Learning Energy Networks with Generalized Fenchel-Young Losses

Anonymous Author(s)
Affiliation
Address
email

Abstract

Energy-based models, a.k.a. energy networks, perform inference by optimizing an energy function, typically parametrized by a neural network. This allows one to capture potentially complex relationships between inputs and outputs. To learn the parameters of the energy function, the solution to that optimization problem is typically fed into a loss function. The key challenge for training energy networks lies in computing loss gradients, as this typically requires argmin/argmax differentiation. In this paper, building upon a generalized notion of conjugate function, which replaces the usual bilinear pairing with a general energy function, we propose generalized Fenchel-Young losses, a natural loss construction for learning energy networks. Our losses enjoy many desirable properties and their gradients can be computed efficiently without argmin/argmax differentiation. We also prove the calibration of their excess risk in the case of linear-concave energies. We demonstrate our losses on multilabel classification and imitation learning tasks.

1 Introduction

Training a neural network usually involves finding its parameters by minimizing a loss function, which captures how well the network fits the data. A typical example is the multiclass logistic loss (a.k.a. cross-entropy loss) from logistic regression, which is the canonical loss associated with the softmax output layer and the categorical distribution. If we replace the softmax with another output layer, what loss function should we use instead? In generalized linear models [51, 48], which include logistic regression and Poisson regression as special cases, the negative log-likelihood gives a loss function associated with a link function, generalizing the softmax to other members of the exponential family [8]. More generally, **Fenchel-Young losses** [18] provide a generic way to construct a canonical convex loss function if the associated output layer can be written in a certain argmax form. Besides the aforementioned generalized linear models, models that fall in this family include sparsemax [47], the structured perceptron [27] and conditional random fields [43, 68]. However, the theory of Fenchel-Young losses is currently limited to argmax output layers that use a **bilinear** pairing.

To increase expressivity, energy-based models [44], a.k.a. **energy networks**, perform inference by optimizing an **energy function**, typically parametrized by a neural network. This leads to an inner optimization problem, which can capture potentially complex relationships between inputs and outputs. A similar approach is taken in SPENs (Structured Prediction Energy Networks) [11, 13], in which a continuous relaxation of the inner optimization problem is used, amenable to projected gradient descent or mirror descent. In input-convex neural networks [4], energy functions are restricted to be convex, so as to make the inner optimization problem easy to solve optimally. To learn the parameters of the energy function, the solution of the inner optimization problem is typically fed into a loss function, leading to an outer optimization problem. In order to solve that problem by stochastic gradient descent, the main challenge lies in computing the loss gradients. Indeed, when using an arbitrary loss function, by the chain rule, computing gradients requires differentiating through the inner optimization problem solution, often referred to as **argmin** or **argmax differentiation**. This

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

can be done by backpropagation through unrolled algorithm iterates [13] or implicit differentiation through the optimality conditions [4]. This issue can be circumvented by using a generalized perceptron loss [44] or a max margin loss [11]. However, these losses lack strong motivation and theoretical guarantees for learning energy networks.

In this paper, we propose to extend the theory of Fenchel-Young losses [18], so as to create a canonical loss function associated with an energy network. Our proposal builds upon two main ingredients. First, we use **generalized conjugate functions**, also known as **Fenchel-Moreau conjugates** [49, 61], allowing us to go beyond bilinear pairings and to support **general energy functions** such as neural networks. Second, we use **envelope theorems** to obtain easy-to-compute gradients, without argmin or argmax differentiation. We provide novel guarantees on the **calibration** of the excess risk, when our loss is used as a surrogate for another discrete loss. To sum up, we obtain a well-motivated loss construction for general energy networks. The rest of the paper is organized as follows.

- After providing some background (§2), we describe **regularized energy networks**, identify a classification of energy functions and give several possible examples that fall in this family (§3).
- We define **generalized conjugates**, an extension of Legendre-Fenchel conjugates, and state their properties (§4). We establish novel conditions for a generalized conjugate to be a smooth function.
- We then introduce **generalized Fenchel-Young losses** and show that they enjoy many of the same favorable properties as the regular Fenchel-Young losses (§5). On the theoretical side, we prove **calibration** of the excess risk for linear-concave energies and strongly-convex regularizers (§6).
- We demonstrate our losses on **multilabel classification** and **imitation learning** tasks (§7).

2 Background

Convex conjugates. Let $\mathcal{C} \subseteq \mathbb{R}^k$ be an output set. Let $\Omega \colon \mathcal{C} \to \mathbb{R}$ be a function, such that $\Omega(p) = \infty$ for all $p \notin \mathcal{C}$, i.e., $\mathrm{dom}(\Omega) = \mathcal{C}$. Given $v \in \mathcal{V} \subseteq \mathbb{R}^k$, we define the convex conjugate [22] of Ω, also known as the Legendre-Fenchel transform of Ω, by

$$\Omega^*(v) \coloneqq \max_{p \in \mathcal{C}} \ \langle v, p \rangle - \Omega(p). \tag{1}$$

The conjugate Ω^* is always convex, even if Ω is not. We define the corresponding argmax as

$$p_\Omega(v) \coloneqq \operatorname*{argmax}_{p \in \mathcal{C}} \ \langle v, p \rangle - \Omega(p) = \nabla \Omega^*(v). \tag{2}$$

The latter equality follows from Danskin's theorem [29, 15], under the assumption that Ω is strictly convex (otherwise, we obtain a subgradient). It is well-known that Ω^* is $\frac{1}{\gamma}$-smooth w.r.t. the dual norm $\|\cdot\|_*$ if and only if Ω is γ-strongly convex w.r.t. the norm $\|\cdot\|$ [37, 39, 10, 74].

Fenchel-Young losses. Suppose that $v = g_\theta(x) \in \mathbb{R}^k$ are the logits / scores produced by a neural network g, where $x \in \mathcal{X} \subseteq \mathbb{R}^d$ and $\theta \in \Theta$ are the network's input features and parameters, respectively. What loss function should we use if we want to use (2) as output layer? The Fenchel-Young loss [18] generated by $\Omega \colon \mathcal{C} \to \mathbb{R}$ provides a natural solution. It is defined by

$$L_\Omega(v, y) \coloneqq \Omega^*(v) + \Omega(y) - \langle v, y \rangle, \tag{3}$$

where $y \in \mathcal{Y} \subseteq \mathcal{C}$ is the ground-truth label. Earlier instances of this loss were independently proposed in the contexts of ranking [2] and multiclass classification [30]. Among many useful properties, this loss satisfies $L_\Omega(v, y) \geq 0$ and $L_\Omega(v, y) = 0 \Leftrightarrow y = p_\Omega(v)$ if Ω is strictly convex [18]. In that sense, it is the canonical loss associated with (2). Interestingly, many existing loss functions are recovered as special cases. For instance, if $\mathcal{C} = \mathbb{R}^k$ and $\Omega(p) = \frac{1}{2}\|p\|_2^2$, which is 1-strongly convex w.r.t. $\|\cdot\|_2$ over $\mathbb{R}^k$, then we obtain the self-dual, the identity mapping and the squared loss:

$$\Omega^*(v) = \frac{1}{2}\|v\|_2^2, \quad p_\Omega(v) = v, \quad L_\Omega(v, y) = \frac{1}{2}\|v - y\|_2^2.$$

If $\mathcal{C}$ is the probability simplex $\triangle^k \coloneqq \{p \in \mathbb{R}_+^k \colon \sum_{i=1}^k p_i = 1\}$ and $\Omega(p)$ is the scaled Shannon negentropy $\gamma\langle p, \log p \rangle$, which is γ-strongly convex w.r.t. $\|\cdot\|_1$ over $\triangle^k$, then we obtain

$$\Omega^*(v) = \mathrm{LSE}^\gamma(v), \quad p_\Omega(v) = \mathrm{softmax}^\gamma(v), \quad L_\Omega(v, y) = \mathrm{KL}(y, \mathrm{softmax}^\gamma(v)),$$

where we used the log-sum-exp $\mathrm{LSE}^\gamma(v) \coloneqq \gamma \log(\sum_{i=1}^k \exp(v_i/\gamma))$, $\mathrm{softmax}^\gamma(v) \propto \exp(v/\gamma)$, and the Kullback-Leibler divergence. More generally, when $\mathcal{C} = \mathrm{conv}(\mathcal{Y})$, the convex hull of $\mathcal{Y}$, p_Ω corresponds to a projection, for which efficient algorithms exist in numerous cases [18, 16]. The calibration of the excess risk of Fenchel-Young losses when Ω is strongly convex was established in [55, 16]. However, the theory of Fenchel-Young losses is currently limited to the bilinear pairing $\langle v, p \rangle$, which restricts their expressivity and scope.

3 Regularized energy networks

Energy networks. Also known as energy-based models or EBMs [44], of which SPENs [11, 13] are a particular case, these networks compute predictions by solving an optimization problem of the form

$$p^\Phi(v) \coloneqq \operatorname*{argmax}_{p \in \mathcal{C}} \ \Phi(v, p),$$

where $v = g_\theta(x) \in \mathcal{V}$ is the energy network's input, $\Phi(v, p)$ is a scalar-valued energy function, and $\mathcal{C}$ is an output set. Throughout this paper, we use the convention that higher energy indicates higher degree of compatibility between v and the prediction p. Since $\Phi(v, p)$ is a general energy function, we emphasize that v and p do not need to have the same dimensions, unlike with the bilinear pairing $\langle v, p \rangle$. Any neural network $g_\theta(x) \in \mathbb{R}^k$ can be written in energy network form, since $\operatorname{argmax}_{p \in \mathbb{R}^k} -\|g_\theta(x) - p\|_2^2 = g_\theta(x)$. The key advantage of energy networks, however, is their ability to capture complex interactions between inputs and outputs. This is especially useful in the structured prediction setting [6], where predictions are made of sub-parts, such as sequences being composed of individual elements.

Introducing regularization. In this paper, we compute predictions by solving

$$p_\Omega^\Phi(v) \coloneqq \operatorname*{argmax}_{p \in \mathcal{C}} \ \Phi(v, p) - \Omega(p), \tag{4}$$

where we further added a regularization function $\Omega \colon \mathcal{C} \to \mathbb{R}$. We call $\Phi(v, p) - \Omega(p)$ a **regularized energy function** and we call the corresponding model a **regularized energy network**. We obviously recover usual energy networks by setting Ω to the indicator function of the set $\mathcal{C}$, i.e., $\Omega(p) = 0$ if $p \in \mathcal{C}$, ∞ otherwise. While it is in principle possible to absorb Ω into Φ, keeping Ω explicit has several advantages: 1) it allows to introduce generalized conjugate functions and their properties (§4) 2) it allows to introduce generalized Fenchel-Young losses (§5), which mirror the original Fenchel-Young losses 3) we obtain closed forms for (4) in certain cases (Table 1).

Solving the maximization problem. The ability to solve the maximization problem in (4) efficiently depends on the properties of $\Phi(v, p)$ w.r.t. p. If $\Phi(v, p)$ is **linear** in p, for instance $\Phi(v, p) = \langle v, Up \rangle$, then we obtain $p_\Omega^\Phi(v) = p_\Omega(U^\top v)$. Thus, the computation of (4) reduces to (2), for which closed forms are often available. If $\Phi(v, p)$ is **concave** in p and $\mathcal{C}$ is a convex set, then we can solve (4) in polynomial time using an iterative algorithm, such as projected gradient ascent. If $\Phi(v, p)$ is nonconcave in p, then it is not possible to solve (4) in polynomial time in general, unless $\mathcal{C}$ is a discrete set of small cardinality. In general, we emphasize that $\Phi(v, p)$ can be nonconvex in v, as is typical with neural networks, since the maximization problem in (4) is with respect to $p \in \mathcal{C}$. However, in the sequel of this paper, certain properties we establish require $\Phi(v, p)$ to be convex in v.

We now give examples of regularized energy networks in increasing order of expressivity / complexity.

