Loss Functions and Operators Generated by f-Divergences
由 f-散度生成的损失函数和算子

Vincent Roulet [1] Tianlin Liu [1] Nino Vieillard [1] Michaël E. Sander [1] Mathieu Blondel [1]

Abstract

The logistic loss (a.k.a. cross-entropy loss) is one of the most popular loss functions used for multiclass classification. It is also the loss function of choice for next-token prediction in language modeling. It is associated with the Kullback–Leibler (KL) divergence and the softargmax operator. In this work, we propose to construct new convex loss functions based on f-divergences. Our loss functions generalize the logistic loss in two directions: i) by replacing the KL divergence with f-divergences and ii) by allowing non-uniform reference measures. We instantiate our framework for numerous f-divergences, recovering existing losses and creating new ones. By analogy with the logistic loss, the loss function generated by an f-divergence is associated with an operator, that we dub f-softargmax. We derive a novel parallelizable bisection algorithm for computing the f-softargmax associated with any f-divergence. On the empirical side, one of the goals of this paper is to determine the effectiveness of loss functions beyond the classical cross-entropy in a language model setting, including on pre-training, post-training (SFT) and distillation. We show that the loss function generated by the α-divergence (which is equivalent to Tsallis α-negentropy in the case of unit reference measures) with $\alpha = 1.5$ performs well across several tasks.

逻辑损失（a.k.a. 交叉熵损失）是用于多类分类的最流行的损失函数之一。它也是语言建模中下一个 token 预测的首选损失函数。它与 Kullback-Leibler (KL) 散度和 softargmax 运算符相关联。在这项工作中，我们提出基于 f-散度构建新的凸损失函数。我们的损失函数从两个方向概括了逻辑损失：i) 用 f-散度替换 KL 散度；ii) 允许非统一参考度量。我们实例化了用于众多 f-散度的框架，恢复现有损失并创建新损失。与逻辑损失类似，由 f-散度生成的损失函数与一个运算符相关联，我们将其称为 f-softargmax。我们推导出一种新的可并行二分算法，用于计算与任何 f 散度相关的 f-softargmax。在经验方面，本文的目标之一是确定语言模型设置中超越经典交叉熵的损失函数的有效性，包括在训练前、训练后 (SFT) 和提炼方面。我们表明，由 α 散度（在单位参考度量的情况下相当于 Tsallis α-负熵）生成的损失函数在 $\alpha = 1.5$ 时在多个任务中表现良好。逻辑损失（又名交叉熵损失）是用于多类分类的最流行的损失函数之一。它也是语言建模中下一个 token 预测的首选损失函数。它与 Kullback-Leibler (KL) 散度和 softargmax 运算符相关。在这项工作中，我们提出基于 f 散度构建新的凸损失函数。我们的损失函数从两个方面概括了逻辑损失：i) 用 f 散度替换 KL 散度；ii) 允许非统一参考度量。我们实例化了众多 f 散度的框架，恢复现有损失并创建新损失。与逻辑损失类似，由 f 散度生成的损失函数与一个运算符相关联，我们将其称为 f-softargmax。我们推导出一种新的可并行二分算法，用于计算与任何 f 散度相关的 f-softargmax。在实证方面，本文的目标之一是确定语言模型设置中超越经典交叉熵的损失函数的有效性，包括预训练、后训练 (SFT) 和蒸馏。我们表明，由 α-divergence（在单位参考度量的情况下相当于 Tsallis α-negentropy）生成的损失函数在 $\alpha = 1.5$ 的多个任务中表现良好。

[1]Google DeepMind. Correspondence to: Mathieu Blondel <mblondel@google.com>.

Preprint.

1. Introduction

The logistic loss, a.k.a. cross-entropy loss, is widely used for multiclass classification. It is associated with the softargmax (a.k.a. softmax) operator, which turns logits into class probabilities. The logistic loss and the softargmax operators are also frequently used in the space of tokens for language model pre-training, post-training and distillation.
逻辑损失，又称交叉熵损失，广泛用于多类分类。它与 softargmax（又称 softmax）运算符相关联，后者将 logit 转换为类概率。逻辑损失和 softargmax 运算符也经常用于语言模型预训练、后训练和提炼的 token 空间中。

The classical softargmax is known to optimize a trade-off between the expected value of the logits and the Kullback–Leibler (KL) divergence with a uniform measure. This perspective is also adopted in the space of sequences, replacing the uniform measure with a reference measure, in reinforcement learning from human feedback (RLHF) (Christiano et al., 2017) and in direct preference optimization (DPO) (Rafailov et al., 2024). Recent works extended this perspective to general f-divergences (Go et al., 2023; Wang et al., 2024). However, these works naively compose an f-divergence with the (classical) softargmax, which does not result in a convex loss function.
众所周知，经典的 softargmax 以均匀度量优化了 logits 的期望值和 Kullback-Leibler (KL) 散度之间的权衡。在序列空间中也采用了这种观点，用参考度量代替了均匀度量，在从人类反馈进行强化学习 (RLHF) (Christiano et al., 2017) 和直接偏好优化 (DPO) (Rafailov et al., 2024) 中也是如此。最近的研究将这种观点扩展到一般的 f-散度 (Go et al., 2023; Wang et al., 2024)。然而，这些研究天真地用 (经典) softargmax 组成了一个 f-散度，这不会导致凸损失函数。

In this work, building upon Fenchel–Young losses (Blondel et al., 2020), we propose to construct new convex losses based on f-divergences. Our losses generalize the logistic loss in two directions: i) by replacing the KL divergence with f-divergences and ii) by allowing non-uniform reference measures (class prior probabilities). Our loss construction generalizes the sparsemax (Martins & Astudillo, 2016) and entmax (Peters et al., 2019) losses, and allows us to create entirely new losses; see Table 1. In addition, each loss generated by an f-divergence is associated with a new operator for turning logits into probabilities, that we dub f-softargmax.
在这项工作中，我们基于 Fenchel-Young 损失 (Blondel et al., 2020)，我们提出基于 f 散度构建新的凸损失。我们的损失在两个方向上概括了逻辑损失：i) 用 f 散度替换 KL 散度；ii) 允许非均匀参考度量（类先验概率）。我们的损失构造概括了 sparsemax (Martins & Astudillo, 2016) 和 entmax (Peters et al., 2019) 损失，并允许我们创建全新的损失；参见表 1。此外，由 f 散度生成的每个损失都与一个将逻辑转换为概率的新运算符相关联，我们将其称为 f-softargmax。

On the empirical side, one of the goals of this paper is to determine the effectiveness of loss functions beyond the classical cross-entropy in a language model setting, including on pre-training, post-training (SFT) and distillation.
从实证方面来看，本文的目标之一是确定语言模型设置中超越经典交叉熵的损失函数的有效性，包括预训练、后训练（SFT）和提炼。

