}\Delta(y, g(x))$ is equal to Equation (17), let $\mathbf{I}_2 = \arg\max_{\mathbf{I} \in \mathcal{P}(V)} l(\mathbf{I})\big(1 -$

$\sum_{i\in\mathbf{I}}\langle w, x^i\rangle y^i\big)$. As $\mathbf{S}\Delta(y, g(x))$ is an extension, $\mathbf{S}\Delta(y, g(x)) = \Delta(y, \text{sign}(g(x)))$ at the vertex $\mathbf{I}_2$.

$$\forall \mathbf{I}_1 \in \mathcal{P}(V) \setminus \{\emptyset\},\ l(\mathbf{I}_2) \geq l(\mathbf{I}_1)\,(1 - |(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1|) \tag{18}$$

This leads also to two cases,

1. if $|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1| = 0$, $(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1 = \emptyset$, which implies $\mathbf{I}_1 \subseteq \mathbf{I}_2$, then from Equation (18) we get $l(\mathbf{I}_2) \geq l(\mathbf{I}_1)$. This implies that l and therefore Δ are increasing;

2. if $|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1| \geq 1$, this means the rhs of Equation (18) is negative, then it turns out to be redundant with $l(\mathbf{I}_2) \geq 0$ which is always true.

To conclude, given $\mathbf{S}\Delta(y, \cdot)$ is an extension of a set function $\Delta(y, \cdot)$, it is always the case that Δ is increasing.

To demonstrate the sufficiency, we need to show Equation (18) is always true if l is increasing. We note that (18) holds if $|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1| = 0$, because if $\mathbf{I}_1 \subseteq \mathbf{I}_2$ then $l(\mathbf{I}_2) \geq l(\mathbf{I}_1)$. Then if $|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1| \geq 1$, (18) always holds even for arbitrary l. We conclude that $\mathbf{S}\Delta(y, g(x)) = \Delta(y, \text{sign}(g(x)))$ at the vertices when Equation (16) holds. As for the case of Equation (15), it is trivial as $u = \mathbf{0}$. So $\mathbf{S}\Delta(y, \cdot)$ yields a extension of $\Delta(y, \cdot)$ if $\Delta(y, \cdot)$ is increasing. □

3.3. Margin rescaling

It is a necessary, but not sufficient condition that $\Delta(y, \tilde{y})$ be increasing for margin rescaling to yield an extension. However, we note that for all increasing $\Delta(y, \tilde{y})$ there exists a positive scaling $\gamma \in \mathbb{R}$ such that margin rescaling yields an extension. This is an important result for regularized risk minimization as we may simply rescale Δ to guarantee that margin rescaling yields an extension, and simultaneously scale the regularization parameter such that the relative contribution of the regularizer and loss is unchanged at the vertices of the unit cube.

Proposition 2. *For all increasing set functions l such that $\exists y$ for which $\mathbf{M}\Delta(y, \cdot)$ is not an extension of $\Delta(y, \cdot)$, we can always find a positive scale factor γ specific to l such that margin rescaling yields an extension. We denote $\mathbf{M}\gamma\Delta$ and $\gamma\Delta$ as the rescaled functions.*

Proof. Similar to Proposition 1, we analyse two cases to determine the values of $\mathbf{M}\gamma\Delta(y, g(x))$:

1. if $u = \mathbf{0}$, $\mathbf{M}\gamma\Delta(y, g(x)) = \gamma l(\varnothing)$ where u is defined as in Equation (14). It is typically the case that $l(\varnothing) = 0$, but this is not a technical requirement.