Generalized linear models. As a warm up, we consider the case of **bilinear** energy $\Phi(v, p)$. For instance, for probabilistic classification with k classes, where the goal is to predict $p \in \triangle^k$ from $x \in \mathbb{R}^d$, we can use $\Phi(v, p) = \langle v, p \rangle$ with $v = Wx + b$, where $W \in \mathbb{R}^{k \times d}$ and $b \in \mathbb{R}^k$. In this case, we therefore obtain $p_\Omega^\Phi(v) = p_\Omega(Wx + b)$, recovering generalized linear models [51, 48], the structured perceptron [27] and conditional random fields [43, 68].

Rectifier and maxout networks. We now consider the case of **convex-linear** energy $\Phi(v, p)$. A first example is a rectifier network [33] with one hidden layer. Indeed, $\Phi(v, p) = \langle \sigma(v), Up \rangle$ is convex in v if σ is an element-wise, convex and non-decreasing activation function, such as the relu or softplus, and if U and p are non-negative. A second example is a maxout network [34] (i.e., a max-affine function) or its smoothed counterpart, the log-sum-exp network [24]. Indeed, $\Phi(v, p) = \sigma(v) \cdot p$, where σ is the max or log-sum-exp operator, is convex in v, if the scalar p is non-negative. In general, it is always possible to construct a **convex-concave** energy from a jointly-convex function using the Legendre-Fenchel transform in the first argument.

Table 1: Examples of regularized energy networks. The v and p columns indicate the property of the energy function $\Phi(v, p)$ in these variables. The $L_\Omega^\Phi(v, y)$ column indicates the property of the loss in v. The linear-quadratic energy uses $v = (A, b)$ where A is negative semi-definite and $\Omega(p) = \frac{\gamma}{2}\|p\|_2^2$.

	$\Phi(v,p)$	v	p	$L_\Omega^\Phi(v,y)$	$p_\Omega^\Phi(v)$
GLM	$\langle v, p\rangle$	linear	linear	convex	$p_\Omega(v)$
Linear-quadratic	$\frac{1}{2}\langle p, Ap\rangle + \langle p, b\rangle$	linear	quadratic	convex	$(\gamma I - A)^{-1}b$
Rectifier network	$\langle \text{relu}(v), Up\rangle$	convex	linear	DC	$p_\Omega(U^\top \text{relu}(v))$
Maxout network	$p \cdot \max(v)$	convex	linear	DC	$p_\Omega(\max(v))$
LSE network	$p \cdot \text{LSE}^\gamma(v)$	convex	linear	DC	$p_\Omega(\text{LSE}^\gamma(v))$
ICNN	$-\text{ICNN}(v, p)$	nonconvex	concave	nonconvex	no closed form
Probabilistic	$\sum_{y\in\mathcal{Y}} p(y)E(v, y)$	nonconvex	linear	nonconvex	$\frac{\exp(E(v,\cdot))}{\sum_{y'\in\mathcal{Y}}\exp(E(v,y'))}$
Arbitrary	$\Phi(v, p)$	nonconvex	nonconcave	nonconvex	no closed form

Input-convex neural networks. As an example of **nonconvex-concave** energy, we consider $\Phi(v, p) = -\text{ICNN}(v, p)$, where ICNN is an input-convex neural network [4], i.e., it is convex in p but can be nonconvex in v. If $\Omega(p)$ and $\mathcal{C}$ are convex, then $\Phi(v, p)$ is concave in p and (4) can be solved in polynomial time using an iterative algorithm, such as projected gradient ascent.

Probabilistic energy networks. It is often desirable to define a conditional probability distribution

$$\mathbb{P}(Y = y|X = x) := \frac{\exp(E(v, y))}{\sum_{y'\in\mathcal{Y}} \exp(E(v, y'))}.$$

Such networks are typically trained using a cross-entropy loss (or equivalently, negative log-likelihood). Unfortunately, when $\mathcal{Y}$ is large or infinite, this loss and its gradients are intractable to compute, due to the normalization constant. Therefore, recent research has been devoted to developing approximate training schemes [66, 35, 50]. Although this is not the focus of this paper, we point out that probabilistic energy networks can also be seen as regularized energy networks in the space of probability distributions $\mathcal{C} = \triangle^{|\mathcal{Y}|}$, if we set $\Phi(v, p) = \sum_{y\in\mathcal{Y}} p(y)E(v, y)$ and $\Omega(p) = \sum_{y\in\mathcal{Y}} p(y) \log p(y)$. While the resulting optimization problem is intractable in general, it does suggest possible approximation schemes, such as the use of Frank-Wolfe methods [12, 42, 54].

Existing loss functions for energy networks. Given a pair (x, y) and output $v = g_\theta(x)$, how do we measure the discrepancy between $p_\Omega^\Phi(v)$ and y? One possibility [4, 13] is to use the composition of a differentiable loss $L\colon \mathcal{C} \times \mathcal{Y} \to \mathbb{R}_+$ with the argmax output, namely $(v, y) \mapsto L(p_\Omega^\Phi(v), y)$. However, computing the loss gradient w.r.t. v then requires to differentiate through $p_\Omega^\Phi(v)$, either through unrolling or implicit differentiation. This is particularly problematic when $\mathcal{C}$ is a complicated convex set, as differentiating through a projection can be challenging. In contrast, our proposed loss completely circumvents this need and enjoys easy-to-compute gradients. For unregularized energy networks, a naive idea, called the energy loss, is to use $(v, y) \mapsto -\Phi(v, y)$. However, this loss only works well if Φ is a similarity measure and works poorly in general [44]. A better choice is the generalized perceptron loss $(v, y) \mapsto \max_{p\in\mathcal{C}} \Phi(v, p) - \Phi(v, y)$ [44]. Our loss can be seen as a principled generalization of this loss to regularized energy networks, with theoretical guarantees.

4 Generalized conjugates

In order to devise generalized Fenchel-Young losses, we build upon a generalization due to Moreau [49, Chapter 14] of the convex conjugate Ω^*. Denoted Ω^Φ, it replaces the bilinear pairing in (1) with a more general coupling Φ [61, Chapter 11, Section L]. In this section, we state their definition, properties, closed-form expressions and connection to the C-transform in optimal transport.

Definition. Let $\Phi(v, p) \in \mathbb{R}$ be a coupling / energy function. The **Φ-convex conjugate** of $\Omega\colon \mathcal{C} \to \mathbb{R}$, also known as **Fenchel-Moreau conjugate**, is then defined by the value function

$$\Omega^\Phi(v) := \max_{p\in\mathcal{C}} \ \Phi(v, p) - \Omega(p). \tag{5}$$

The Φ-convex conjugate is an important tool in abstract convex analysis [65, 62]. Recently, it has been used to provide "Bellman-like" equations in stochastic dynamic programming [25] and to

provide tropical analogues of reproducing kernels [5]. We assume that the maximum is feasible for all $v \in \mathcal{V} = \mathbb{R}^d$, meaning that $\mathrm{dom}(\Omega^\Phi) = \mathcal{V}$. We emphasize again that, unlike with (1), v and p do not need to have compatible dimensions. We denote the argmax solution corresponding to (5) by

$$p_\Omega^\Phi(v) \coloneqq \operatorname*{argmax}_{p \in \mathcal{C}} \; \Phi(v,p) - \Omega(p). \tag{6}$$

If a function $F(v)$ can be written as $F(v) = \Omega^\Phi(v)$ for some Ω, it is called **Φ-convex** (in analogy, a function $f(v)$ is convex and closed if and only if it can be written as $f(v) = \Omega^*(v)$ for some Ω).

Properties. Φ-convex conjugates enjoy many useful properties, some of them are natural extensions of the usual convex conjugate properties. Proofs are provided in Appendix B.1.

Proposition 1 (Properties of Φ-convex conjugates)**.** *Let* $\Omega \colon \mathcal{C} \to \mathbb{R}$ *and* $\Phi \colon \mathcal{V} \times \mathcal{C} \to \mathbb{R}$.

1. **Generalized Fenchel-Young inequality:** for all $v \in \mathcal{V}$ and $p \in \mathcal{C}$,
$$\Omega^\Phi(v) + \Omega(p) - \Phi(v,p) \geq 0.$$
2. **Convexity:** If $\Phi(v,p)$ is convex in v, then $\Omega^\Phi(v)$ is convex (even if $\Omega(p)$ is nonconvex).
3. **Order reversing:** if $\Omega(p) \leq \Lambda(p)$ for all $p \in \mathcal{C}$, then $\Omega^\Phi(v) \geq \Lambda^\Phi(v)$ for all $v \in \mathcal{V}$.
4. **Continuity:** Ω^Φ shares the same continuity modulus as Φ.
5. **Gradient (envelope theorem):** Under assumptions (see paragraph below), we have $\nabla\Omega^\Phi(v) = \nabla_1 \Phi(v, p_\Omega^\Phi(v))$, where ∇_1 denotes the gradient in the first argument.
6. **Smoothness:** If $\mathcal{C}$ is a compact convex set, $\Phi(v,p)$ is β-smooth in (v,p), concave in p and $\Omega(p)$ is γ-strongly convex in p, then $\Omega^\Phi(v)$ is $(\beta + \beta^2/\gamma)$-smooth and $p_\Omega^\Phi(v)$ is β/γ-Lipschitz.

The condition on Φ and Ω in item 6 for Ω^Φ to be a smooth function (i.e., with Lipschitz-continuous gradients) is a novel result and will play a crucial role for establishing calibration guarantees in §6.

Assumptions for envelope theorems. The expression in item 5 allows to compute $\nabla\Omega^\Phi(v)$ without *argmax* differentiation. It is based on envelope theorems, which can be used for *max* differentiation. Indeed, we have $\Omega^\Phi(v) = \max_{p \in \mathcal{C}} F(v,p)$, where $F(v,p) \coloneqq \Phi(v,p) - \Omega(p)$. We assume that the maximum is unique and $\mathcal{C}$ is a compact. If $F(v,p)$ is convex in v, we apply Danskin's theorem [29] [15, Proposition B.25]. Without convexity assumption in v, if $F(v,p)$ is continuously differentiable in v for all $p \in \mathcal{C}$, $\nabla_1 F$ is continuous, we apply [61, Theorem 10.31]. When we do not compute the exact solution of (6), we only obtain an approximation of the gradient $\nabla\Omega^\Phi(v)$; see [1] for approximation guarantees. For other envelope theorem usecases in machine learning, see, e.g., [21].

Closed forms. While (5) and (6) may need to be solved numerically in general, they enjoy closed-form expressions in simple cases. Proofs are provided in Appendix B.2.

Proposition 2 (Closed-form expressions)**.** *Let* $\Omega \colon \mathcal{C} \to \mathbb{R}$ *and* $\Phi \colon \mathcal{V} \times \mathcal{C} \to \mathbb{R}$.

1. **Bilinear coupling:** If $\Phi(v,p) = \langle v, Up \rangle$, then $\Omega^\Phi(v) = \Omega^*(U^\top v)$ and $p_\Omega^\Phi(v) = p_\Omega(U^\top v)$.
2. **Linear-quadratic coupling:** If $\mathcal{C} = \mathbb{R}^k$, $\Omega(p) = \frac{\gamma}{2}\|p\|_2^2$ and $\Phi(v,p) = \frac{1}{2}\langle p, Ap \rangle + \langle p, b \rangle$, where $v = (A,b)$ and A is such that $(\gamma I - A)$ is positive definite, we obtain
$$\Omega^\Phi(v) = \frac{1}{2}\langle b, (\gamma I - A)^{-1} b \rangle \quad \text{and} \quad p_\Omega^\Phi(v) = (\gamma I - A)^{-1} b. \tag{7}$$
3. **Metric coupling:** If $\mathcal{V} = \mathcal{C}$, $\Phi = -C$ where $C(v,p)$ is a metric and Ω is M-Lipschitz with $M \leq 1$, then $\Omega^\Phi = -\Omega$ and $p_\Omega^\Phi(v) = v$.

Relation with the C-transform. Given a cost function C, we may define a $\min$ counterpart of (5),

$$\Lambda_C(v) \coloneqq \min_{p \in \mathcal{C}} \; C(v,p) - \Lambda(p). \tag{8}$$

In the optimal transport literature, this is known as the C-transform of Λ [64, 59]. When C is bilinear, this recovers the notion of concave conjugate [20]. When $C(v,p) = c(v-p)$ for some c, this recovers the infimal convolution [20]. It is easy to check that $\Omega = -\Lambda \Leftrightarrow \Omega^\Phi = -\Lambda_C$ with $C = -\Phi$. Thus, Ω^Φ and Λ_C are the natural extensions of convex and concave conjugates, respectively. We opt for the former in this paper to closely mirror the usual convex conjugates and Fenchel-Young losses.