To summarize, we make the following contributions.
总而言之，我们做出以下贡献。

- We propose to use f-divergence regularization to generate loss functions. Each loss function is associated with a new operator, that we dub f-softargmax. We instantiate this framework for numerous f-divergences, recovering existing losses and creating new ones.
  我们建议使用 f-散度正则化来生成损失函数。每个损失函数都与一个新运算符相关联，我们将其称为 f-softargmax。我们为众多 f-散度实例化此框架，恢复现有损失并创建新损失。
- We derive a novel bisection algorithm for computing the f-softargmax associated with any f-divergence. Our algorithm parallelizes well on modern hardware.
  我们推导出一种新颖的二分算法，用于计算与任何 f-divergence 相关的 f-softargmax。我们的算法在现代硬件上可以很好地并行化。
- We demonstrate our loss functions on image classification, language model post-training and distillation.
  我们在图像分类、语言模型后训练和提炼方面展示了我们的损失函数。

Notation. Throughout this paper, k denotes the number of classes in a classification problem or the vocabulary size in language modeling. We denote the set $[k] \coloneqq \{1, \dots, k\}$. We denote the probability simplex by $\triangle^k \coloneqq \{\boldsymbol{p} \in \mathbb{R}_+^k \colon \langle \boldsymbol{p}, \mathbf{1} \rangle = 1\}$. We denote the domain of f by $\mathrm{dom}(f) \coloneqq \{u \colon -\infty < f(u) < \infty\}$. We denote the convex conjugate of f by f^*, where $f^*(v) \coloneqq \sup_u uv - f(u)$. We denote by $\boldsymbol{\theta} \coloneqq h_{\boldsymbol{w}}(\boldsymbol{x})$ the logits produced by a network h with parameters $\boldsymbol{w} \in \mathcal{W}$ for the input $\boldsymbol{x} \in \mathcal{X}$. We denote hard (one-hot) labels as $\boldsymbol{y} \in \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_k\}$, which are used for classification. More generally, we denote soft labels as $\boldsymbol{y} \in \triangle^k$, which are useful for learning from label proportions, as in distillation.

在本文中，k 表示分类问题中的类别数量或语言建模中的词汇量。我们用集合 $[k] \coloneqq \{1, \dots, k\}$ 表示集合。我们用 $\triangle^k \coloneqq \{\boldsymbol{p} \in \mathbb{R}_+^k \colon \langle \boldsymbol{p}, \mathbf{1} \rangle = 1\}$ 表示概率单纯形。我们用 $\mathrm{dom}(f) \coloneqq \{u \colon -\infty < f(u) < \infty\}$ 表示 f 的定义域。我们用 f^* 表示 f 的凸共轭，其中 $f^*(v) \coloneqq \sup_u uv - f(u)$。我们用 $\boldsymbol{\theta} \coloneqq h_{\boldsymbol{w}}(\boldsymbol{x})$ 表示由网络 h 生成的对数，网络参数为 $\boldsymbol{w} \in \mathcal{W}$，输入为 $\boldsymbol{x} \in \mathcal{X}$。我们将硬（独热）标签表示为 $\boldsymbol{y} \in \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_k\}$，用于分类。更一般地，我们将软标签表示为 $\boldsymbol{y} \in \triangle^k$，它们可用于从标签比例中学习，就像在蒸馏中一样。

2. Background

2.1. Logistic loss

Given logits $\boldsymbol{\theta} \in \mathbb{R}^k$, we define the softargmax as

给定 logits $\boldsymbol{\theta} \in \mathbb{R}^k$，我们将 softargmax 定义为

$$[\mathrm{softargmax}(\boldsymbol{\theta})]_j \coloneqq \frac{\exp(\theta_j)}{\sum_{j'=1}^k \exp(\theta_{j'})} \quad j \in [k]. \tag{1}$$

Note that (1) is commonly known in the literature as "softmax," which is a misnomer as it is a smooth approximation of the argmax function (Blondel & Roulet, 2024). We instead define the softmax as

请注意，(1) 在文献中通常称为“softmax”，这是一个误称，因为它是 argmax 函数 (Blondel & Roulet, 2024) 的平滑近似。我们将 softmax 定义为

$$\mathrm{softmax}(\boldsymbol{\theta}) \coloneqq \log \sum_{j=1}^k \exp(\theta_j),$$

which is a smooth approximation of the maximum function. The softargmax and softmax are related by

这是最大函数的平滑近似。softargmax 和 softmax 之间的关系为

$$\nabla \mathrm{softmax} = \mathrm{softargmax}.$$

The logistic loss (a.k.a. cross-entropy loss) between logits $\boldsymbol{\theta} \in \mathbb{R}^k$ and a ground-truth $\boldsymbol{y} \in \triangle^k$ is often defined as

logits $\boldsymbol{\theta} \in \mathbb{R}^k$ 与真实值 $\boldsymbol{y} \in \triangle^k$ 之间的逻辑损失（又称交叉熵损失）通常定义为

$$-\sum_{j=1}^k y_j \log\Big([\mathrm{softargmax}(\boldsymbol{\theta})]_j\Big) = \mathrm{softmax}(\boldsymbol{\theta}) - \langle \boldsymbol{\theta}, \boldsymbol{y} \rangle.$$

The logistic loss is equivalent up to a constant to the KL divergence between the target and the softargmax output

逻辑损失等于目标和 softargmax 输出之间的 KL 散度的一个常数，

$$\begin{aligned} \ell(\boldsymbol{\theta}, \boldsymbol{y}) &\coloneqq \mathrm{KL}(\boldsymbol{y}, \mathrm{softargmax}(\boldsymbol{\theta})) \\ &= \mathrm{softmax}(\boldsymbol{\theta}) - \langle \boldsymbol{\theta}, \boldsymbol{y} \rangle + \langle \boldsymbol{y}, \log \boldsymbol{y} \rangle. \end{aligned} \tag{2}$$

Adding the constant term $\langle \boldsymbol{y}, \log \boldsymbol{y} \rangle$ ensures that the loss is non-negative even when using soft labels (the constant term is zero with hard labels). The loss ℓ is convex in $\boldsymbol{\theta}$, and its gradient w.r.t. $\boldsymbol{\theta} \in \mathbb{R}^k$ is

添加常数项 $\langle \boldsymbol{y}, \log \boldsymbol{y} \rangle$ 可确保即使使用软标签，损失也是非负的（硬标签的常数项为零）。损失 ℓ 在 $\boldsymbol{\theta}$ 上是凸的，其梯度 w.r.t. $\boldsymbol{\theta} \in \mathbb{R}^k$ 为

$$\nabla_{\boldsymbol{\theta}} \ell(\boldsymbol{\theta}, \boldsymbol{y}) = \mathrm{softargmax}(\boldsymbol{\theta}) - \boldsymbol{y}. \tag{3}$$

2.2. Binary logistic loss

In the special case $k = 2$, letting $\boldsymbol{\theta} \coloneqq (0, \theta)$ and $\boldsymbol{y} \coloneqq (1-y, y)$ for $\theta \in \mathbb{R}$ and $y \in [0, 1]$, we obtain the softplus