2. if $u \neq \mathbf{0}$, let $\mathbf{I} = \{i | u^i \neq 0\}$, then $\mathbf{M}\gamma\Delta(y, g(x))$ takes the value of the following equation:

$$\max_{\mathbf{I}\in\mathcal{P}(V)}\ \gamma l(\mathbf{I}) - \sum_{i\in\mathbf{I}}\langle w, x^i\rangle y^i \tag{19}$$

To satisfy Definition 3, we must find a $\gamma > 0$ such that $\mathbf{M}\gamma\Delta(y, g(x)) = \gamma\Delta(y, \text{sign}(g(x)))$ at the vertices. Note that it is trivial when $u = \mathbf{0}$ so the first case is true for arbitrary $\gamma > 0$.

For the second case as in Equation (19), let $\mathbf{I}_2 = \arg\max_{\mathbf{I}\in\mathcal{P}(V)} \big(l(\mathbf{I}) - \sum_{i\in\mathbf{I}}\langle w, x^i\rangle y^i\big)$. We have $\mathbf{M}\gamma\Delta(y, g(x)) = \Delta(y, \text{sign}(g(x)))$ at the vertices $\mathbf{I}_2$ according to the extension. The scale factor should satisfy:

$$\forall \mathbf{I}_1 \in \mathcal{P}(V)\setminus\{\emptyset\},\ \gamma\big(l(\mathbf{I}_2) - l(\mathbf{I}_1)\big) \geq -|(V\setminus\mathbf{I}_2)\cap\mathbf{I}_1| \tag{20}$$

which leads to the following cases:

1. if $|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1| = 0$, we have $(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1 = \emptyset$, which implies $\mathbf{I}_1 \subseteq \mathbf{I}_2$. Equation (20) reduces to

$$\gamma\big(l(\mathbf{I}_2) - l(\mathbf{I}_1)\big) \geq 0 \tag{21}$$

   and l is an increasing function so $l(\mathbf{I}_1) \leq l(\mathbf{I}_2)$ and Equation (21) is always true as $\gamma > 0$.

2. if $|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1| \neq 0$, we need to discuss the relationship between $l(\mathbf{I}_1)$ and $l(\mathbf{I}_2)$:

   (a) if $l(\mathbf{I}_2) = l(\mathbf{I}_1)$, then Equation (21) becomes $0 \geq -|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1|$, for which the rhs is negative so it is always true.

   (b) if $l(\mathbf{I}_2) > l(\mathbf{I}_1)$, then

$$\gamma \geq \frac{-|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1|}{l(\mathbf{I}_2) - l(\mathbf{I}_1)} \tag{22}$$

   for which the rhs is negative so it is redundant with $\gamma > 0$.

   (c) if $l(\mathbf{I}_2) < l(\mathbf{I}_1)$, then

$$\gamma \leq \frac{-|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1|}{l(\mathbf{I}_2) - l(\mathbf{I}_1)} \tag{23}$$

   for which the rhs is *strictly* positive so it becomes an upper bound on γ .

In summary the scale factor γ should satisfy the following constraint for an increasing loss function l:

$$\forall \mathbf{I}_1, \mathbf{I}_2 \in \mathcal{P}(V) \setminus \{\emptyset\}, 0 < \gamma \leq \frac{-|(V \setminus \mathbf{I}_2) \cap \mathbf{I}_1|}{l(\mathbf{I}_2) - l(\mathbf{I}_1)}$$

Finally, we note that the rightmost ratio is always strictly positive. □

3.4. Complexity of subgradient computation

Although we have proven that slack and margin rescaling yield extensions to the underlying discrete loss under fairly general conditions, their key shortcoming is in the complexity of the computation of subgradients for submodular losses. The subgradient computation for slack and margin rescaling requires the computation of $\arg\max_{\tilde{y}} \Delta(y, \tilde{y})(1 + h(x, \tilde{y}) - h(x, y))$ and $\arg\max_{\tilde{y}} \Delta(y, \tilde{y}) + h(x, \tilde{y})$, respectively. Both of these functions require the maximization of the loss, which corresponds to supermodular minimization in the case of a submodular loss. This computation is NP-hard, and such loss functions are not feasible with these existing methods in practice. Furthermore, approximate inference, e.g. based on (Nemhauser et al., 1978), leads to poor convergence when used to train a structured output SVM resulting in a high error rate (Finley & Joachims, 2008). We therefore introduce the Lovász hinge as an alternative operator to construct feasible convex surrogates for submodular losses.