5 Generalized Fenchel-Young losses

Definition. The generalized Fenchel-Young inequality in Proposition 1 leads us to propose the generalized Fenchel-Young loss, the natural extension of (3) to Φ-convex conjugates:

$$L_\Omega^\Phi(v, y) \coloneqq \Omega^\Phi(v) + \Omega(y) - \Phi(v, y). \tag{9}$$

We also have the relationships $L_\Omega^\Phi(v, y) = F(v, y) - F(v, p_\Omega^\Phi(v))$, where $F(v, p) = \Omega(p) - \Phi(v, p)$, and $p_\Omega^\Phi(v) = \operatorname{argmin}_{p \in \mathcal{C}} \; L_\Omega^\Phi(v, p)$. We now give an intuitive geometric interpretation of (9).

Geometric interpretation. For the sake of illustration, let us set $\Omega(p) = \frac{\gamma}{2}p^2$ and $\Phi(v, p) = \frac{1}{2}ap^2 + bp$. From (1), the line $p \mapsto \langle u, p \rangle - \Omega^*(u)$, where $\Omega^*(u) = \frac{1}{2\gamma}u^2$, is the tightest linear lower bound on $\Omega(p)$, here depicted with $u = 0.6$. It is the tangent of $\Omega(p)$ at $p = p_\Omega(u)$. Similarly, from (5), $p \mapsto \Phi(v, p) - \Omega^\Phi(v)$ is the tightest lower-bound of this form given v. It touches $\Omega(p)$ at $p = p_\Omega^\Phi(v)$, here depicted with $v = (a, b)$, $a = -1$ and $b = 1$, for which we have the closed form $\Omega^\Phi(v) = \frac{1}{2}(\gamma - a)^{-1}b^2$ (Proposition 2). The generalized Fenchel-Young loss (9) is then the **gap** between $\Omega(p)$ and $\Phi(v, p) - \Omega^\Phi(v)$, evaluated at the ground-truth $p = y$. The goal of training is to adjust the parameters θ of a network $v = g_\theta(x)$ so as to minimize this gap averaged over all (x, y) training pairs.

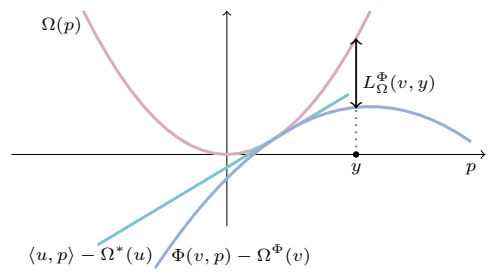

Figure 1: Geometric interpretation with $\Omega(p) = \frac{\gamma}{2}p^2$ and $\Phi(v, p) = \frac{1}{2}ap^2 + bp$.

Properties. Generalized Fenchel-Young losses enjoy many desirable properties, as we now show.

Proposition 3 (Properties of generalized F-Y losses)**.** *Let* $\Omega \colon \mathcal{C} \to \mathbb{R}$ *and* $\Phi \colon \mathcal{V} \times \mathcal{C} \to \mathbb{R}$.

1. **Non-negativity:** $L_\Omega^\Phi(v, p) \geq 0$ for all $v \in \mathcal{V}$ and $p \in \mathcal{C}$.
2. **Zero loss:** if the maximum in (5) exists and is unique, $L_\Omega^\Phi(v, y) = 0 \Leftrightarrow y = p_\Omega^\Phi(v)$.
3. **Gradient:** $\nabla_1 L_\Omega^\Phi(v, p) = \nabla \Omega^\Phi(v) - \nabla_1 \Phi(v, p)$.
4. **Difference of convex (DC):** if $\Phi(v, p)$ is convex in v, then $L_\Omega^\Phi(v, p)$ is a difference of convex functions in v: $\Omega^\Phi(v)$ and $\Phi(v, p)$. If $\Phi(v, p)$ is linear in v, then $L_\Omega^\Phi(v, p)$ is convex in v.
5. **Smaller output set, smaller loss.** If $\mathcal{C}' \subseteq \mathcal{C}$ and Ω' is the restriction of Ω to $\mathcal{C}'$, then $L_{\Omega'}^\Phi(v, p) \leq L_\Omega^\Phi(v, p)$ for all $v \in \mathcal{V}$ and $p \in \mathcal{C}'$.
6. **Quadratic lower-bound.** If $\Phi(v, p) - \Omega(p)$ is γ-strongly concave in p w.r.t. $\|\cdot\|$ over $\mathcal{C}$ and $\mathcal{C}$ is a closed convex set, then for all $v \in \mathcal{V}$ and $p \in \mathcal{C}$
$$\frac{\gamma}{2}\|p - p_\Omega^\Phi(v)\|^2 \leq L_\Omega^\Phi(v, p). \tag{10}$$
7. **Upper-bounds.** If $\Phi(v, p) - \Omega(p)$ is α-Lipschitz in p w.r.t. $\|\cdot\|$ over $\mathcal{C}$, then $L_\Omega^\Phi(v, p) \leq \alpha\|p - p_\Omega^\Phi(v)\|$. If $\Phi(v, p)$ is concave in p, then $L_\Omega^\Phi(v, p) \leq L_\Omega(\nabla_2 \Phi(v, p), p)$.

Proofs are given in Appendix B.3. The **non-negativity** is a good property for a loss function. The **zero-loss** property $L_\Omega^\Phi(v, y) = 0 \Leftrightarrow y = p_\Omega^\Phi(v)$, which is true for instance if $p \mapsto \Phi(v, p) - \Omega(p)$ is strictly concave, is key as it allows to use $p_\Omega^\Phi(v)$ defined in (6) as the (implicit) output layer associated with an energy network. The **gradient** $\nabla_1 L_\Omega^\Phi(v, y)$ does not require to differentiate through the argmax problem in (6) needed for computing $p_\Omega^\Phi(v)$. Typically, differentiating through an argmax or argmin, as is done in input-convex neural networks [4], requires either unrolling or implicit differentiation [36, 14, 41, 19, 17] and is therefore more costly.

These two paragraphs are now better connected to Proposition 3.

The fact that $L_\Omega^\Phi(v, p)$ is a **difference of convex (DC)** functions in v when $\Phi(v, p)$ is convex in v suggests that we can use DC programming techniques, such as the convex-concave procedure [71], for training such energy networks. We leave the investigation of this observation to future work. The **"smaller output set, smaller loss"** property means that we can achieve the smallest loss by choosing the smallest set $\mathcal{C}$ in (6) such that $\mathcal{Y} \subseteq \mathcal{C}$. The smallest such convex set is the convex hull of $\mathcal{Y}$, also

known as marginal polytope [70] when $\mathcal{Y} \subseteq \{0,1\}^k$. The **quadratic lower-bound** relates our loss to using $p_\Omega^\Phi(v)$ within a squared norm loss. The **upper-bounds** relate our loss to using $p_\Omega^\Phi(v)$ within a norm loss and to using a regular Fenchel-Young loss with a linearized energy.

If Ω is the indicator function of $\mathcal{C}$, i.e., $\Omega(p) = 0$ if $p \in \mathcal{C}$, ∞ otherwise, then it can be checked that we recover the "generalized perceptron" loss [45, 44] as a special case of (9). Proposition 3 therefore provides new properties to understand and analyze this loss.

Regular Fenchel-Young losses are closely related to Bregman divergences [18]. In Appendix A, we build a generalized notion of Bregman divergence using generalized conjugates.

Training. To train regularized energy networks with our framework, the user should choose an energy $\Phi(v,p)$, a regularization $\Omega(p)$, an output set $\mathcal{C}$ (from Proposition 3 item 5, the smaller this set the better) and the model $v = g_\theta(x)$ with input x and parameters θ. Given a set of input-output pairs $(x_1, y_1), \ldots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$, where $\mathcal{Y} \subseteq \mathcal{C}$, we can find the parameters θ by minimizing the empirical risk objective regularized by $R\colon \Theta \to \mathbb{R}$,

$$\widehat{\theta} = \operatorname*{argmin}_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} L_\Omega^\Phi(g_\theta(x_i), y_i) + R(\theta). \tag{11}$$

Thanks to the easy-to-compute gradients of L_Ω^Φ, we can easily solve (11) using any (stochastic) solver.

6 Calibration guarantees

Many times, notably for differentiability reasons, the loss used at training time, here our generalized Fenchel-Young loss $L_\Omega^\Phi(v,y)$, is used as a surrogate / proxy for a different (potentially discrete loss) $L\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, used at test time. Calibration guarantees [72, 73, 9, 67] ensure that minimizing the excess of risk of the train loss will also minimize that of the test loss (a.k.a. target loss). We study in this section such guarantees, assuming that L satisfies an affine decomposition property [26, 16]:

$$L(\widehat{y}, y) = \langle \varphi(\widehat{y}), V\varphi(y) + b \rangle + c(y), \tag{12}$$

where $\varphi(y) \mapsto V\varphi(y) + b$ is an affine map, $\varphi(y)$ is a label embedding and $c(y)$ is any function that depends only on y. Numerous losses can be written in this form. Examples include the zero-one, Hamming, NDCG and precision at k losses [55, 16]. Inference in this setting works in two steps. First, we compute a "soft" (continuous) prediction $p = p_\Omega^\Phi(v) \in \mathcal{C}$. Second, we compute a "hard" (discrete) prediction by a decoding / rounding from $\mathcal{C}$ to $\mathcal{Y}$, calibrated for the loss L:

$$y_L(p) \coloneqq \operatorname*{argmin}_{\widehat{y} \in \mathcal{Y}} L(\widehat{y}, p) = \operatorname*{argmin}_{\widehat{y} \in \mathcal{Y}} \langle \varphi(\widehat{y}), V\varphi(p) + b \rangle. \tag{13}$$

The target risk of $f\colon \mathcal{X} \to \mathcal{Y}$ and the surrogate risk of $g\colon \mathcal{X} \to \mathcal{V}$ are defined by

$$\mathcal{L}(f) \coloneqq \mathbb{E}_{(X,Y)\sim\rho}\, L(f(X), Y) \quad \text{and} \quad \mathcal{L}_\Omega^\Phi(g) \coloneqq \mathbb{E}_{(X,Y)\sim\rho}\, L_\Omega^\Phi(g(X), Y),$$

where ρ is a typically unknown distribution over $\mathcal{X} \times \mathcal{Y}$. The Bayes predictors are defined by $f^\star \coloneqq \operatorname{argmin}_{f\colon \mathcal{X} \to \mathcal{Y}} \mathcal{L}(f)$ and $g^\star \coloneqq \operatorname{argmin}_{g\colon \mathcal{X} \to \mathcal{V}} \mathcal{L}_\Omega^\Phi(g)$. We now establish calibration of the surrogate excess risk, under the assumption that the energy $\Phi(v,p)$ is **linear-concave**, i.e., it can be written as $\Phi(v,p) = \langle v, \varphi(p) \rangle$, for some function φ.

Proposition 4. *Calibration of target and surrogate excess risks*

Assume $L_\Omega^\Phi(v,y)$ is M-smooth in v w.r.t. the dual norm $\|\cdot\|_*$, $\mathcal{C}$ is a compact convex set such that $\mathcal{Y} \subseteq \mathcal{C}$ and $\Phi(v,p) = \langle v, \varphi(p) \rangle$. Let $\sigma \coloneqq \sup_{y \in \mathcal{Y}} \|V^\top \varphi(y)\|_*$. Then, the generalized Fenchel-Young loss (9) is calibrated with the target loss (12) with decoder $d = y_L \circ p_\Omega^\Phi$:

$$\forall g\colon \mathcal{X} \to \mathcal{V} \quad \frac{(\mathcal{L}(d \circ g) - \mathcal{L}(f^\star))^2}{8\sigma^2 M} \leq \mathcal{L}_\Omega^\Phi(g) - \mathcal{L}_\Omega^\Phi(g^\star),$$

where $p_\Omega^\Phi\colon \mathcal{V} \to \mathcal{C}$ is defined in (6) and $y_L\colon \mathcal{C} \to \mathcal{Y}$ is defined in (13).