在特殊情况下 $k = 2$，设 $\boldsymbol{\theta} \coloneqq (0, \theta)$ 和 $\boldsymbol{y} \coloneqq (1 - y, y)$，其中 $\theta \in \mathbb{R}$ 和 $y \in [0, 1]$，我们得到 softplus

$$\mathrm{softplus}(\theta) \coloneqq \log(1 + \exp(\theta)) = \mathrm{softmax}(\boldsymbol{\theta})$$

and the sigmoid

和 sigmoid

$$\mathrm{sigmoid}(\theta) \coloneqq \frac{1}{1 + \exp(-\theta)} = [\mathrm{softargmax}(\boldsymbol{\theta})]_1.$$

The two operators are again related by

这两个运算符再次关联起来

$$\mathrm{softplus}' = \mathrm{sigmoid}.$$

The binary logistic loss between the logit $\theta \in \mathbb{R}$ and the ground-truth label

logit $\theta \in \mathbb{R}$ 与真实标签之间的二元逻辑损失 $y \in [0, 1]$ is then

$$\ell(\theta, y) \coloneqq \mathrm{softplus}(\theta) + (1 - y) \log(1 - y) + y \log y - \theta y. \tag{4}$$

Beyond binary classification, a binary logistic loss can be used for pairwise ranking if we define $\theta \coloneqq \theta_i - \theta_j$, where θ_i and θ_j are the scores of items $i \in [k]$ and $j \in [k]$, respectively. The probability that item $i \in [k]$ is ranked higher than item $j \in [k]$ according to the model is then
除了二元分类之外，如果我们定义 $\theta \coloneqq \theta_i - \theta_j$，其中 θ_i 和 θ_j 分别是项目 $i \in [k]$ 和 $j \in [k]$ 的分数，则二元逻辑损失可用于成对排名。根据模型，项目 $i \in [k]$ 排名高于项目 $j \in [k]$ 的概率为

$$\operatorname{sigmoid}(\theta_i - \theta_j) = \frac{\exp(\theta_i)}{\exp(\theta_i) + \exp(\theta_j)}.$$

This is called the Bradley-Terry model (1952).
这被称为 Bradley-Terry 模型（1952）。

2.3. Fenchel–Young losses

It is well-known that the softmax and softargmax can be written from a variational perspective as
众所周知，softmax 和 softargmax 可以从变分的角度写成

$$\operatorname{softmax}(\boldsymbol{\theta}) = \max_{\boldsymbol{p} \in \triangle^k} \langle \boldsymbol{p}, \boldsymbol{\theta} \rangle - \langle \boldsymbol{p}, \log \boldsymbol{p} \rangle$$
$$\operatorname{softargmax}(\boldsymbol{\theta}) = \operatorname*{argmax}_{\boldsymbol{p} \in \triangle^k} \langle \boldsymbol{p}, \boldsymbol{\theta} \rangle - \langle \boldsymbol{p}, \log \boldsymbol{p} \rangle.$$

This suggests that we can create more general softmax and softargmax operators if we replace Shannon's negative entropy $\langle \boldsymbol{p}, \log \boldsymbol{p} \rangle$ by more general regularization $\Omega \colon \triangle^k \to \mathbb{R}$, namely,
这表明，如果我们用更通用的正则化 $\Omega \colon \triangle^k \to \mathbb{R}$ 代替香农的负熵 $\langle \boldsymbol{p}, \log \boldsymbol{p} \rangle$，我们就可以创建更通用的 softmax 和 softargmax 运算符，即

$$\operatorname{softmax}_\Omega(\boldsymbol{\theta}) \coloneqq \max_{\boldsymbol{p} \in \triangle^k} \langle \boldsymbol{p}, \boldsymbol{\theta} \rangle - \Omega(\boldsymbol{p})$$
$$\operatorname{softargmax}_\Omega(\boldsymbol{\theta}) \coloneqq \operatorname*{argmax}_{\boldsymbol{p} \in \triangle^k} \langle \boldsymbol{p}, \boldsymbol{\theta} \rangle - \Omega(\boldsymbol{p}).$$

By analogy with (2), we define the Fenchel–Young loss (Blondel et al., 2020) generated by Ω as
与 (2) 类似，我们将 Ω 生成的 Fenchel–Young 损失 (Blondel et al., 2020) 定义为

$$\ell_\Omega(\boldsymbol{\theta}, \boldsymbol{y}) \coloneqq \operatorname{softmax}_\Omega(\boldsymbol{\theta}) - \langle \boldsymbol{\theta}, \boldsymbol{y} \rangle + \Omega(\boldsymbol{y}) \tag{5}$$
$$\leq B_\Omega(\boldsymbol{y}, \operatorname{softargmax}_\Omega(\boldsymbol{\theta})), \tag{6}$$

where B_Ω is the Bregman divergence generated by Ω. The loss in (5) is always convex w.r.t. $\boldsymbol{\theta}$, unlike the loss in (6). Among many other desirable properties, if Ω is strictly convex, Fenchel–Young losses satisfy
其中 B_Ω 是由 Ω 生成的 Bregman 散度。(5) 中的损失始终是凸的 w.r.t. $\boldsymbol{\theta}$，与 (6) 中的损失不同。除了许多其他理想的特性外，如果 Ω 是严格凸的，Fenchel–Young 损失满足

$$\ell_\Omega(\boldsymbol{\theta}, \boldsymbol{y}) = 0 \iff \operatorname{softargmax}_\Omega(\boldsymbol{\theta}) = \boldsymbol{y} \tag{7}$$

and the gradient w.r.t. $\boldsymbol{\theta} \in \mathbb{R}^k$ is
并且梯度 w.r.t. $\boldsymbol{\theta} \in \mathbb{R}^k$ 是

$$\nabla_{\boldsymbol{\theta}} \ell_\Omega(\boldsymbol{\theta}, \boldsymbol{y}) = \operatorname{softargmax}_\Omega(\boldsymbol{\theta}) - \boldsymbol{y}.$$

In the binary classification setting, we can similarly replace sigmoid with $\operatorname{sigmoid}_\Omega$ in (4), as well as the regularization term $(1-y)\log(1-y) + y \log y$ with $\Omega((1-y, y))$.
在二分类设置中，我们可以类似地将 (4) 中的 sigmoid 替换为 $\operatorname{sigmoid}_\Omega$，并将正则化项 $(1-y)\log(1-y) + y \log y$ 替换为 $\Omega((1-y, y))$。

3. Generating losses from f-divergences

In this paper, we propose to study Fenchel–Young losses and associated operators when the regularizer is defined as
在本文中，我们提出研究 Fenchel-Young 损失及其相关算子，其中正则化子定义为

$$\Omega_f(\boldsymbol{p}; \boldsymbol{q}) \coloneqq D_f(\boldsymbol{p}, \boldsymbol{q}), \tag{8}$$

where D_f is a f-divergence and $\boldsymbol{q} \in \mathbb{R}_+^k$ is a reference measure, which contains the prior class weights. When $\boldsymbol{q}$ is not available, we can simply use $\boldsymbol{q} = \mathbf{1}$, in which case we will obtain negative f-entropies, as explained below.
其中 D_f 是 f 散度，$\boldsymbol{q} \in \mathbb{R}_+^k$ 是参考度量，其中包含先前的类权重。当 $\boldsymbol{q}$ 不可用时，我们可以简单地使用 $\boldsymbol{q} = \mathbf{1}$，在这种情况下我们将获得负的 f 熵，如下所述。