4. Lovász Hinge

We now construct a convex surrogate for submodular losses. This surrogate is based on a fundamental result by Lovász relating submodular sets and piecewise linear convex functions. The Lovász extension allows the extension of a set-function defined on the vertices of the hypercube $\{0,1\}^p$ to the full hypercube $[0,1]^p$ (Lovász, 1983):

Definition 4. *The Lovász extension $\hat{l}$ of a set function l, $\hat{l} : [0,1]^p \to \mathbb{R}$, is defined as follows: for $s^\pi \in [0,1]^p$ with decreasing components $s^{\pi_1} \geq s^{\pi_2} \geq \cdots \geq s^{\pi_p}$ ordered by a permutation $\pi = (\pi_1, \pi_2, \cdots, \pi_p)$, $\hat{l}(s)$ is defined as:*

$$\hat{l}(s) = \sum_{j=1}^{p} s^{\pi_j} \left(l\left(\{\pi_1, \cdots, \pi_j\}\right) - l\left(\{\pi_1, \cdots, \pi_{j-1}\}\right)\right) \tag{24}$$

It has been proven that a set-function l is submodular iff its Lovász extension is convex (Lovász, 1983). Based on this definition, we propose our novel convex surrogate for submodular functions:

Definition 5. *The Lovász hinge, $\mathbf{L}$, is defined as the unique operator such that, for l submodular*

$$\mathbf{L}\Delta(y, g(x)) := \tag{25}$$
$$\left(\max_{\pi} \sum_{j=1}^{p} s^{\pi_j} \left(l\left(\{\pi_1, \cdots, \pi_j\}\right) - l\left(\{\pi_1, \cdots, \pi_{j-1}\}\right)\right)\right)_+$$

where $(\cdot)_+ = \max(\cdot, 0)$, π is a permutation,

$$s^{\pi_j} = 1 - g^{\pi_j}(x) y^{\pi_j}, \tag{26}$$

and $g^{\pi_j}(x)$ is the π_jth dimension of $g(x)$.

If l is increasing, when $s_i^{\pi_j}$ becomes negative we may threshold the corresponding components to zero as in standard hinge loss. By thresholding negative $s_i^{\pi_j}$ to zero, we have that the Lovász hinge coincides exactly with slack rescaling in the special case of a modular loss, and coincides with an SVM in the case of Hamming loss. In the case that l is non-increasing, the thresholding strategy no longer coincides with the Lovász extension over the unit cube, and will not yield an extension. We therefore will not apply thresholding in the non-increasing case, but we still have that $\mathbf{L}\Delta \geq 0$ and is convex.[1]

As the computation of a cutting plane or loss gradient is precisely the same procedure as computing a value of the Lovász extension, we have the same computational complexity, which is $\mathcal{O}(p \log p)$ to sort the p coefficients, followed by $\mathcal{O}(p)$ oracle accesses to the loss function (Lovász, 1983). This is precisely an application of the greedy algorithm to optimize a linear program over the submodular polytope (Edmonds, 1971). In our implementation, we have employed a one-slack cutting-plane optimization with ℓ_2 regularization analogous to (Joachims et al., 2009). We observe empirical convergence of the primal-dual gap at a rate comparable to that of a structured output SVM (Fig. 3).