The proof is in Appendix B.4. Proposition 1 item 6 shows that the smoothness of Φ and strong convexity of Ω ensure the smoothness of L_Ω^Φ. Calibration also implies Fisher consistency, namely

Table 2: Multilabel classification results using various energies (test accuracy in %).

Energy	yeast	scene	mediamill	birds	emotions	cal500
Unary (linear)	79.76	89.14	96.84	86.47	78.22	85.67
Unary (rectifier network)	80.03	91.35	96.91	91.74	79.79	86.25
Pairwise	**80.19**	**91.58**	**96.95**	91.55	**80.56**	85.73
SPEN	79.99	91.24	96.68	91.41	79.35	86.25
Input-concave SPEN	80.00	90.64	**96.95**	**91.77**	79.73	**86.35**

$\mathcal{L}_\Omega^\Phi(g) = \mathcal{L}_\Omega^\Phi(g^\star) \Rightarrow \mathcal{L}(d \circ g) = \mathcal{L}(f^\star)$, when using the decoder $d = y_L \circ p_\Omega^\Phi$. The existing proof technique for the calibration of regular Fenchel-Young losses [55, 16] assumes a bilinear pairing and a loss of the form $L_{\Omega_\varphi}(u, \varphi(y)) = \Omega_\varphi^*(u) + \Omega_\varphi(\varphi(y)) - \langle u, \varphi(y) \rangle$, where Ω_φ is a strongly-convex regularizer w.r.t. $\varphi(y)$. Our novel proof technique is more general, as it works with any linear-concave energy and the regularizer Ω is w.r.t. y, not $\varphi(y)$. Unlike the existing proof, our proof is valid for the pairwise model we present in the next section.

7 Experiments

7.1 Multilabel classification

We study in this section the application of generalized Fenchel-Young losses to multi-label classification, setting $\mathcal{Y} = \{0, 1\}^k$ and $\mathcal{C} = [0, 1]^k$, where k is the number of labels. When the loss L in (12) is the Hamming loss (1 - accuracy), our loss is calibrated for L and the decoding (13) is just $\widehat{y}_j = 1$ if $p_j > 0.5$ else 0. We therefore report our empirical results using the accuracy metric.

Unary model. We consider a neural network $u = g_\theta(x) \in \mathbb{R}^k$, assigning a score u_j to each label $j \in [k]$. With the bilinear pairing $\Phi(u, p) = \langle u, p \rangle$, we get

$$p_\Omega^\Phi(u) = \operatorname*{argmax}_{p \in [0,1]^k} \langle u, p \rangle - \Omega(p) = p_\Omega(u). \tag{14}$$

That is, (14) is just a normal neural network with p_Ω as output layer. When $\Omega(p) = \Omega_1(p) + \Omega_1(1-p)$, where $\Omega_1(p) := \langle p, \log p \rangle$ is Shannon's negentropy, we get $p_\Omega(v) = \text{sigmoid}(v) := 1/(1+\exp(-v))$ and (9) is just the usual binary logistic / cross-entropy loss. When $\Omega(p) = \Omega_2(p) + \Omega_2(1-p)$, where $\Omega_2(p) := \frac{1}{2}\langle p, 1-p \rangle$ is Gini's negentropy, we get a sparse sigmoid and the binary sparsemax loss [18, §6.2]. Because $\mathcal{C} = [0, 1]^k$ is the convex hull of $\mathcal{Y} = \{0, 1\}^k$, (14) can be intepreted as a **marginal probability** [70]. Indeed, there exists a probability distribution $\mathbb{P}(Y|X)$ over $Y \in \mathcal{Y}$ such that

$$[p_\Omega^\Phi(u)]_j = \mathbb{P}(Y_j = 1 | X = x).$$

Pairwise model. We now additionally use a network $U = h_\theta(x) \in \mathbb{R}^{k \times k}$, assigning a score $U_{i,j}$ to the pairwise interaction between labels i and j. With $v = (u, U)$ and the **linear-quadratic** coupling $\Phi(v, p) = \langle u, p \rangle + \frac{1}{2}\langle p, Up \rangle$, we get

$$p_\Omega^\Phi(v) = \operatorname*{argmax}_{p \in [0,1]^k} \ \langle u, p \rangle + \frac{1}{2}\langle p, Up \rangle - \Omega(p). \tag{15}$$

If U is negative semi-definite, i.e., $U = -AA^\top$ for some matrix $A \in \mathbb{R}^{k \times m}$, then the problem is concave in p and can be solved optimally. Moreover, since $\Phi(v, p)$ is linear-concave, the calibration guarantees in §6 hold and $L_\Omega^\Phi(v, y)$ is convex in v. In our experiments, we use a matrix A of rank 1 (cf. Appendix C). Unlike (7), (15) does not enjoy a closed form due to the constraints. We solve it by coordinate ascent: if Ω is quadratic, as is the case with Gini's negentropy, the coordinate-wise updates can be computed in closed-form. Again, (15) can be interpreted as a marginal probability. In constrast to (15), marginal inference in the closely related **Ising model** is known to be #P-hard [32].

SPEN model. Following SPENs [11, Eq. 4 and 5], we also tried the energy $\Phi(v, p) = \langle u, p \rangle - \Psi(w, p)$, where $v = (u, w)$, $u = g_\theta(x)$ and w are the weights of the "prior network" Ψ (independent of x). We also tried a variant where Ψ is made convex in p, making $\Phi(v, p)$ concave in p. In both cases, we compute $p_\Omega^\Phi(v)$ by solving (6) using projected gradient ascent with backtracking linesearch.

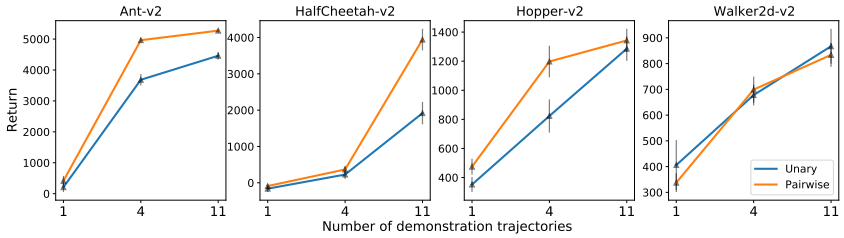

Figure 2: Average performance (higher is better) and standard deviation over 10 seeds.

Experimental setup. We perform experiments on 6 publicly-available datasets, see Appendix C. We use the train-test split from the dataset when provided. When not, we use 80% for training data and 20% for test data. For all models, we solve the outer problem (11) using ADAM. We set $\Omega(p)$ to the Gini negentropy. We hold out 25% of the training data for hyperparameter validation purposes. We set $R(\theta)$ in (11) to $\frac{\lambda}{2}\|\theta\|_2^2$. For the regularization hyper-parameter λ, we search 5 log-spaced values between 10^{-4} and 10^{1}. For the learning rate parameter of ADAM, we search 10 log-spaced valued between 10^{-5} and 10^{-1}. Once we selected the best hyperparameters, we refit the model on the entire training set. We average results over 3 runs with different seeds.

Results. Table 2 shows a model comparison. We observe improvements with the pairwise model on 4 out of 6 datasets, confirming that our losses are able to learn useful models. Input-concavity helps improve SPENs in 4 out of 6 datasets. Table 4 confirms that using the envelope theorem for computing gradients works comparably to (if not better than) the implicit function theorem. Table 5 shows that our losses outperform the energy, the cross-entropy and the generalized perceptron losses.

7.2 Imitation learning

In this section, we study the application of generalized Fenchel-Young losses to imitation learning. This setting consists in learning a policy $\pi : \mathcal{X} \mapsto \mathcal{Y}$, a mapping from states $\mathcal{X}$ to actions $\mathcal{Y}$, from a fixed dataset of expert demonstrations $(x_1, y_1), \ldots, (x_n, y_n) \in \mathcal{X} \times \mathcal{Y}$. In particular, we only consider a Behavior Cloning approach [60], which essentially reduces imitation learning to supervised learning (as opposed to inverse RL methods [63, 53]). The learned policy π is evaluated on its performance, which is the expected sum of rewards of the environment [69].

Experimental setup. We consider four MuJoCo Gym locomotion environments [23] together with the demonstrations provided by Orsini et al. [57]; see Appendix C.2 for details. We evaluate the learned policy π for different number of demonstration trajectories: 1, 4 and 11, consistently with [38, 40, 28]. The action space in the demonstrations is included in $[-1, 1]^k$, where the dimensionality of the action space k corresponds to the torques of the actuators. We scale the action space to the hypercube $\mathcal{Y} = \mathcal{C} = [0, 1]^k$ at learning time and scale it back to the original action space at inference time. Similarly to the multilabel classification setup, we evaluate the unary and pairwise models, with the only difference that we use two hidden layers instead of one, since it leads to significantly better performance. We specify the hyperparameter selection procedure in Appendix C.2.

Results. We run the best hyperparameters over 10 seeds and report final performance over 100 evaluation episodes. Figure 2 shows a clear improvement of the pairwise model over the unary model for 3 out of 4 tasks. Contrary to the unary model, the pairwise model enables to capture the interdependence between the different torques of the action space, translating into better performance.

8 Conclusion

Building upon generalized conjugate functions, we proposed generalized Fenchel-Young losses, a natural loss construction for learning energy networks and studied its properties. Thanks to conditions on the energy Φ and the regularizer Ω ensuring the loss smoothness, we established calibration guarantees for the case of linear-concave energies, a more general result than the existing analysis, restricted to bilinear energies. We demonstrated the effectiveness of our losses on multilabel classification and imitation learning tasks. We hope that this paper will help popularize generalized conjugates as a powerful tool for machine learning and optimization.

Acknowledgments

References

[1] P. Ablin, G. Peyré, and T. Moreau. Super-efficiency of automatic differentiation for functions defined as a minimum. In *Proc. of ICML*, pages 32–41. PMLR, 2020.

[2] S. Acharyya. Learning to rank in supervised and unsupervised settings using convexity and monotonicity. 2013.

[3] S. Amari. *Information Geometry and Its Applications*. Springer, 2016.

[4] B. Amos, L. Xu, and J. Z. Kolter. Input convex neural networks. In *ICML*, pages 146–155. PMLR, 2017.

[5] P.-C. Aubin-Frankowski and S. Gaubert. The tropical analogues of reproducing kernels. *arXiv preprint arXiv:2202.11410*, 2022.

[6] G. BakIr, T. Hofmann, B. Schölkopf, A. J. Smola, and B. Taskar. *Predicting structured data*. MIT press, 2007.

[7] A. Banerjee, S. Merugu, I. S. Dhillon, J. Ghosh, and J. Lafferty. Clustering with bregman divergences. *Journal of machine learning research*, 6(10), 2005.

[8] O. Barndorff-Nielsen. *Information and Exponential Families: In Statistical Theory*. John Wiley & Sons, 1978.

[9] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

[10] A. Beck. *First-order methods in optimization*. SIAM, 2017.

[11] D. Belanger and A. McCallum. Structured prediction energy networks. In *International Conference on Machine Learning*, pages 983–992. PMLR, 2016.

[12] D. Belanger, D. Sheldon, and A. McCallum. Marginal inference in MRFs using Frank-Wolfe. In *NeurIPS Workshop on Greedy Opt., FW and Friends*, 2013.

[13] D. Belanger, B. Yang, and A. McCallum. End-to-end learning for structured prediction energy networks. In *International Conference on Machine Learning*, pages 429–439. PMLR, 2017.

[14] B. M. Bell and J. V. Burke. Algorithmic differentiation of implicit functions and optimal values. In *Advances in Automatic Differentiation*, pages 67–77. Springer, 2008.

[15] D. P. Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

[16] M. Blondel. Structured prediction with projection oracles. *Advances in neural information processing systems*, 32, 2019.

[17] M. Blondel, Q. Berthet, M. Cuturi, R. Frostig, S. Hoyer, F. Llinares-López, F. Pedregosa, and J.-P. Vert. Efficient and modular implicit differentiation. *arXiv preprint arXiv:2105.15183*, 2021.