3.1. f-divergences

Let $f \colon \mathbb{R}_+ \to \mathbb{R}$ be a convex function such that $f(1) = 0$ and $f(0) = \lim_{u \to 0^+} f(u)$. The f-divergence between two discrete positive measures $\boldsymbol{p} \in \mathbb{R}_+^k$ and $\boldsymbol{q} \in \mathbb{R}_+^k$ (Rényi, 1961; Csiszár, 1967; Ali & Silvey, 1966) is then
假设 $f \colon \mathbb{R}_+ \to \mathbb{R}$ 为凸函数，使得 $f(1) = 0$ 且 $f(0) = \lim_{u \to 0^+} f(u)$。那么两个离散正测度 $\boldsymbol{p} \in \mathbb{R}_+^k$ 和 $\boldsymbol{q} \in \mathbb{R}_+^k$ 之间的 f-divergence (Rényi, 1961; Csiszár, 1967; Ali & Silvey, 1966) 就是

$$D_f(\boldsymbol{p}, \boldsymbol{q}) \coloneqq \sum_{j=1}^k f(p_j / q_j) q_j = \langle f(\boldsymbol{p}/\boldsymbol{q}), \boldsymbol{q} \rangle,$$

Table 1. By using f-divergences as regularization, we can generalize existing loss functions (logistic, sparsemax, entmax) to non-uniform reference measure $\boldsymbol{q}$ and we can construct several new loss functions. Reverse f-divergences (not listed below), such as the reverse KL, can also be used to generate alternative loss functions. The f-softargmax operators associated with the chi-square divergence and the α-divergence for $\alpha > 1$ can produce probability distributions with sparse support. When using the unit positive measure $\boldsymbol{q} = \mathbf{1}$, we obtain f-entropies, which recover existing known entropies (Shannon, Gini, Tsallis) up to a constant.
通过使用 f-散度作为正则化，我们可以将现有的损失函数（logistic、sparsemax、entmax）推广到非均匀参考测度 $\boldsymbol{q}$ 并且我们可以构造几个新的损失函数。反向 f-散度（未在下面列出），例如反向 KL，也可用于生成替代损失函数。与卡方散度和 α-散度（$\alpha > 1$）相关的 f-softargmax 运算符可以产生具有稀疏支持的概率分布。当使用单位正测度 $\boldsymbol{q} = \mathbf{1}$ 时，我们获得 f-熵，它将现有的已知熵（Shannon、Gini、Tsallis）恢复到一个常数。

Divergence	Entropy	Loss	Sparse
Kullback–Leibler	Shannon	Logistic	No
Chi-square	Gini	Sparsemax	Yes
α-divergence	Tsallis	Entmax	$\alpha > 1$
Jensen–Shannon	New	New	No
Squared Hellinger	New	New	No

where f and division are applied element-wise. An f-divergence is always non-negative and jointly convex in $\boldsymbol{p}$ and $\boldsymbol{q}$. Many existing divergences can be written in f-divergence form: see Table 1 and Appendix B.
其中 f 和除法是按元素应用的。f 散度始终为非负，且在 $\boldsymbol{p}$ 和 $\boldsymbol{q}$ 上联合凸。许多现有散度都可以写成 f 散度形式：参见表 1 和附录 B。

Divergence reversal. Since an f-divergence is not necessarily symmetric in $\boldsymbol{p}$ and $\boldsymbol{q}$, it may seem that setting the reference measure $\boldsymbol{q}$ as right argument in (8) is arbitrary. However, if we define $g(u) \coloneqq uf(1/u)$, we obtain the reverse divergence of D_f, $D_g(\boldsymbol{p}, \boldsymbol{q}) = D_f(\boldsymbol{q}, \boldsymbol{p})$. Therefore, for any f-divergence, we can always construct the corresponding reverse divergence, without loss of generality.
由于 f-散度不一定在 $\boldsymbol{p}$ 和 $\boldsymbol{q}$ 上对称，因此，将参考测度 $\boldsymbol{q}$ 设置为 (8) 中的右参数似乎是任意的。但是，如果我们定义 $g(u) \coloneqq uf(1/u)$，我们得到 D_f 的反向散度，$D_g(\boldsymbol{p}, \boldsymbol{q}) = D_f(\boldsymbol{q}, \boldsymbol{p})$。因此，对于任何 f-散度，我们总是可以构造相应的反向散度，而不会失去一般性。

3.2. f-entropies

As we emphasized, our proposed framework can take into account a discrete reference measure $\boldsymbol{q} \in \mathbb{R}_+^k$, which intuitively contains the prior class weights. When such a measure is not available, we can simply choose the unit positive measure $\boldsymbol{q} = \mathbf{1}$. In this case, we recover negative f-entropies (Cichocki & Amari, 2010), a.k.a. f-negentropies. They are defined as
正如我们强调的那样，我们提出的框架可以考虑离散参考测度 $\boldsymbol{q} \in \mathbb{R}_+^k$，它直观地包含先前的类权重。当没有这样的测度时，我们可以简单地选择单位正测度 $\boldsymbol{q} = \mathbf{1}$。在这种情况下，我们恢复负 f-熵 (Cichocki & Amari, 2010)，a.k.a. f-negentropies。它们定义为

$$\Omega_f(\boldsymbol{p}) \coloneqq \sum_{j=1}^{k} f(p_j) = \langle f(\boldsymbol{p}), \mathbf{1} \rangle = D_f(\boldsymbol{p}, \mathbf{1}).$$

By choosing f, we recover (up to a constant) numerous existing negentropies. When $f(u) = u \log u$, which is the generating function of the KL divergence, we recover the Shannon negentropy,
通过选择 f，我们可以恢复（最多一个常数）大量现有的负熵。当 $f(u) = u \log u$（即 KL 散度的生成函数）时，我们恢复 Shannon **负熵**，

$$\Omega_f(\boldsymbol{p}) = \langle \boldsymbol{p}, \log \boldsymbol{p} \rangle.$$