4.1. Visualization of convex surrogates

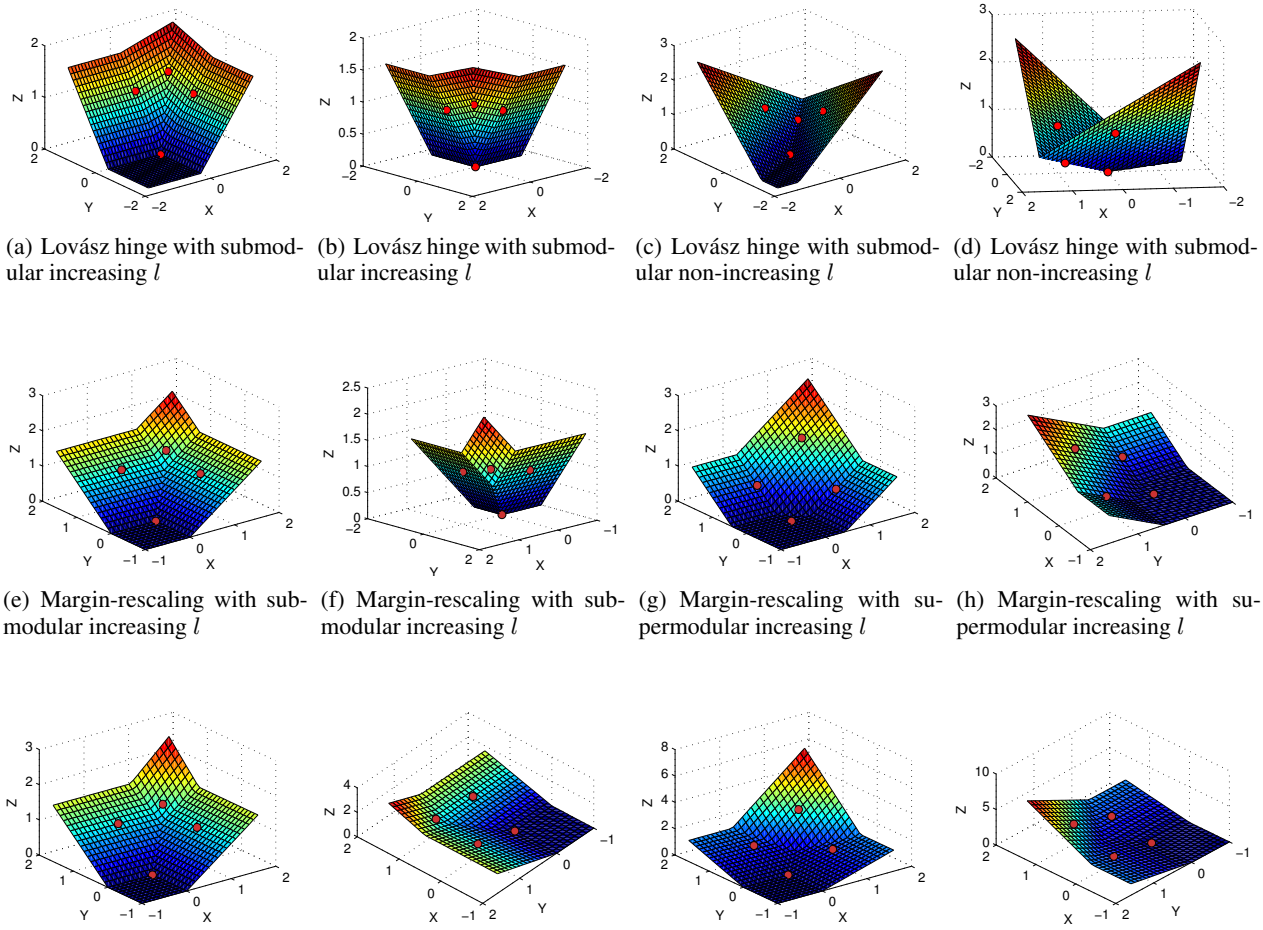
For visualization of the loss surfaces, in this section we consider a simple binary classification problem with two elements with a non-modular loss function l:

$$\mathcal{X} := \mathbb{R}^{d \times 2} \qquad \mathcal{Y} := \{-1, +1\}^2$$

Then with different values for $l(\varnothing)$, $l(\{1\})$, $l(\{2\})$ and $l(\{1,2\})$, we can have different modularity or monotonicity properties of the function l which then defines Δ. We illustrate the Lovász hinge, slack and margin rescaling for the following cases: (i) submodular increasing: $l(\varnothing) = 0$, $l(\{1\}) = l(\{2\}) = 1$, $l(\{1,2\}) = 1.2$, (ii) submodular non-increasing: $l(\varnothing) = 0$, $l(\{1\}) = l(\{2\}) = 1$, $l(\{1,2\}) = 0.4$, and (iii) supermodular increasing: $l(\varnothing) = 0$, $l(\{1\}) = l(\{2\}) = 1$, $l(\{1,2\}) = 2.8$.

In Fig. 1, the x axis represents the value of s_i^1, the y axis represents the value of s_i^2 in Eq (26), and the z axis is the convex loss function given by Equation (12), Equation (13), and Definition 5, respectively. We plot the values of l as solid dots at the vertices of the hypercube. We observe that all the solid dots *touch* the surfaces, which empirically validates that the surrogates are extensions of the discrete loss. Here we set l as symmetric functions, while the extensions can be also validated for asymmetric increasing set functions.

[1] Source code is available for download at https://sites.google.com/site/jiaqianyu08/lovaszhinge.



(a) Lovász hinge with submodular increasing l (b) Lovász hinge with submodular increasing l (c) Lovász hinge with submodular non-increasing l (d) Lovász hinge with submodular non-increasing l

(e) Margin-rescaling with submodular increasing l (f) Margin-rescaling with submodular increasing l (g) Margin-rescaling with supermodular increasing l (h) Margin-rescaling with supermodular increasing l

(i) Slack-rescaling with submodular increasing l (j) Slack-rescaling with submodular increasing l (k) Slack-rescaling with supermodular increasing l (l) Slack-rescaling with supermodular increasing l

Figure 1. We introduce a novel convex surrogate for submodular losses, the Lovász hinge. We show here the Lovász hinge, margin and slack rescaling surfaces with different loss functions l from different views; the x and y axes represent the value of s_i^1 and s_i^2 in Eq. (26); the z axis represents the value of the convex surrogate; the solid red dots represent the values of l at the vertices of the unit hypercube. The convex surrogate strategies yield extensions of the discrete loss.

5. Experimental Results

In this section, we consider a task in which multiple labels need to be predicted simultaneously and for which a submodular loss over labels is to be minimized. If a subsequent prediction task focuses on detecting a *dinner scene*, the initial multi-label prediction should emphasize that all labels e.g. *people*, *dinning tables*, *forks*, and *knifes* must be correct *within a given image*. This contrasts with a traditional multi-label prediction task in which a loss function decomposes over the individual predictions. The misprediction of a single label, e.g. *person*, will preclude the chance to predict correctly the combination of all labels. This corresponds exactly to the property of diminishing returns of a submodular function. Fig. 2 shows example images from the Microsoft COCO dataset (Lin et al., 2014).

While using classic modular losses such as 0-1 loss, the classifier is trained to minimize the sum of incorrect predictions, so the complex interaction between label mispredictions is not considered. In this work, we use a new submodular loss function and apply the Lovász hinge to enable efficient convex risk minimization.

Microsoft COCO The Microsoft COCO dataset (Lin et al., 2014) is an image recognition, segmentation, and captioning dataset. It contains more than 70 categories, more than 300,000 images and around 5 captions per image. We have used frequent itemset mining (Uno et al., 2004) to determine the most common combination of cat-

(a) (b) (c)

Figure 2. Examples from the Microsoft COCO dataset. Fig. 2(a) contains all the categories of interest (cf. Section 5); Fig. 2(b) contains *dining table*, *fork* and *cup*; Fig. 2(c) is not a dining scene but contains *people*.

egories in an image. For sets of size 6, these are: *person*, *cup*, *fork*, *knife*, *chair* and *dining table*.