[18] M. Blondel, A. F. Martins, and V. Niculae. Learning with fenchel-young losses. *JMLR*, 21(35):1–69, 2020.

[19] J. F. Bonnans and A. Shapiro. *Perturbation analysis of optimization problems*. Springer Science & Business Media, 2013.

[20] J. Borwein and A. Lewis. *Convex Analysis*. 2006.

[21] L. Bottou, M. Arjovsky, D. Lopez-Paz, and M. Oquab. Geometrical insights for implicit generative modeling. pages 229–268. 2018.

[22] S. Boyd, S. P. Boyd, and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[23] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym, 2016.

[24] G. C. Calafiore, S. Gaubert, and C. Possieri. Log-sum-exp neural networks and posynomial models for convex and log-log-convex data. *IEEE transactions on neural networks and learning systems*, 31(3):827–838, 2019.

[25] J.-P. Chancelier and M. De Lara. Fenchel-moreau conjugation inequalities with three couplings and application to stochastic bellman equation. *arXiv preprint arXiv:1804.03034*, 2018.

[26] C. Ciliberto, L. Rosasco, and A. Rudi. A consistent regularization approach for structured prediction. In *Proc. of NeurIPS*, pages 4412–4420, 2016.

[27] M. Collins. Discriminative training methods for Hidden Markov Models: Theory and experiments with perceptron algorithms. In *Proc. of EMNLP*, 2002.

[28] R. Dadashi, L. Hussenot, M. Geist, and O. Pietquin. Primal wasserstein imitation learning. *International Conference on Learning Representations*, 2021.

[29] J. M. Danskin. *The theory of max-min and its application to weapons allocation problems*. Springer Science & Business Media, 1967.

[30] J. C. Duchi, K. Khosravi, and F. Ruan. Multiclass classification, information, divergence, and surrogate risk. *The Annals of Statistics*, 46(6B):3246–3275, 2018.

[31] S. Ghadimi and M. Wang. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*, 2018.

[32] A. Globerson, T. Roughgarden, D. Sontag, and C. Yildirim. How hard is inference for structured prediction? In *International Conference on Machine Learning*, pages 2181–2190. PMLR, 2015.

[33] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *AISTATS*, pages 315–323. JMLR Workshop and Conference Proceedings, 2011.

[34] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio. Maxout networks. In *ICML*, pages 1319–1327. PMLR, 2013.

[35] W. Grathwohl, J. Kelly, M. Hashemi, M. Norouzi, K. Swersky, and D. Duvenaud. No mcmc for me: Amortized samplers for fast and stable training of energy-based models. 2021.

[36] A. Griewank and A. Walther. *Evaluating derivatives: principles and techniques of algorithmic differentiation*. SIAM, 2008.

[37] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex analysis and minimization algorithms II*, volume 305. Springer science & business media, 1993.

[38] J. Ho and S. Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, 2016.

[39] S. Kakade, S. Shalev-Shwartz, A. Tewari, et al. On the duality of strong convexity and strong smoothness: Learning applications and matrix regularization. *Tech report*, 2(1):35, 2009.

[40] I. Kostrikov, K. K. Agrawal, D. Dwibedi, S. Levine, and J. Tompson. Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning. *International Conference on Learning Representations*, 2019.

[41] S. G. Krantz and H. R. Parks. *The implicit function theorem: history, theory, and applications*. Springer Science & Business Media, 2012.

[42] R. G. Krishnan, S. Lacoste-Julien, and D. Sontag. Barrier Frank-Wolfe for marginal inference. In *Proc. of NeurIPS*, 2015.

[43] J. D. Lafferty, A. McCallum, and F. C. Pereira. Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. of ICML*, 2001.

[44] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

[45] Y. LeCun and F. J. Huang. Loss functions for discriminative training of energy-based models. In *International Workshop on Artificial Intelligence and Statistics*, pages 206–213. PMLR, 2005.

[46] T. Lin, C. Jin, and M. Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *International Conference on Machine Learning*, pages 6083–6093. PMLR, 2020.

[47] A. F. Martins and R. F. Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proc. of ICML*, 2016.

[48] P. McCullagh and J. A. Nelder. *Generalized Linear Models*, volume 37. CRC press, 1989.

[49] J.-J. Moreau. Fonctionnelles convexes. *Séminaire Jean Leray*, (2):1–108, 1966.

[50] C. Nash and C. Durkan. Autoregressive energy machines. In *International Conference on Machine Learning*, pages 1735–1744. PMLR, 2019.

[51] J. A. Nelder and R. J. Baker. *Generalized Linear Models*. Wiley Online Library, 1972.

[52] Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2003.

[53] A. Y. Ng, S. J. Russell, et al. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning*, 2000.

[54] V. Niculae, A. F. Martins, M. Blondel, and C. Cardie. SparseMAP: Differentiable sparse structured inference. In *Proc. of ICML*, 2018.

[55] A. Nowak-Vila, F. Bach, and A. Rudi. A general theory for structured prediction with smooth convex surrogates. *arXiv preprint arXiv:1902.01958*, 2019.

[56] F. Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

[57] M. Orsini, A. Raichuk, L. Hussenot, D. Vincent, R. Dadashi, S. Girgin, M. Geist, O. Bachem, O. Pietquin, and M. Andrychowicz. What matters for adversarial imitation learning? *Advances in Neural Information Processing Systems*, 2021.

[58] A. Osokin, F. Bach, and S. Lacoste-Julien. On structured prediction theory with calibrated convex surrogate losses. In *Proc. of NIPS*, pages 302–313, 2017.

[59] G. Peyré, M. Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[60] D. A. Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural computation*, 1991.

[61] R. T. Rockafellar and R. J.-B. Wets. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

[62] A. M. Rubinov. *Abstract convexity and global optimization*. Springer Science & Business Media, 2000.

[63] S. Russell. Learning agents for uncertain environments. In *Conference on Computational learning theory*, 1998.

[64] F. Santambrogio. Optimal transport for applied mathematicians. *Birkäuser, NY*, 55(58-63):94, 2015.

[65] I. Singer. *Abstract convex analysis*, volume 25. John Wiley & Sons, 1997.

[66] Y. Song and D. P. Kingma. How to train your energy-based models. *arXiv preprint arXiv:2101.03288*, 2021.

[67] I. Steinwart. How to compare different loss functions and their risks. *Constructive Approximation*, 26(2):225–287, 2007.

[68] C. Sutton, A. McCallum, et al. An introduction to conditional random fields. *Foundations and Trends in Machine Learning*, 4(4):267–373, 2012.

[69] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. 2018.

[70] M. J. Wainwright, M. I. Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

[71] A. L. Yuille and A. Rangarajan. The concave-convex procedure. *Neural computation*, 15(4):915–936, 2003.

[72] T. Zhang. Statistical analysis of some multi-category large margin classification methods. *Journal of Machine Learning Research*, 5(Oct):1225–1251, 2004.

[73] T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statistics*, 32(1):56–85, 2004.

[74] X. Zhou. On the fenchel duality between strong convexity and lipschitz continuous gradient. *arXiv preprint arXiv:1803.06573*, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] for data and instructions, code will be released upon publication
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Generalized Bregman divergences

The expression $\Omega^*(v) + \Omega(p) - \langle v, p\rangle$ at the heart of Fenchel-Young losses is closely related to Bregman divergences; see [3, Theorem 1.1] and [18]. In this section, we develop a similar relationship between generalized Fenchel-Young losses and a new generalized notion of Bregman divergence.

Biconjugates. We begin by recalling well-known results on biconjugates [22]. Applying the conjugate (1) twice, we obtain the biconjugate $\Omega^{**}(p)$. It is well-known that a function Ω is convex *and* closed (i.e., lower-semicontinuous) if and only if $\Omega = \Omega^{**}$. This therefore provides a variational characterization of lower-semicontinuous convex functions. This characterization naturally motivates the class of Φ-convex functions (§4). If Ω is nonconvex, Ω^{**} is Ω's tightest convex lower bound.

Generalized biconjugates. The generalized conjugate in (5) uses maximization w.r.t. the second argument $p \in \mathcal{C}$. To obtain a generalized conjugate whose maximization is w.r.t. the first argument $v \in \mathcal{V}$ instead, we define $\Psi(p, v) \coloneqq \Phi(v, p)$. Note that if Φ is symmetric, the distinction between Φ and Ψ is not necessary. Similarly to (5), we can then define the Ψ-conjugate of $\Lambda\colon \mathcal{V} \to \mathbb{R}$ as

$$\Lambda^{\Psi}(p) = \max_{v\in\mathcal{V}} \ \Psi(p, v) - \Lambda(v) = \max_{v\in\mathcal{V}} \ \Phi(v, p) - \Lambda(v), \tag{16}$$

i.e., the maximization is over the left argument of Φ. We define the corresponding argmax as

$$v^{\Psi}_{\Lambda}(p) \coloneqq \operatorname*{argmax}_{v\in\mathcal{V}} \ \Psi(p, v) - \Lambda(v) = \operatorname*{argmax}_{v\in\mathcal{V}} \ \Phi(v, p) - \Lambda(v). \tag{17}$$

In particular, with $\Lambda \coloneqq \Omega^{\Phi}$, we can define the generalized biconjugate $\Lambda^{\Psi} = \Omega^{\Phi\Psi}\colon \mathcal{C} \to \mathbb{R}$. Generalized biconjugates enjoy similar properties as regular biconjugates, as we now show.

Proposition 5 (Properties of generalized biconjugates)**.** *Let* $\Omega\colon \mathcal{C} \to \mathbb{R}$, $\Phi\colon \mathcal{V} \times \mathcal{C} \to \mathbb{R}$ *and* $\Psi(p, v) \coloneqq \Phi(v, p)$.

1. **Lower-bound:** $\Omega^{\Phi\Psi}(p) \leq \Omega(p)$ for all $p \in \mathcal{C}$.
2. **Equality:** $\Omega^{\Phi\Psi}(p) = \Omega(p)$ if and only if Ω is Ψ-convex.
3. **Tightest lower-bound:** $\Omega^{\Phi\Psi}(p)$ is the tightest Ψ-convex lower-bound of $\Omega(p)$.

Proofs are given in Appendix B.5. Similar results hold for $\Lambda^{\Psi\Phi}$, where $\Lambda\colon \mathcal{V} \to \mathbb{R}$.

Definition. We can now define the **generalized Bregman divergence** $D^{\Phi}_{\Omega}\colon \mathcal{C} \times \mathcal{C} \to \mathbb{R}_+$ as

$$D^{\Phi}_{\Omega}(p, p') \coloneqq \Omega(p) - \Phi(v^{\Psi}_{\Omega^{\Phi}}(p'), p) - \Omega(p') + \Phi(v^{\Psi}_{\Omega^{\Phi}}(p'), p'), \tag{18}$$

where we recall that $\Psi(p, v) \coloneqq \Phi(v, p)$ and using (17) we have

$$v^{\Psi}_{\Omega^{\Phi}}(p) = \operatorname*{argmax}_{v\in\mathcal{V}} \ \Psi(p, v) - \Omega^{\Phi}(v) = \operatorname*{argmax}_{v\in\mathcal{V}} \ \Phi(v, p) - \Omega^{\Phi}(v). \tag{19}$$

We have therefore obtained a notion of Bregman divergence parametrized by a coupling $\Phi(v, p)$, such as a neural network.

As shown in the proposition below, generalized Bregman divergences enjoy similar properties as regular Bregman divergences. Proofs are given in Appendix B.6.