When $f(u) = \frac{1}{2}(u^2 - 1)$, which is the generating function of the chi-square divergence, we recover the Gini negentropy, used for the sparsemax loss (Martins & Astudillo, 2016),
当 $f(u) = \frac{1}{2}(u^2 - 1)$（这是卡方散度的生成函数）时，我们恢复了 Gini **负熵**，用于 sparsemax 损失 (Martins & Astudillo, 2016)，

$$\Omega_f(\boldsymbol{p}) \doteq \frac{1}{2}(\|\boldsymbol{p}\|_2^2 - 1)$$

(we use $\doteq$ for equality up to a constant). More generally, with $f(u) = \frac{(u^\alpha - 1) - \alpha(u-1)}{\alpha(\alpha-1)}$, which is the generating function of α-divergences, we recover the Tsallis negentropy,
（我们使用 $\doteq$ 表示相等到常数）。更一般地，使用 $f(u) = \frac{(u^\alpha - 1) - \alpha(u-1)}{\alpha(\alpha-1)}$，它是 α-发散的生成函数，我们恢复 Tsallis **负熵**，

$$\Omega_f(\boldsymbol{p}) \doteq \frac{1}{\alpha}\langle \boldsymbol{p}, \log_\alpha(\boldsymbol{p}) \rangle = \frac{1}{\alpha(\alpha-1)}(\|\boldsymbol{p}\|_\alpha^\alpha - 1),$$

used by the entmax loss (Peters et al., 2019). The Tsallis negentropy itself is very general, as it recovers the Shannon negentropy when $\alpha \to 1$ and the Gini negentropy when $\alpha = 2$. In our experiments, we will demonstrate good results with the choice $\alpha = 1.5$, which can be thought as a middle ground between Shannon entropy (used by the logistic loss) and Gini entropy (used by the sparsemax loss).
entmax 损失 (Peters et al., 2019) 所使用的。Tsallis 负熵本身非常通用，因为它在 $\alpha \to 1$ 时恢复 Shannon 负熵，在 $\alpha = 2$ 时恢复 Gini 负熵。在我们的实验中，我们将展示选择 $\alpha = 1.5$ 的良好结果，这可以被认为是 Shannon 熵（由 logistic 损失使用）和 Gini 熵（由 sparsemax 损失使用）之间的中间值。

New entropy instances. Beyond existing entropies, some choices of f lead to entropies that, to our knowledge, had not been considered before. For example, in Figure 1, we show the f-entropies associated with the Jensen-Shannon and squared Hellinger divergences.
除了现有的熵之外，一些 f 的选择会导致熵，据我们所知，这些熵以前从未被考虑过。例如，在图 1 中，我们展示了与 Jensen-Shannon 和平方 Hellinger 散度相关的 f 熵。

Effective domain. We point, however, that some f-entropies are only well-defined on the relative interior of the probability simplex, if $\lim_{u\to 0} f(u) = -\infty$. This is for instance the case of the f-entropies associated with the reverse KL and Jeffrey divergences. This means that the loss functions generated by these choices only work with strictly positive soft labels.
然而，我们指出，如果 $\lim_{u\to 0} f(u) = -\infty$，则某些 f-熵仅在概率单纯形的相对内部定义明确。例如，与逆 KL 和 Jeffrey 散度相关的 f-熵就是这种情况。这意味着这些选择生成的损失函数仅适用于严格正的软标签。

3.3. f-softmax and f-softargmax

Overloading the notation, we define the f-softmax as
重载符号，我们将 f-softmax 定义为

$$\operatorname{softmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) \coloneqq \max_{\boldsymbol{p}\in\triangle^k} \langle \boldsymbol{p}, \boldsymbol{\theta}\rangle - D_f(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{R} \quad (9)$$

and the f-softargmax as
并且 f-softargmax 为

$$\operatorname{softargmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) \coloneqq \operatorname*{argmax}_{\boldsymbol{p}\in\triangle^k} \langle \boldsymbol{p}, \boldsymbol{\theta}\rangle - D_f(\boldsymbol{p}, \boldsymbol{q}) \in \triangle^k. \quad (10)$$

Compared to a classical softmax and softargmax, our operators use a function f and include an additional reference measure $\boldsymbol{q}$ as argument. When $f(u) = u \log u$ and $\boldsymbol{q} = \mathbf{1}$, we recover the classical softmax and softargmax.

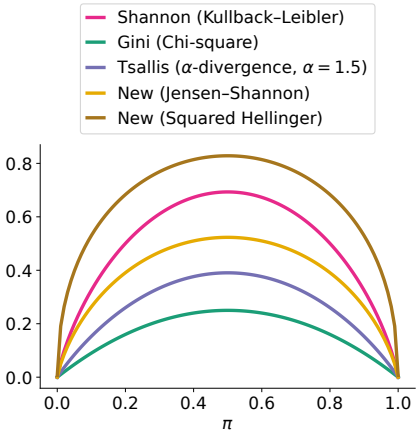

Figure 1. Illustration of f-entropies $-\Omega_f(\boldsymbol{p})$ for $\boldsymbol{p} = (1-\pi, \pi)$ and varying $\pi \in [0, 1]$. We add a constant $f(0)$ to ensure non-negativity of the f-entropies.
对于 $\boldsymbol{p} = (1-\pi, \pi)$ 和变化的 $\pi \in [0, 1]$，f-熵 $-\Omega_f(\boldsymbol{p})$ 的图示。我们添加一个常数 $f(0)$ 来确保 f-熵的非负性。

与经典的 softmax 和 softargmax 相比，我们的运算符使用函数 f 并包含一个额外的参考度量 $\boldsymbol{q}$ 作为参数。当 $f(u) = u \log u$ 和 $\boldsymbol{q} = \mathbf{1}$ 时，我们恢复经典的 softmax 和 softargmax。

Scaling the divergence by a temperature parameter $\beta > 0$ can easily be done. Indeed, for any $\beta > 0$,
可以轻松通过温度参数 $\beta > 0$ 缩放散度。实际上，对于任何 $\beta > 0$，

$$\operatorname{softmax}_{\beta f}(\boldsymbol{\theta}; \boldsymbol{q}) = \beta \operatorname{softmax}_f(\boldsymbol{\theta}/\beta; \boldsymbol{q})$$
$$\operatorname{softargmax}_{\beta f}(\boldsymbol{\theta}; \boldsymbol{q}) = \operatorname{softargmax}_f(\boldsymbol{\theta}/\beta; \boldsymbol{q}).$$

Sparse distributions. As summarized in Table 1, the f-softargmax associated with the chi-square and α divergences for $\alpha > 1$ can produce probability distributions with sparse support, meaning that some classes have exactly zero probability according to the model. As will be clear from Proposition 1, the f-softargmax can be sparse when $0 \in \operatorname{dom}(f') \iff \lim_{u\to 0} f'(u) > -\infty$.
如表 1 中所述，与 $\alpha > 1$ 的卡方和 α 散度相关的 f-softargmax 可以产生具有稀疏支持的概率分布，这意味着根据模型，某些类别的概率恰好为零。从命题 1 可以清楚地看出，当 $0 \in \operatorname{dom}(f') \iff \lim_{u\to 0} f'(u) > -\infty$ 时，f-softargmax 可以是稀疏的。