For the experiments, we repeatedly sample sets of images containing k ($k = 0, 1, 2, \cdots, 6$) categories. The training/validation set and testing set have the same distribution. For a single iteration, we sample 1050 images for the training/validation set including 150 images each for sets containing k ($k = 0, 1, 2, \cdots, 6$) of the target labels. More than 12,000 images from the entire dataset are sampled at least once as we repeat the experiments to compute statistical significance.

We use Overfeat (Sermanet et al., 2014) to extract image features following the procedure of (Razavian et al., 2014). Overfeat has been trained for the image classification task on ImageNet ILSVRC 2013, and has achieved good performance on a range of image classification problems including the Microsoft COCO dataset.

Submodular Losses In this task, we first define a submodular loss function as follows:

$$\Delta(y, \tilde{y}) := 1 - \exp\left(-\alpha|\mathbf{I}|\right) \tag{27}$$

where $\mathbf{I}$ is the set of mispredicted labels. $1 - \exp\left(-|\mathbf{I}|\right)$, is a concave function depending only on the size of $\mathbf{I}$, as a consequence it is a submodular function. The empirical results with this submodular loss are shown in Table 1 and Fig. 3(a).

We have carried out experiments with another submodular loss function:

$$\Delta(y, \tilde{y}) := 1 - \exp\left(-|\mathbf{I}|\right) + \langle \beta, (1 - y \odot \tilde{y})/2 \rangle \tag{28}$$

where $\odot$ is the Hadamard product. The final summand is a modular function that penalizes labels proportionate to the coefficient vector $\beta > 0$. In the experiments, we order the category labels as *person*, *dining table*, *chair*, *cup*, *fork*, and *knife*. We set $\beta = [1\ 0.8\ 0.7\ 0.6\ 0.5\ 0.4]^T$ according to the size of the object. The results with this submodular loss are shown in Table 2 and Fig. 3(b). We also train and test with the 0-1 loss, which is equivalent to an SVM.

We compare different losses employed during training and during testing. m_{sub} and s_{sub} denote the use of the submodular loss with margin and slack rescaling, respectively. As this optimization is NP-hard, we have employed the simple application of the greedy approach as is common in (non-monotone) submodular maximization (e.g. (Krause & Golovin, 2014)).

Results We repeated each experiment 10 times with random sampling in order to obtain an estimate of the average performance. Table 1 and Table 2 show the cross comparison of average loss values (with standard error) using different loss functions during training and during testing for the COCO dataset. As predicted by theory, training with the same loss function as used during testing yields the best results. Slack and margin rescaling fail due to the necessity of approximate inference, which results in a poor discriminant function. By contrast, the Lovász hinge yields the best performance when the submodular loss is used to evaluate the test predictions. We do not expect that optimizing the submodular loss should give the best performance when the 0-1 loss is used to evaluate the test predictions. Indeed in this case, the Lovász hinge trained on 0-1 loss corresponds with the best performing system.

Fig. 3(a) and Fig. 3(b) show for the two experiments the primal-dual gap as a function of the number of cutting-plane iterations using the Lovász hinge with submodular loss, as well as for a SVM (labeled 0-1), and margin and slack rescaling (labeled m_{sub} and s_{sub}). This demonstrates that the empirical convergence of the Lovász hinge is at a rate comparable to an SVM, and is feasible to optimize in practice for real-world problems.

		testing loss	
		l_{sub}	0-1
training loss	l_{sub}	$\mathbf{0.7908 \pm 0.0113}$	1.3035±0.0182
	0-1	0.7941±0.0102	$\mathbf{1.2863 \pm 0.0175}$
	s_{sub}	0.8739±0.0133	1.4397±0.0219
	m_{sub}	0.8722±0.0115	1.4365±0.0206