Proposition 6 (Properties of generalized Bregman divergences)**.** *Let Ω be a Ψ-convex function, where $\Psi(p, v) := \Phi(v, p)$.*

1. **Link with generalized FY loss:** Denoting $v := v^{\Psi}_{\Omega^{\Phi}}(p')$, we have $D^{\Phi}_{\Omega}(p, p') = L^{\Phi}_{\Omega}(v, p)$.
2. **Non-negativity:** $D^{\Phi}_{\Omega}(p, p') \geq 0$, for all $p, p' \in \mathcal{C}$.
3. **Identity of indiscernibles:** $D^{\Phi}_{\Omega}(p, p') = 0 \Leftrightarrow p = p'$ if $p \mapsto \Phi(v, p) - \Omega(p)$ is strictly concave for all $v \in \mathcal{V}$.
4. **Convexity:** if $p \mapsto \Phi(v, p) - \Omega(p)$ is concave for all $v \in \mathcal{V}$, then $D^{\Phi}_{\Omega}(p, p')$ is convex in p.
5. **Recovering Bregman divergences:** If $\Phi(v, p)$ is the bilinear coupling $\langle v, p\rangle$, then we recover the usual Bregman divergence $D_{\Omega} \colon \mathcal{C} \times \mathcal{C} \to \mathbb{R}_+$

$$D^{\Phi}_{\Omega}(p, p') = D_{\Omega}(p, p') := \Omega(p) - \Omega(p') - \langle \nabla\Omega(p'), p - p'\rangle. \tag{20}$$

Some remarks:

- The generalized Bregman divergence is between objects p and p' of the same space $\mathcal{C}$, while the generalized Fenchel-Young loss is between objects v and p of mixed spaces $\mathcal{V}$ and $\mathcal{C}$.
- If $\Phi(v, p) - \Omega(p)$ is concave in p, then $D^{\Phi}_{\Omega}(p, p')$ is convex in p, as is the case of the usual Bregman divergence $D_{\Omega}(p, p')$. However, (19) is not easy to solve globally in general, as it is the maximum of a difference of convex functions in v. This can be done approximately by using the convex-concave procedure [71], linearizing the left part. We have the opposite situation with the generalized Fenchel-Young loss: if $\Phi(v, p) - \Omega(p)$ is convex-concave, $L^{\Phi}_{\Omega}(v, y)$ is easy to compute but it is a difference of convex functions in v.
- Similarly, we may also define $D^{\Psi}_{\Omega^{\Phi}}(v, v') = \Omega^{\Phi}(v) - \Psi(p^{\Phi}_{\Omega}(v'), v) - \Omega^{\Phi}(v') + \Psi(p^{\Phi}_{\Omega}(v'), v')$, where we recall that $p^{\Phi}_{\Omega}(v) := \operatorname{argmax}_{p\in\mathcal{C}} \Phi(v, p) - \Omega(p)$. We have thus obtained a divergence between two objects v and v' in the same space $\mathcal{V}$.

B Proofs

B.1 Proofs for Proposition 1 (properties of generalized conjugates)

1. **Generalized Fenchel-Young inequality.** From (5), we immediately obtain $\Omega^{\Phi}(v) \geq \Phi(v, p) - \Omega(p)$ for all $v \in \mathcal{V}$ and all $p \in \mathcal{C}$.
2. **Convexity.** If $\Phi(v, p)$ is convex in v, then $\Omega^{\Phi}(v)$ is the maximum of a family of convex functions indexed by p. Therefore, $\Omega^{\Phi}(v)$ is convex. Note that this is the case even if $\Omega(p)$ is nonconvex.
3. **Order reversing.** Since $\Omega \leq \Lambda$, we have
$$\Omega^{\Phi}(v) = \max_{p\in\mathcal{C}} \Phi(v, p) - \Omega(p) \geq \max_{p\in\mathcal{C}} \Phi(v, p) - \Lambda(p) = \Lambda^{\Phi}(v).$$
4. **Continuity.** See [64, Box 1.8].
5. **Gradient.** See "Assumptions for envelope theorems" in §4.
6. **Smoothness.** We follow the proof technique of [46, Lemma 4.3], which states that $v \mapsto \max_{p\in\mathcal{C}} E(v, p)$ is $(\beta + \beta^2/\gamma)$-smooth if E is β-smooth in (v, p) and γ-strongly concave in p over $\mathcal{C}$. We show here that if $E(v, p)$ decomposes as $E(v, p) = \Phi(v, p) - \Omega(p)$, where Ω is γ-strongly convex over $\mathcal{C}$ **but possibly nonsmooth** (as is the case for instance of Shannon's negentropy), and $\Phi(v, p)$ is β-smooth in (v, p) and concave in p, then we still have that $\Omega^{\Phi}(v) = \max_{p\in\mathcal{C}} \Phi(v, p) - \Omega(p)$ is $(\beta + \beta^2/\gamma)$-smooth.

   For brevity, let us define the shorthand $p^{\Phi}_{\Omega}(v) := p^{\star}(v)$. From [46, Section A.3], we have
$$\gamma\|p^{\star}(v_2) - p^{\star}(v_1)\|^2 \leq \langle p^{\star}(v_2) - p^{\star}(v_1), \nabla_2 E(v_2, p^{\star}(v_2)) - \nabla_2 E(v_1, p^{\star}(v_2))\rangle$$
   for all $v_1, v_2 \in \mathcal{V}$. With $E(v, p) = \Phi(v, p) - \Omega(p)$, we obtain
$$\gamma\|p^{\star}(v_2) - p^{\star}(v_1)\|^2 \leq \langle p^{\star}(v_2) - p^{\star}(v_1), \nabla_2 \Phi(v_2, p^{\star}(v_2)) - \nabla_2 \Phi(v_1, p^{\star}(v_2))\rangle.$$
   Since Φ is β-smooth, we have for all $p \in \mathcal{C}$
$$\|\nabla_2 \Phi(v_2, p) - \nabla_2 \Phi(v_1, p)\|_* \leq \beta\|(v_2, p) - (v_1, p)\| = \beta\|v_2 - v_1\|.$$

Combined with the Holder inequality $\langle a, b\rangle \leq \|a\|_* \|b\|$, this gives

$$\gamma \|p^\star(v_2) - p^\star(v_1)\|^2 \leq \beta \|p^\star(v_2) - p^\star(v_1)\| \|v_2 - v_1\|.$$

Simplifying, we get

$$\|p^\star(v_2) - p^\star(v_1)\| \leq \frac{\beta}{\gamma} \|v_2 - v_1\|. \tag{21}$$

Therefore, $p_\Omega^\Phi(v) = p^\star(v)$ is β/γ-Lipschitz.

From Rockafellar's envelope theorem [61, Theorem 10.31], $\nabla \Omega^\Phi(v) = \nabla_1 \Phi(v, p^\star(v))$. Since Φ is β-smooth, we therefore have

$$\begin{aligned}
\|\nabla \Omega^\Phi(v_2) - \nabla \Omega^\Phi(v_1)\|_* &= \|\nabla_1 \Phi(v_2, p^\star(v_2)) - \nabla_1 \Phi(v_1, p^\star(v_1))\|_* \\
&\leq \beta \|(v_2 - p^\star(v_2)) - (v_1, p^\star(v_1))\| \\
&\leq \beta (\|v_2 - v_1\| + \|p^\star(v_2) - p^\star(v_1)\|) \\
&\leq \left(\beta + \frac{\beta^2}{\gamma}\right) \|v_2 - v_1\| \\
&\leq 2 \frac{\beta^2}{\gamma} \|v_2 - v_1\|,
\end{aligned}$$

where we used (21) and $\frac{\beta}{\gamma} \geq 1$.

A related result in the context of bilevel programming but with different assumptions and proof is stated in [31, Lemma 2.2].

B.2 Proofs for Proposition 2 (closed forms)

1. **Bilinear coupling.** This follows from

$$\Omega^\Phi(v) = \max_{p \in \mathcal{C}} \langle v, Up \rangle - \Omega(p) = \max_{p \in \mathcal{C}} \langle U^\top v, p \rangle - \Omega(p) = \Omega^*(U^\top v)$$

and similarly for $p_\Omega(v)$.

2. **Linear-quadratic coupling.** Let us define the function

$$F(p) = \frac{\gamma}{2} \|p\|_2^2 - \frac{1}{2} \langle p, Ap \rangle - \langle p, b \rangle = \frac{1}{2} \langle p, (\gamma I - A) p \rangle - \langle p, b \rangle.$$

Its gradient is $\nabla F(p) = (\gamma I - A)p - b$. Setting $\nabla F(p^\star) = 0$, we obtain

$$(\gamma I - A) p^\star = b \Leftrightarrow p^\star = (\gamma I - A)^{-1} b,$$

where we assumed that $(\gamma I - A)$ is positive definite. We therefore get

$$F(p^\star) = \frac{1}{2} \langle p^\star, b \rangle - \langle p^\star, b \rangle = -\frac{1}{2} \langle p^\star, b \rangle = -\frac{1}{2} \langle b, (\gamma I - A)^{-1} b \rangle.$$

3. **Metric coupling.** From (8), it is easy to check that $\Omega = -\Lambda \Leftrightarrow \Omega^\Phi = -\Lambda_C$ with $C = -\Phi$. From [59, Proposition 6.1], we have $\Lambda_C = -\Lambda$. Therefore, $\Omega^\Phi = -\Omega$.

B.3 Proofs for Proposition 3 (Properties of generalized Fenchel-Young losses)

1. **Non-negativity.** This follows immediately from the generalized Fenchel-Young inequality.
2. **Zero loss.** If $p_\Omega^\Phi(v) = y$, then using $L_\Omega^\Phi(v, y) = F(v, y) - F(v, p_\Omega^\Phi(v))$, where $F(v, p) = \Omega(p) - \Phi(v, p)$, we obtain $L_\Omega^\Phi(v, y) = 0$. Let's now prove the reverse direction. Since the maximum in (5) exists, we have $F(v, p_\Omega^\Phi(v)) \leq F(v, y)$. If $L_\Omega^\Phi(v, y) = 0$, we have $F(v, p_\Omega^\Phi(v)) = F(v, y)$. Since the maximum is unique by assumption, this proves that $y = p_\Omega^\Phi(v)$.
3. **Gradient.** This follows directly from the definition (9).
4. **Difference of convex (DC).** If $\Phi(v, p)$ is convex in p, $\Omega^\Phi(v)$ is convex (Proposition 1). Therefore, $v \mapsto \Omega^\Phi(v) - \Phi(v, p)$ for all $p \in \mathcal{C}$.

5. **Smaller output set, smaller loss.** Let $\Omega\colon \mathcal{C} \to \mathbb{R}$ and let Ω' be the restriction of Ω to $\mathcal{C}' \subseteq \mathcal{C}$, i.e., $\Omega'(p) \coloneqq \Omega + I_{\mathcal{C}'}$, where $I_{\mathcal{C}'}$ is the indicator function of $\mathcal{C}'$. From (5), we have $(\Omega')^\Phi(v) \leq \Omega^\Phi(v)$ for all $v \in \mathcal{V}$. From (9), we therefore have $L^\Phi_{\Omega'}(v,p) \leq L^\Phi_\Omega(v,p)$ for all $v \in \mathcal{V}$ and all $p \in \mathcal{C}'$.