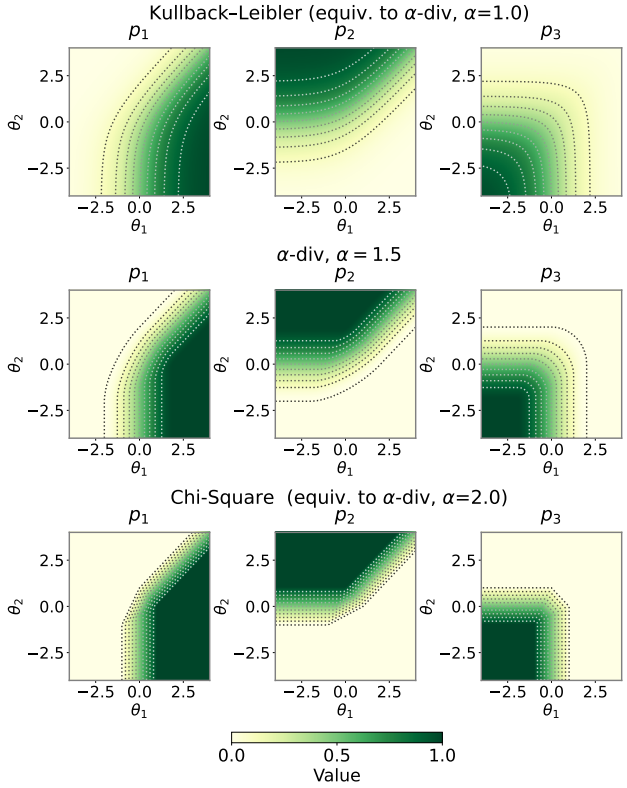

Figure 2. Illustration of $(p_1, p_2, p_3) = \operatorname{softargmax}_f(\theta_1, \theta_2, 0; \boldsymbol{q})$ when varying $\theta_1, \theta_2 \in \mathbb{R}$ for three possible divergences and with $\boldsymbol{q} = (1, 1, 1)$. More illustrations are given in Appendix B.
当 $\theta_1, \theta_2 \in \mathbb{R}$ 变化时，$(p_1, p_2, p_3) = \operatorname{softargmax}_f(\theta_1, \theta_2, 0; \boldsymbol{q})$ 的图示，其中 $\theta_1, \theta_2 \in \mathbb{R}$ 表示三种可能的发散，并且 $\boldsymbol{q} = (1, 1, 1)$。附录 B 中给出了更多图示。

3.4. f-softplus and f-sigmoid

As special cases of f-softmax and f-softargmax, by defining $\boldsymbol{\theta} \coloneqq (0, \theta)$, $\boldsymbol{p} \coloneqq (1 - \pi, \pi)$ and $\boldsymbol{q} \coloneqq (q_0, q_1)$, we obtain the f-softplus
作为 f-softmax 和 f-softargmax 的特殊情况，通过定义 $\boldsymbol{\theta} \coloneqq (0, \theta)$、$\boldsymbol{p} \coloneqq (1 - \pi, \pi)$ 和 $\boldsymbol{q} \coloneqq (q_0, q_1)$，我们得到 f-softplus

$$\operatorname{softplus}_f(\theta; \boldsymbol{q}) \coloneqq \max_{\pi \in [0,1]} \pi\theta - D_f((1 - \pi, \pi), \boldsymbol{q}) \in \mathbb{R}$$

and the f-sigmoid

$$\operatorname{sigmoid}_f(\theta; \boldsymbol{q}) \coloneqq \operatorname*{argmax}_{\pi \in [0,1]} \pi\theta - D_f((1-\pi, \pi), \boldsymbol{q}) \in [0, 1].$$

The effect of the prior class weights $\boldsymbol{q} \in \mathbb{R}^2_+$ on the shape of the sigmoid is illustrated in Figure 8 in Appendix B. The f-softplus and f-sigmoid can be used, not only for binary classification, but also for pairwise ranking, by analogy with Section 2.2. Similarly to the f-softmax and f-softargmax, we can easily scale D_f by a temperature parameter $\beta > 0$.
附录 B 中的图 8 说明了先验类权重 $\boldsymbol{q} \in \mathbb{R}^2_+$ 对 S 形的影响。

f-softplus 和 f-sigmoid 不仅可用于二元分类，还可用于成对排序，与第 2.2 节类似。

与 f-softmax 和 f-softargmax 类似，我们可以轻松地通过温度参数 $\beta > 0$ 缩放 D_f。

3.5. Loss function

To obtain a loss function associated with the f-softargmax, we instantiate the Fenchel–Young loss defined in (5) with the regularization defined in (8) to define
为了获得与 f-softargmax 相关的损失函数，我们实例化 (5) 中定义的 Fenchel-Young 损失，并使用 (8) 中定义的正则化来定义

$$\begin{aligned} \ell_f(\boldsymbol{\theta}, \boldsymbol{y}; \boldsymbol{q}) &\coloneqq \operatorname{softmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) + \Omega_f(\boldsymbol{y}; \boldsymbol{q}) - \langle \boldsymbol{\theta}, \boldsymbol{y} \rangle \\ &= \operatorname{softmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) + D_f(\boldsymbol{y}, \boldsymbol{q}) - \langle \boldsymbol{\theta}, \boldsymbol{y} \rangle, \end{aligned} \tag{11}$$

where $\boldsymbol{\theta} \in \mathbb{R}^k$ are the logits, $\boldsymbol{y} \in \triangle^k$ is the ground-truth and $\boldsymbol{q} \in \mathbb{R}^k_+$ is a reference measure, which contains the prior class weights. This loss inherits from all the desirable properties of Fenchel–Young losses. In particular, it is convex w.r.t. $\boldsymbol{\theta}$ and it is differentiable everywhere if f is strictly convex. On the other hand, it is not necessarily convex w.r.t. $\boldsymbol{q}$, since $\operatorname{softmax}_f$ involves the maximum over a collection of concave functions of $\boldsymbol{q}$. Because $\boldsymbol{y} \in \triangle^k$ and not just $\boldsymbol{y} \in \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_k\}$, this loss can also be used for learning from label proportions, as is useful in distillation for example.
其中 $\boldsymbol{\theta} \in \mathbb{R}^k$ 是 logits，$\boldsymbol{y} \in \triangle^k$ 是基本事实，$\boldsymbol{q} \in \mathbb{R}^k_+$ 是参考度量，其中包含先前的类权重。此损失继承了 Fenchel–Young 损失的所有理想属性。具体而言，它是凸的 w.r.t. $\boldsymbol{\theta}$，并且如果 f 是严格凸的，则它在任何地方都是可微的。另一方面，它不一定是凸的 w.r.t. $\boldsymbol{q}$，因为 $\operatorname{softmax}_f$ 涉及 $\boldsymbol{q}$ 的 凹函数集合的最大值。因为 $\boldsymbol{y} \in \triangle^k$ 而不仅仅是 $\boldsymbol{y} \in \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_k\}$，所以此损失还可用于从标签比例中学习，例如在蒸馏中很有用。

Choosing $\boldsymbol{q}$. One distinctive feature of our losses compared to usual instances of Fenchel–Young losses is the possibility to adjust the reference measure $\boldsymbol{q}$. In the absence of prior knowledge, we can simply use $\boldsymbol{q} = \mathbf{1}$ or $\boldsymbol{q} = \mathbf{1}/k$. This recovers Fenchel–Young losses generated by f-negentropies. If class weights are available, we can use this prior knowledge as $\boldsymbol{q}$.
与常见的 Fenchel-Young 损失相比，我们的损失的一个显著特点是可以调整参考度量 $\boldsymbol{q}$。在没有先验知识的情况下，我们可以简单地使用 $\boldsymbol{q} = \mathbf{1}$ 或 $\boldsymbol{q} = \mathbf{1}/k$。这恢复了由 f-negentropies 产生的 Fenchel-Young 损失。如果有类权重，我们可以将此先验知识用作 $\boldsymbol{q}$。