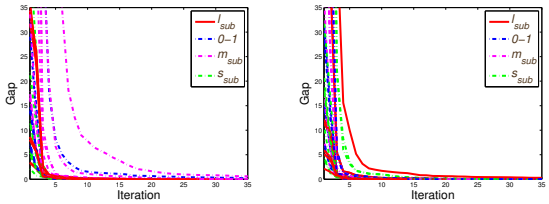
Table 1. Comparison of average loss values (with standard error) using different loss functions during training and during testing. l_{sub} is as in Equation (27). Training with the same loss function as used during testing yields the best results. Slack and margin rescaling fail due to the necessity of approximate inference, which results in a poor discriminant function. The Lovász hinge trained on the appropriate loss yields the best performance in both cases.

6. Discussion & Conclusions

In this work, we have introduced a novel convex surrogate loss function, the Lovász hinge, which makes tractable for the first time learning with submodular loss functions. In contrast to margin and slack rescaling, computation of

		testing loss	
		l_{sub}	0-1
training loss	l_{sub}	$\mathbf{1.3767 \pm 0.0143}$	1.3003±0.0176
	0-1	1.3813±0.0135	$\mathbf{1.2975 \pm 0.0152}$
	s_{sub}	1.4711±0.0153	1.3832±0.0156
	m_{sub}	1.4811±0.0117	1.4016±0.0136

Table 2. The cross comparison of average loss values (with standard error) using different loss functions during training and during testing. l_{sub} is as in Equation (28) (cf. comments for Table. 1).



(a) (b)

Figure 3. The primal-dual gap as a function of the number of cutting-plane iterations using the Lovász hinge with submodular loss, a SVM (labeled 0-1), and margin and slack rescaling with greedy inference (labeled m_{sub} and s_{sub}). Fig. 3(a) for the experiment using Equation (27) and Fig. 3(b) for Equation (28). This demonstrates that empirical convergence of the Lovász hinge is at a rate comparable to an SVM, and is feasible to optimize in practice for real-world problems.

the gradient or cutting plane can be achieved in $\mathcal{O}(p \log p)$ time. Margin and slack rescaling are NP-hard to optimize in this case.

We have proven necessary and sufficient conditions for margin and slack rescaling to yield tight convex surrogates to a discrete loss function. These conditions are that the discrete loss be a (properly scaled) increasing function. However, it may be of interest to consider non-increasing functions in some domains. The Lovász hinge can be applied also to non-increasing functions.

We have demonstrated the correctness and utility of the Lovász hinge on a natural image categorization task. We have shown that training by minimization of the Lovász hinge applied to multiple submodular loss functions results in a lower empirical test error than existing methods, as one would expect from a correctly defined convex surrogate. Slack and margin rescaling both fail in practice as approximate inference does not yield a good approximation of the discriminant function. The causes of this have been studied in a different context in (Finley & Joachims, 2008), but are effectively due to (i) repeated approximate inference compounding errors, and (ii) erroneous early termination due to underestimation of the primal objective. We empirically observe that the Lovász hinge delivers much better performance by contrast, and makes no approximations in the subgradient computation. Exact inference should yield a good predictor for slack and margin rescaling, but sub-exponential optimization only exists if P=NP. Therefore, the Lovász hinge is the *only* polynomial time option in the literature for learning with such losses.

The introduction of this novel strategy for constructing convex surrogate loss functions for submodular losses points to many interesting areas for future research. Among them are then definition and characterization of useful loss functions in specific application areas. Furthermore, theoretical convergence results for a cutting plane optimization strategy are of interest.

Acknowledgements

This work is partially funded by ERC Grant 259112, and FP7-MC-CIG 334380. The first author is supported by a fellowship from the China Scholarship Council.

References

Bach, Francis R. Structured sparsity-inducing norms through submodular functions. In *Advances in Neural Information Processing Systems*, pp. 118–126, 2010.