6. **Quadratic lower-bound.** If a function F is γ-strongly convex over $\mathcal{C}$ w.r.t. a norm $\|\cdot\|$, then

$$\frac{\gamma}{2}\|p - p'\|^2 \leq F(p) - F(p') - \langle \nabla F(p'), p - p' \rangle \tag{22}$$

for all $p, p' \in \mathcal{C}$. If $\mathcal{C}$ is a closed convex set, we also have that $p^\star = \operatorname{argmin}_{p\in\mathcal{C}} F(p)$ satisfies the optimality condition

$$\langle \nabla F(p^\star), p - p^\star \rangle \geq 0 \tag{23}$$

for all $p \in \mathcal{C}$ [52, Eq. (2.2.13)]. Combining (22) with $p' = p^\star$ and (23), we obtain

$$\frac{\gamma}{2}\|p - p^\star\|^2 \leq F(p) - F(p^\star).$$

Applying the above with $F(p) = \Omega(p) - \Phi(v,p)$ and using $p^\star = p^\Phi_\Omega(v)$, we obtain

$$\frac{\gamma}{2}\|y - p^\Phi_\Omega(v)\|^2 \leq L^\Phi_\Omega(v,y).$$

7. **Upper-bounds.** If F is α-Lipschitz over $\mathcal{C}$ with respect to a norm $\|\cdot\|$, then for all $p, p' \in \mathcal{C}$

$$|F(p) - F(p')| \leq \alpha\|p - p'\|.$$

Applying the above with $F(p) = \Omega(p) - \Phi(v,p)$ and $p' = p^\Phi_\Omega(v)$, we obtain

$$L^\Phi_\Omega(v,p) \leq \alpha\|p - p^\Phi_\Omega(v)\|.$$

If $\Phi(v,p)$ is concave in p, then its linear approximation always lies above:

$$\Phi(v,p') \leq \Phi(v,p) + \langle \nabla_2\Phi(v,p), p' - p \rangle,$$

for all $p, p' \in \mathcal{C}$. We then have

$$\begin{aligned}
L^\Phi_\Omega(v,p) &= \Omega^\Phi(v) + \Omega(p) - \Phi(v,p) \\
&= \max_{p'\in\mathcal{C}} \ \Phi(v,p') - \Omega(p') + \Omega(p) - \Phi(v,p) \\
&\leq \max_{p'\in\mathcal{C}} \ \langle \nabla_2\Phi(v,p), p' \rangle - \Omega(p') + \Omega(p) - \langle \nabla_2\Phi(v,p), p \rangle \\
&= \Omega^*(\nabla_2\Phi(v,p)) + \Omega(p) - \langle \nabla_2\Phi(v,p), p \rangle \\
&= L_\Omega(\nabla_2\Phi(v,p), p).
\end{aligned}$$

B.4 Proof of Proposition 4 (calibration)

Background. The pointwise target risk of $\widehat{y} \in \mathcal{Y}$ according to $q \in \triangle^{|\mathcal{Y}|}$ is

$$\ell(\widehat{y}, q) \coloneqq \mathbb{E}_{Y\sim q}\, L(\widehat{y}, Y).$$

We also define the corresponding excess of pointwise risk, the difference between the pointwise risk and the pointwise Bayes risk:

$$\delta\ell(\widehat{y}, q) \coloneqq \ell(\widehat{y}, q) - \min_{y'\in\mathcal{Y}} \ell(y', q).$$

We can then write the risk of $f\colon \mathcal{X} \to \mathcal{Y}$ in terms of the pointwise risk

$$\begin{aligned}
\mathcal{L}(f) &\coloneqq \mathbb{E}_{(X,Y)\sim\rho}\, L(f(X), Y) \\
&= \mathbb{E}_{X\sim\rho_\mathcal{X}} \mathbb{E}_{Y\sim\rho(\cdot|X)} L(f(X), Y) \\
&= \mathbb{E}_{X\sim\rho_\mathcal{X}} \ell(f(X), \rho(\cdot|X)).
\end{aligned}$$

Let us define the Bayes predictor

$$f^\star \coloneqq \operatorname*{argmin}_{f\colon \mathcal{X}\to\mathcal{Y}} \mathcal{L}(f).$$

The Bayes risk is then

$$\begin{aligned}\mathcal{L}(f^\star) &= \min_{f\colon \mathcal{X}\to\mathcal{Y}} \mathbb{E}_{X\sim\rho_\mathcal{X}} \mathbb{E}_{Y\sim\rho(\cdot|X)} L(f(X),Y)\\ &= \mathbb{E}_{X\sim\rho_\mathcal{X}} \min_{y'\in\mathcal{Y}} \mathbb{E}_{Y\sim\rho(\cdot|X)} L(y',Y)\\ &= \mathbb{E}_{X\sim\rho_\mathcal{X}} \min_{y'\in\mathcal{Y}} \ell(y',\rho(\cdot|X)).\end{aligned}$$

Combining the above, we can write the excess of risk of $f\colon \mathcal{X}\to\mathcal{Y}$ as

$$\mathcal{L}(f) - \mathcal{L}(f^\star) = \mathbb{E}_{X\sim\rho_\mathcal{X}} \delta\ell(f(X),\rho(\cdot|X)).$$

Similarly, with the generalized Fenchel-Young loss (9), the pointwise surrogate risk of $v \in \mathcal{V}$ according to $q \in \triangle^{|\mathcal{Y}|}$ is

$$\ell_\Omega^\Phi(v,q) \coloneqq \mathbb{E}_{Y\sim q}\, L_\Omega^\Phi(v,Y)$$

and the excess of pointwise surrogate risk is

$$\delta\ell_\Omega^\Phi(v,q) \coloneqq \ell_\Omega^\Phi(v,q) - \min_{v'\in\mathcal{V}} \ell_\Omega^\Phi(v',q).$$

Let us define the surrogate risk of $g\colon \mathcal{X}\to\mathcal{V}$ by

$$\mathcal{L}_\Omega^\Phi(g) \coloneqq \mathbb{E}_{(X,Y)\sim\rho}\, L_\Omega^\Phi(g(X),Y)$$

and the corresponding Bayes predictor by

$$g^\star \coloneqq \operatorname*{argmin}_{g\colon \mathcal{X}\to\mathcal{V}} \mathcal{L}_\Omega^\Phi(g).$$

We can then write the excess of surrogate risk of $g\colon \mathcal{X}\to\mathcal{V}$ as

$$\mathcal{L}_\Omega^\Phi(g) - \mathcal{L}_\Omega^\Phi(g^\star) = \mathbb{E}_{X\sim\rho_\mathcal{X}} \delta\ell_\Omega^\Phi(g(X),\rho(\cdot|X)).$$

A calibration function [67] $\xi\colon \mathbb{R}_+\to\mathbb{R}_+$ is a function relating the excess of pointwise target risk and pointwise surrogate risk. It should be non-negative, convex and non-decreasing on $\mathbb{R}_+$, and satisfy $\xi(0)=0$. Formally, given a decoder $d\colon \mathcal{V}\to\mathcal{Y}$, ξ should satisfy

$$\xi(\delta\ell(d(v),q)) \le \delta\ell_\Omega^\Phi(v,q) \tag{24}$$

for all $v\in\mathcal{V}$ and $q\in\triangle^{|\mathcal{Y}|}$. By Jensen's inequality, this implies that the target and surrogate risks are calibrated for all $g\colon \mathcal{X}\to\mathcal{V}$ [58, 55]

$$\xi(\mathcal{L}(d\circ g) - \mathcal{L}(f^\star)) \le \mathcal{L}_\Omega^\Phi(g) - \mathcal{L}_\Omega^\Phi(g^\star).$$

From now on, we can therefore focus on proving (24).

Upper-bound on the pointwise target excess risk. We now make use of the affine decomposition (12). Let $\sigma \coloneqq \sup_{y\in\mathcal{Y}} \|V^\top\varphi(y)\|_*$, where $\|\cdot\|_*$ denotes the dual norm of $\|\cdot\|$. Let us define

$$\tilde{y}_L(u) \coloneqq \operatorname*{argmin}_{\widehat{y}\in\mathcal{Y}} \langle \varphi(\widehat{y}), Vu+b\rangle$$

and $\mu_\varphi(q) \coloneqq \mathbb{E}_{Y\sim q}[\varphi(Y)] \in \operatorname{conv}(\varphi(\mathcal{Y})) \subseteq \varphi(\mathbb{R}^k)$. From [16, Lemma 2],

$$\delta\ell(\tilde{y}_L(u),q) \le 2\sigma\|\mu_\varphi(q)-u\| \quad \forall u\in\varphi(\mathbb{R}^k), q\in\triangle^{|\mathcal{Y}|}.$$

Using $y_L(p)=\tilde{y}_L(u)$ with $u=\varphi(p)$, we thus get

$$\delta\ell(y_L(p),q) \le 2\sigma\|\mu_\varphi(q)-\varphi(p)\| \quad \forall p\in\mathbb{R}^k, q\in\triangle^{|\mathcal{Y}|}. \tag{25}$$

Bound on the pointwise surrogate excess risk (bilinear case). To highlight the difference with our proof technique, we first prove the result in the bilinear case, assuming Ω is γ-strongly convex. We follow the same proof technique as [55, 16] but unlike these works we do not require any Legendre-type assumption on Ω. If $\Phi(v,p) = \langle v, \varphi(p)\rangle = \langle v, p\rangle$, we have

$$\begin{aligned}
\ell_\Omega^\Phi(v,q) &= \mathbb{E}_{Y\sim q}\, L_\Omega(v,Y) \\
&= \Omega^*(v) + \mathbb{E}_{Y\sim q}\Omega(Y) - \langle v, \mu(q)\rangle \\
&= \Omega^*(v) + \Omega(\mu(q)) - \langle v, \mu(q)\rangle + \mathbb{E}_{Y\sim q}\Omega(Y) - \Omega(\mu(q)) \\
&= L_\Omega(v,\mu(q)) + \mathbb{E}_{Y\sim q}\Omega(Y) - \Omega(\mu(q)),
\end{aligned}$$

where, when $\varphi(y) = y$, we denote $\mu(q) \coloneqq \mathbb{E}_{Y\sim q}[Y] \in \operatorname{conv}(\mathcal{Y}) \subseteq \mathbb{R}^k$ for short. The quantity $\mathbb{E}_{Y\sim q}\Omega(Y) - \Omega(\mu(q))$ is called Bregman information [7] or Jensen gap, and is non-negative if Ω is convex. This term cancels out in the excess of pointwise surrogate risk

$$\delta\ell_\Omega(v,q) \coloneqq \ell_\Omega(v,q) - \min_{v'\in\mathcal{V}} \ell_\Omega(v',q) = L_\Omega(v,\mu(q)) - \min_{v'\in\mathcal{V}} L_\Omega(v',\mu(q)).$$

Since the Fenchel-Young loss achieves its minimum at 0, we have

$$\delta\ell_\Omega(v,q) = L_\Omega(v,\mu(q)).$$

Therefore, the excess of pointwise surrogate risk can be written in Fenchel-Young loss form. By the quadratic lower-bound (10) and the upper-bound (25), we have

$$\delta\ell_\Omega(v,q) = L_\Omega(v,\mu(q)) \geq \frac{\gamma}{2}\|\mu(q) - p_\Omega(v)\|^2 \geq \frac{\gamma}{8\sigma^2}\delta\ell(y_L(p_\Omega(v)),q))^2.$$

Therefore the calibration function with the decoder $d = y_L \circ p_\Omega$ is

$$\xi(\varepsilon) = \frac{\gamma\varepsilon^2}{8\sigma^2}.$$

Bound on the pointwise surrogate excess risk (linear-concave case). We now prove the bound assuming that L_Ω^Φ is smooth and $\Phi(v,p) = \langle v, \varphi(p)\rangle$. This includes the previous proof as special case because when $\Phi(v,p) = \langle v,p\rangle$, then $L_\Omega^\Phi(v,y) = L_\Omega(v,y)$ is $\frac{1}{\gamma}$-smooth in v if and only if $\Omega(p)$ is γ-strongly convex in p (cf. §2).

By Theorem 4.22 in [56], if a function $f(v)$ is M-smooth in v w.r.t. the dual norm $\|\cdot\|_*$ and is bounded below, then

$$f(v) - \min_{v'\in\mathcal{V}} f(v') \geq \frac{1}{2M}\|\nabla f(v)\|^2.$$

With $f(v) = \ell_\Omega^\Phi(v,q) = \mathbb{E}_{Y\sim q}\, L_\Omega(v,Y)$, which is non-negative, we obtain

$$\begin{aligned}
\delta\ell_\Omega^\Phi(v,q) &\coloneqq \ell_\Omega^\Phi(v,q) - \min_{v'\in\mathcal{V}} \ell_\Omega^\Phi(v',q) \\
&\geq \frac{1}{2M}\|\nabla_1 \ell_\Omega^\Phi(v,q)\|^2 \\
&= \frac{1}{2M}\|\mathbb{E}_{Y\sim q}\nabla_1 L_\Omega^\Phi(v,Y)\|^2 \\
&= \frac{1}{2M}\|\nabla\Omega^\Phi(v) - \mathbb{E}_{Y\sim q}\nabla_1\Phi(v,Y)\|^2 \\
&= \frac{1}{2M}\|\nabla_1\Phi(v,p_\Omega^\Phi(v)) - \mathbb{E}_{Y\sim q}\nabla_1\Phi(v,Y)\|^2 \\
&= \frac{1}{2M}\|\varphi(p_\Omega^\Phi(v)) - \mathbb{E}_{Y\sim q}\varphi(Y)\|^2 \\
&= \frac{1}{2M}\|\varphi(p_\Omega^\Phi(v)) - \mu_\varphi(q)\|^2.
\end{aligned}$$

We therefore have

$$\delta\ell_\Omega^\Phi(v,q) \geq \frac{1}{2M}\|\mu_\varphi(q) - \varphi(p_\Omega^\Phi(v))\|^2 \geq \frac{1}{8\sigma^2 M}\delta\ell(y_L(p_\Omega^\Phi(v)),q))^2.$$

Therefore the calibration function with the decoder $d = y_L \circ p_\Omega^\Phi$ is

$$\xi(\varepsilon) = \frac{\varepsilon^2}{8\sigma^2 M}.$$

B.5 Proofs for Proposition 5 (Properties of generalized biconjugates)

The proofs are similar to that for C-transforms [64, Proposition 1.34].