3.6. Computation

On first sight, it is not obvious how to solve the variational problems (9) and (10) involved in computing the f-softmax and f-softargmax, respectively. We need an algorithm that works for any valid choice of f and parallelizes well, as we often need to compute the operators on a batch of b logits $\boldsymbol{\theta}_{i_1}, \ldots, \boldsymbol{\theta}_{i_b}$, where $\boldsymbol{\theta}_{i_j} \coloneqq h_{\boldsymbol{w}}(\boldsymbol{x}_{i_j})$.
乍一看，如何解决分别计算 f-softmax 和 f-softargmax 所涉及的变分问题 (9) 和 (10) 并不明显。我们需要一种适用于任何有效 f 选择且并行性良好的算法，因为我们经常需要计算一批 b 个 logits $\boldsymbol{\theta}_{i_1}, \ldots, \boldsymbol{\theta}_{i_b}$ 上的运算符，其中 $\boldsymbol{\theta}_{i_j} \coloneqq h_{\boldsymbol{w}}(\boldsymbol{x}_{i_j})$。

In this section, we introduce a novel generic algorithm for computing the f-softmax and f-softargmax given access to f and f^*, where f^* is the convex conjugate of f. The latter is usually available in closed form; we give numerous examples in Appendix B.
在本节中，我们介绍了一种新颖的通用算法，用于在给定 f 和 f^* 的情况下计算 f-softmax 和 f-softargmax，其中 f^* 是 f 的凸共轭。后者通常以封闭形式提供；我们在附录 B 中给出了大量示例。

For convenience, with a slight abuse of notation, we define the shorthand $f'_* \coloneqq (f^*)'$. We denote $f'(0) = \lim_{x \to 0, x \geq 0} f'(x)$ which may be finite or $-\infty$.
为了方便起见，我们稍微滥用符号，定义简写 $f'_* \coloneqq (f^*)'$。我们表示 $f'(0) = \lim_{x \to 0, x \geq 0} f'(x)$，它可以是有限的，也可以是 $-\infty$。

Proposition 1. Reduction to root finding

Let f be a strictly convex and differentiable function such that $(0, +\infty) \subseteq \operatorname{dom} f'$. Then, for any $\boldsymbol{\theta} \in \mathbb{R}^k$,
假设 f 是严格凸且可微的函数，并且 $(0, +\infty) \subseteq \operatorname{dom} f'$。然后，对于任何 $\boldsymbol{\theta} \in \mathbb{R}^k$，

$$\operatorname{softmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) = \tau^\star + \sum_{j=1}^{k} q_j f^*(\max\{\theta_j - \tau^\star, f'(0)\}) \tag{12}$$

$$[\operatorname{softargmax}_f(\boldsymbol{\theta}; \boldsymbol{q})]_j = q_j f'_*(\max\{\theta_j - \tau^\star, f'(0)\}), \tag{13}$$

where $\tau^\star$ is the unique solution of
其中 $\tau^\star$ 是

$$\sum_{j=1}^{k} q_j f'_*(\max\{\theta_j - \tau, f'(0)\}) = 1, \tag{14}$$

on $\tau \in [\tau_{\min}, \tau_{\max}]$, where, for $j^\star \in \operatorname{argmax}_{j \in [k]} \theta_j$:

$$\tau_{\min} \coloneqq \theta_{j^\star} - f'(1/q_{j^\star})$$
$$\tau_{\max} \coloneqq \theta_{j^\star} - f'\left(1/\left(\textstyle\sum_{j=1}^{k} q_j\right)\right).$$

A proof is given in Appendix C.4. Proposition 1 is a generalization of (Blondel et al., 2020, Proposition 9) to arbitrary reference measures $\boldsymbol{q}$. Our proof technique is different: it uses Fenchel duality as opposed to Lagrange duality and it rigorously accounts for the domain of f', which is one of the technical difficulties for supporting general f-divergences. Proposition 1 is a generalization of (Wang et al., 2024, Theorem 1) to the case $0 \in \operatorname{dom}(f')$, that is, to f-softargmax operators with sparse output. A Newton algorithm was proposed in (Terjék, 2021) for computing the same operator. However, that algorithm was used for regularized optimal transport, not for creating loss functions. In addition, it assumes that $(f + \delta_{\mathbb{R}_+})^*$ is twice differentiable, which is not the case when $0 \in \operatorname{dom}(f')$, and does not enjoy convergence guarantees, as a line search was not used. See also the discussion in (Belousov & Peters, 2017) on the implications of $0 \notin \operatorname{dom}(f')$.
附录 C.4 中给出了证明。命题 1 是 (Blondel et al., 2020, 命题 9) 对任意参考测度 $\boldsymbol{q}$ 的推广。我们的证明技术有所不同：它使用 Fenchel 对偶而不是 Lagrange 对偶，并且严格考虑了 f' 的定义域，这是支持一般 f-散度的技术难点之一。命题 1 是 (Wang et al., 2024, 定理 1) 对 $0 \in \operatorname{dom}(f')$ 情况的推广，即具有稀疏输出的 f-softargmax 运算符。在 (Terjék, 2021) 中提出了一种用于计算相同运算符的牛顿算法。但是，该算法用于正则化最优传输，而不是用于创建损失函数。此外，它假设 $(f + \delta_{\mathbb{R}_+})^*$ 是二阶可微的，而当 $0 \in \operatorname{dom}(f')$ 时情况并非如此，并且不享受收敛保证，因为没有使用线搜索。另请参阅 (Belousov & Peters, 2017) 中关于 $0 \notin \operatorname{dom}(f')$ 含义的讨论。

Implementation. In practice, we solve the root equation in (14) by bisection (Algorithm 1). This algorithm has several advantages: it is simple, parallelizes well on GPU and TPU (we often need to compute the f-softargmax of a batch) and achieves an error on iteration t of $(\tau_{\max} - \tau_{\min})/2^t$. That is, the error exponentially decreases with the number of iterations. We show in Appendix A.3 that the overhead of our proposed algorithm is negligible compared to a classical softargmax.
在实践中，我们通过二分法（算法 1）求解 (14) 中的根方程。该算法有几个优点：它很简单，在 GPU 和 TPU 上并行性很好（我们经常需要计算一批的 f-softargmax），并且在迭代 t 时实现 $(\tau_{\max} - \tau_{\min})/2^t$ 的误差。也就是说，误差随着迭代次数呈指数下降。我们在附录 A.3 中表明，与经典的 softargmax 相比，我们提出的算法的开销可以忽略不计。