Bartlett, Peter L., Jordan, Michael I., and McAuliffe, Jon D. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

Blaschko, Matthew B. and Lampert, Christoph H. Learning to localize objects with structured output regression. In Forsyth, David, Torr, Philip, and Zisserman, Andrew (eds.), *European Conference on Computer Vision*, volume 5302 of *Lecture Notes in Computer Science*, pp. 2–15. 2008.

Cheng, Weiwei, Hüllermeier, Eyke, and Dembczynski, Krzysztof J. Bayes optimal multilabel classification via probabilistic classifier chains. In *Proceedings of the International Conference on Machine Learning*, pp. 279–286, 2010.

Doppa, Janardhan Rao, Yu, Jun, Ma, Chao, Fern, Alan, and Tadepalli, Prasad. HC-search for multi-label prediction: An empirical study. In *Proceedings of AAAI Conference on Artificial Intelligence*, 2014.

Edmonds, Jack. Matroids and the greedy algorithm. *Mathematical programming*, 1(1):127–136, 1971.

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

Fujishige, Satoru. *Submodular functions and optimization*. Elsevier, 2005.

Iyer, Rishabh and Bilmes, Jeff. The Lovász-Bregman divergence and connections to rank aggregation, clustering, and web ranking: Extended version. *Uncertainity in Artificial Intelligence*, 2013.

Joachims, Thorsten, Finley, Thomas, and Yu, Chun-Nam. Cutting-plane training of structural SVMs. *Machine Learning*, 77(1):27–59, 2009.

Krause, Andreas and Golovin, Daniel. Submodular function maximization. In Bordeaux, Lucas, Hamadi, Youssef, and Kohli, Pushmeet (eds.), *Tractability: Practical Approaches to Hard Problems*. Cambridge University Press, 2014.

Li, Chun-Liang and Lin, Hsuan-Tien. Condensed filter tree for cost-sensitive multi-label classification. In *Proceedings of the 31st International Conference on Machine Learning*, pp. 423–431, 2014.

Lin, Tsung-Yi, Maire, Michael, Belongie, Serge, Hays, James, Perona, Pietro, Ramanan, Deva, Dollár, Piotr, and Zitnick, C. Lawrence. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014.

Lovász, László. Submodular functions and convexity. In *Mathematical Programming The State of the Art*, pp. 235–257. Springer, 1983.

McAllester, David. Generalization bounds and consistency for structured labeling. In *Predicting Structured Data*. MIT Press, 2007.

Nemhauser, George L., Wolsey, Laurence A., and Fisher, Marshall L. An analysis of approximations for maximizing submodular set functions–I. *Mathematical Programming*, 14(1):265–294, 1978.

Nowozin, Sebastian. Optimal decisions from probabilistic models: The intersection-over-union case. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014.

Petterson, James and Caetano, Tibério S. Submodular multi-label learning. In *Advances in Neural Information Processing Systems*, pp. 1512–1520, 2011.

Razavian, Ali Sharif, Azizpour, Hossein, Sullivan, Josephine, and Carlsson, Stefan. CNN features off-the-shelf: An astounding baseline for recognition. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 512–519, 2014.

Sermanet, Pierre, Eigen, David, Zhang, Xiang, Mathieu, Michael, Fergus, Rob, and LeCun, Yann. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *International Conference on Learning Representations*, 2014.

Tewari, Ambuj and Bartlett, Peter L. On the consistency of multiclass classification methods. *The Journal of Machine Learning Research*, 8:1007–1025, 2007.

Tsochantaridis, Ioannis, Joachims, Thorsten, Hofmann, Thomas, and Altun, Yasemin. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6(9):1453–1484, 2005.

Uno, Takeaki, Asai, Tatsuya, Uchida, Yuzo, and Arimura, Hiroki. An efficient algorithm for enumerating closed patterns in transaction databases. In *Discovery Science*, pp. 16–31. Springer, 2004.

Vapnik, Vladimir N. *The Nature of Statistical Learning Theory*. Springer, 1995.