1. **Lower bound.** Let $\Omega\colon \mathcal{C} \to \mathbb{R}$. We have

$$\begin{aligned}
\Omega^{\Phi\Psi}(p) &\coloneqq \max_{v\in\mathcal{V}} \Psi(p,v) - \Omega^{\Phi}(v) \\
&= \max_{v\in\mathcal{V}} \Phi(v,p) - \Omega^{\Phi}(v) \\
&= \max_{v\in\mathcal{V}} \Phi(v,p) - \left[\max_{p'\in\mathcal{C}} \Phi(v,p') - \Omega(p')\right] \\
&\leq \max_{v\in\mathcal{V}} \Phi(v,p) - \Phi(v,p) + \Omega(p) \\
&= \Omega(p).
\end{aligned}$$

   Therefore, $\Omega^{\Phi\Psi}(p) \leq \Omega(p)$ for all $p \in \mathcal{C}$. Analogously, for a function $\Lambda\colon \mathcal{V} \to \mathbb{R}$, we have $\Lambda^{\Psi\Phi}(v) \leq \Lambda(v)$ for all $v \in \mathcal{V}$.

2. **Equality.** Since Ω is Ψ-convex, there exits $\Lambda\colon \mathcal{V} \to \mathbb{R}$ such that $\Omega = \Lambda^{\Psi}$. We then have $\Omega^{\Phi} = \Lambda^{\Psi\Phi}$. Using the lower bound property, we get $\Omega^{\Phi}(v) = \Lambda^{\Psi\Phi}(v) \leq \Lambda(v)$ for all $v \in \mathcal{V}$. By the order reversing property, we have $\Omega^{\Phi\Psi}(p) \geq \Lambda^{\Psi}(p) = \Omega(p)$ for all $p \in \mathcal{C}$. However, we also have $\Omega^{\Phi\Psi}(p) \leq \Omega(p)$ for all $p \in \mathcal{C}$. Therefore, $\Omega^{\Phi\Psi}(p) = \Omega(p)$ for all $p \in \mathcal{C}$.

3. **Tightest lower-bound.** Let Ω' be any lower bound of Ω that is Ψ-convex. Therefore, there exists $\Lambda\colon \mathcal{V} \to \mathbb{R}$ such that $\Omega'(p) = \Lambda^{\Psi}(p) \leq \Omega(p)$ for all $p \in \mathcal{C}$. By the order reversing property, we have $\Lambda^{\Psi\Phi}(v) \geq \Omega^{\Phi}(v)$ for all $v \in \mathcal{V}$. By the lower bound property, we also have $\Lambda^{\Psi\Phi}(v) \leq \Lambda(v)$ for all $v \in \mathcal{V}$ and therefore, $\Lambda(v) \geq \Omega^{\Phi}(v)$ for all $v \in \mathcal{V}$. Applying the order reversing property once more, we get $\Omega'(p) = \Lambda^{\Psi}(p) \leq \Omega^{\Phi\Psi}(p)$ for all $p \in \mathcal{C}$. Therefore $\Omega^{\Phi\Psi}$ is the tightest lower bound of Ω.

B.6 Proofs for Proposition 6 (Properties of generalized Bregman divergences)

1. **Link with generalized Fenchel-Young losses.** From (16) and (17), we have

$$\Lambda^{\Psi}(p) = \Phi(v_{\Lambda}^{\Psi}(p), p) - \Lambda(v_{\Lambda}^{\Psi}(p)) \quad \forall p \in \mathcal{C}$$

   for any $\Lambda\colon \mathcal{V} \to \mathbb{R}$. With $\Lambda = \Omega^{\Phi}$, if Ω is Φ-convex, using Proposition 5, we have

$$\Omega^{\Phi\Psi}(p) = \Omega(p) = \Phi(v_{\Lambda}^{\Psi}(p), p) - \Lambda(v_{\Lambda}^{\Psi}(p)) \quad \forall p \in \mathcal{C}.$$

   Plugging $\Omega(p')$ in (18) and using the shorthand $v \coloneqq v_{\Omega^{\Phi}}^{\Psi}(p')$, we get

$$D_{\Omega}^{\Phi}(p,p') = \Omega(y) - \Phi(v,p) + \Omega^{\Phi}(v) = L_{\Omega}^{\Phi}(v,p).$$

2. **Non-negativity.** This follows directly from the non-negativity of $L_{\Omega}^{\Phi}(v,p)$.

3. **Identity of indiscernibles**. If $p = p'$, we immediately obtain $D_{\Omega}^{\Phi}(p,p') = 0$ from (18). Let's prove the reverse direction. If $D_{\Omega}^{\Phi}(p,p') = 0$, we have $F(v_{\Omega^{\Phi}}^{\Psi}(p'), p) = F(v_{\Omega^{\Phi}}^{\Psi}(p'), p')$ where $F(v,p) = \Omega(p) - \Phi(v,p)$. Since by assumption F is strictly convex in p, we obtain $p = p'$.

4. **Convexity.** This follows from $D_{\Omega}^{\Phi}(p,p') = \Omega(p) - \Phi(v,p) + \text{const}$.

5. **Recovering Bregman divergences.** If $\Phi(v,p) = \langle v, p\rangle$, we have

$$v_{\Omega^{\Phi}}^{\Psi}(p) = \operatorname*{argmax}_{v\in\mathcal{V}} \Phi(v,p) - \Omega^{\Phi}(v) = \operatorname*{argmax}_{v\in\mathcal{V}} \langle v, p\rangle - \Omega^{*}(v) = \nabla\Omega(p).$$

   Plugging back in (18), we obtain the Bregman divergence (20).

C Experimental details

C.1 Multilabel classification

Datasets. We used public datasets available at https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/. Dataset statistics are summarized in Table 3. For all datasets, we normalize samples to have zero mean unit variance.

Table 3: Multilabel dataset statistics.

Dataset	Type	Train	Dev	Test	Features	Classes	Avg. labels
Birds	Audio	134	45	172	260	19	1.96
Cal500	Music	376	126	101	68	174	25.98
Emotions	Music	293	98	202	72	6	1.82
Mediamill	Video	22,353	7,451	12,373	120	101	4.54
Scene	Images	908	303	1,196	294	6	1.06
Yeast	Micro-array	1,125	375	917	103	14	4.17

Experimental details. In all experiments, we set the activation σ to $\text{relu}(a) := \max\{0, a\}$.

For the unary model, we use a neural network with one hidden-layer, i.e., $g_\theta(x) = W_2\sigma(W_1 x + b_1) + b_2$, where $\theta = (W_2, b_2, W_1, b_1)$, $W_2 \in \mathbb{R}^{k\times m}$, $b_2 \in \mathbb{R}^k$, $W_1 \in \mathbb{R}^{m\times d}$, $b_1 \in \mathbb{R}^m$, and m is the number of hidden units. We use the heuristic $m = \min\{100, d/3\}$, where d is the dimensionality of x.

For the pairwise model, in order to obtain a negative semi-definite matrix U, we parametrize $U = -AA^\top$ with $A = [W_1 x + b_1, \dots, W_m x + b_m]$, where $W_j \in \mathbb{R}^{k\times d}$ and $b_j \in \mathbb{R}^k$ In our experiments, we choose a rank-one model, i.e., $m = 1$. Note that we use distinct parameters for the unary and pairwise models but sharing parameters would be possible.

For the SPEN model, following [11, Eq. 4 and 5], we set the energy to $\Phi(v, p) = \langle u, p\rangle - \Psi(w, p)$, where $v = (u, w)$, $u = g_\theta(x)$ and w are the weights of the "prior network" Ψ (independent of x). We parametrize $\Psi(w, p) = W_2\sigma(W_1 p + b_1) + b_2$, where $w = (W_2, b_2, W_1, b_1)$, $W_2 \in \mathbb{R}^{1\times m}$, $b_2 \in \mathbb{R}^1$ and $W_1 \in \mathbb{R}^{m\times k}$, $b_1 \in \mathbb{R}^m$. To further impose convexity of Ψ in p, W_2 needs to be non-negative. To do so without using constrained optimization, we use the change of variable $W_2 = \text{softplus}(W_2')$, where $\text{softplus}(a) := \log(1 + \exp(a))$ is used as an element-wise bijective mapping.

Additional results. The gradient of $\Omega^\Phi(v)$ can be computed using the envelope theorem (Proposition 1), which does not require to differentiate through $p_\Omega^\Phi(v)$. Alternatively, since $\Omega^\Phi(v) = \Phi(v, p_\Omega^\Phi(v)) - \Omega(p_\Omega^\Phi(v))$, we can also compute the gradient of $\Omega^\Phi(v)$ by using the implicit function theorem, differentiating through $p_\Omega^\Phi(v)$. To do so, we use the approach detailed in [17]. Results in Table 4 show that the envelope theorem performs comparably to the implicit function theorem, if not slightly better. Differentiating through $p_\Omega^\Phi(v)$ using the implicit function theorem requires to solve a $k \times k$ system and its implementation is more complicated than the envelope theorem. Therefore, we suggest to use the envelope theorem in practice.

Table 4: Comparison of envelope and implicit function theorems on the pairwise model (test accuracy in %).

	yeast	scene	mediamill	birds	emotions	cal500
Envelope theorem	80.19	**91.58**	**96.95**	**91.55**	**80.56**	**85.73**
Implicit function theorem	**80.33**	**91.58**	**96.95**	91.54	80.53	85.57

In addition, we also compared the proposed generalized Fenchel-Young loss with the energy loss, the binary cross-entropy loss and the generalized perceptron loss, which corresponds to setting $\Omega(p) = 0$. As explained in §3, the cross-entropy loss requires to differentiate through $p_\Omega^\Phi(v)$; we do so by implicit differentiation. Table 5 shows that the generalized Fenchel-Young loss outperforms these losses. As expected, the energy loss performs very poorly, as it can only push the model in one direction [44]. Using regularization Ω, as advocated in this paper, is empirically confirmed to be beneficial for accuracy.

Table 5: Comparison of loss functions for the pairwise model (accuracy in %).

	yeast	scene	mediamill	birds	emotions	cal500
Generalized FY loss	**80.19**	**91.58**	**96.95**	91.55	**80.56**	85.73
Energy loss	42.35	33.02	40.92	14.29	55.50	39.27
Cross-entropy loss	79.00	90.78	96.77	**91.56**	78.08	**85.89**
Generalized perceptron loss	68.36	89.33	93.24	88.92	66.34	80.11

C.2 Imitation learning

Experimental details. We run a hyperparameter search over the learning rate of the ADAM optimizer, the number of hidden units in the layers, the weight of the L2 parameters regularization term and the scale of the energy regularization term Ω. We run the hyperparameter search for 4 demonstration trajectories and select the best performing ones based on the final performance (averaged over 3 seeds).

Table 6: Hyperparameter search for imitation learning.

Model	Learning rate {1e-4, 5e-4, 1e-3}	Params regularization {0., 1., 10.}	Energy regularization {0.1, 1., 10.}	Hidden units {16, 32, 64, 128}
Unary	5e-4	0.0	1.	16
Pairwise	1e-4	0.0	10.	32

Environments. We also provide performance of the expert agent as detailed by Orsini et al. [57] as well as the description of the observation and action spaces for each environment.

Table 7: Dimension of observation space, dimension of action space, expert performance, and random policy performance for each environment.

Task	Observations	Actions	Random policy score	Expert score
HalfCheetah-v2	17	6	-282	8770
Hopper-v2	11	3	18	2798
Walker-v2	17	6	1.6	4118
Ant-v2	111	8	123	5637