Algorithm 1 Computing f-softmax and f-softargmax

Input: logits $\boldsymbol{\theta} \in \mathbb{R}^k$, prior $\boldsymbol{q} \in \mathbb{R}^k_+$, tolerance $\epsilon > 0$
$[\boldsymbol{p}(\tau)]_j \coloneqq q_j f'_*(\max\{\theta_j - \tau, f'(0)\})$, $j \in [k]$
$\phi(\tau) \coloneqq \langle \boldsymbol{p}(\tau), \mathbf{1} \rangle - 1$
$j^\star \in \operatorname{argmax}_{j \in [k]} \theta_j$
$\tau_{\min} \leftarrow \theta_{j^\star} - f'(1/q_{j^\star})$
$\tau_{\max} \leftarrow \theta_{j^\star} - f'\left(1/\left(\sum_{j=1}^k q_j\right)\right)$
$\tau \leftarrow (\tau_{\min} + \tau_{\max})/2$
while $|\phi(\tau)| > \epsilon$
  if $\phi(\tau) < 0$ $\tau_{\max} \leftarrow \tau$
  else $\tau_{\min} \leftarrow \tau$
  $\tau \leftarrow (\tau_{\min} + \tau_{\max})/2$
Output: $\operatorname{softargmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) \approx \boldsymbol{p}(\tau)$
$\operatorname{softmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) \approx \tau + \sum_{j=1}^k q_j f^*(\max\{\theta_j - \tau, f'(0)\})$

Closed forms in the binary case. In the special case $k = 2$ (binary classification), we can often derive closed-form solutions for the f-softplus and the f-sigmoid operator. For completeness, we derive the expressions for numerous cases in Appendix C.3.
在特殊情况下 $k = 2$（二元分类），我们通常可以推导出 f-softplus 和 f-sigmoid 算子的闭式解。为了完整起见，我们在附录 C.3 中推导出许多情况的表达式。

3.7. Differentiation

Differentiating through f-softmax and loss. In order to differentiate the f-softmax, we can simply use Danskin's theorem to obtain
为了区分 f-softmax，我们可以简单地使用 Danskin 定理来获得

$$\nabla_{\boldsymbol{\theta}} \operatorname{softmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) = \boldsymbol{p}^\star \coloneqq \operatorname{softargmax}_f(\boldsymbol{\theta}; \boldsymbol{q})$$
$$\nabla_{\boldsymbol{q}} \operatorname{softmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) = -\nabla_{\boldsymbol{q}} D_f(\boldsymbol{p}^\star, \boldsymbol{q}).$$

As a result, the loss gradients are

$$\nabla_{\boldsymbol{\theta}} \ell_f(\boldsymbol{\theta}, \boldsymbol{y}; \boldsymbol{q}) = \boldsymbol{p}^\star - \boldsymbol{y} = \operatorname{softargmax}_f(\boldsymbol{\theta}; \boldsymbol{q}) - \boldsymbol{y}$$
$$\nabla_{\boldsymbol{q}} \ell_f(\boldsymbol{\theta}, \boldsymbol{y}; \boldsymbol{q}) = \nabla_{\boldsymbol{q}} D_f(\boldsymbol{y}, \boldsymbol{q}) - \nabla_{\boldsymbol{q}} D_f(\boldsymbol{p}^\star, \boldsymbol{q}).$$

The first equation is in complete analogy with (3). The second equation is the difference of the f-divergence gradients evaluated at the ground-truth $\boldsymbol{y}$ and the prediction $\boldsymbol{p}^\star$. Differentiating through the f-softplus operator and its associated loss function is similar.
第一个方程与 (3) 完全类似。第二个方程是在真实值 $\boldsymbol{y}$ 和预测值 $\boldsymbol{p}^\star$ 处求得的 f-发散梯度之差。通过 f-softplus 算子及其相关损失函数进行区分是类似的。

Differentiating through f-softargmax. When the goal is to use the f-softargmax as the operator associated with our loss functions, we do not need to differentiate through the f-softargmax. As explained above, thanks to Danskin's theorem, differentiating through the f-softmax is sufficient. When the goal is to use the f-softargmax as an attention mechanism, however, we do need to differentiate through the f-softargmax. This is more challenging as, from Proposition 1, we need to differentiate through the solution of a root equation. Under assumptions on f, we can apply the implicit function theorem (Krantz & Parks, 2002) through the root equation's solution. This can be implemented using automatic implicit differentiation (Blondel et al., 2022). Importantly, implicit differentiation does not require solving a costly linear system here. Since the root equation is one-dimensional, a simple division is sufficient.
当目标是使用 f-softargmax 作为与我们的损失函数相关的运算符时，我们不需要通过 f-softargmax 进行区分。如上所述，由于 Danskin 定理，通过 f-softmax 进行区分就足够了。然而，当目标是使用 f-softargmax 作为注意力机制时，我们确实需要通过 f-softargmax 进行区分。这更具挑战性，因为根据命题 1，我们需要通过根方程的解进行区分。在对 f 的假设下，我们可以通过根方程的解应用隐函数定理 (Krantz & Parks, 2002)。这可以使用自动隐式微分 (Blondel et al., 2022) 来实现。重要的是，隐式微分不需要在这里求解昂贵的线性系统。由于根方程是一维的，因此简单的除法就足够了。

4. Experiments

To evaluate different f-divergence generated losses, we apply them to tasks of different data modalities, including image classification (Section 4.1) and text generation (Section 4.2). These experiments also cover different training strategies: from scratch, finetuning, and distillation.
为了评估不同的 f 散度产生的损失，我们将它们应用于不同数据模态的任务，包括图像分类 (Section 4.1) 和文本生成 (Section 4.2)。这些实验还涵盖了不同的训练策略：从头开始、微调和提炼。

4.1. ImageNet classification

We apply different f-divergence generated losses to train a ResNet50 model (He et al., 2016) on the ImageNet-2012 dataset (Russakovsky et al., 2015). The ImageNet dataset contains 1.28 million training images and 50,000 validation images, belonging to one of 1,000 classes. The ResNet50 model is a standard choice for ImageNet.
我们应用不同的 f 散度生成损失在 ImageNet-2012 数据集 (Russakovsky et al., 2015) 上训练 ResNet50 模型 (He et al., 2016)。ImageNet 数据集包含 128 万张训练图像和 50,000 张验证图像，属于 1,000 个类别之一。ResNet50 模型是 ImageNet 的标准选择。

We use an SGD optimizer with 0.9 momentum to train the ResNet50 model for 90 epochs. During the initial 5 epochs, we use a linear warmup that achieves a peak learning rate of 0.2; we then use cosine annealing to reduce the learning rate to 0. The weight decay is set to be 10^{-4}. The batch size is 512.
我们使用动量为 0.9 的 SGD 优化器对 ResNet50 模型进行 90 个 epoch 的训练。在最初的 5 个 epoch 中，我们使用线性预热，达到峰值学习率 0.2；然后我们使用余弦退火将学习率降低到 0。权重衰减设置为 10^{-4}。批量大小为 512。

Table 2 shows the validation accuracy of training ResNet50 using different f-divergence generated losses; boldface indicates the highest accuracy. Recall that the KL generated loss is equivalent to the standard cross-entropy loss; therefore, using the KL generated loss should match the result of prior work (He et al., 2016). Our experiments confirm this: the KL loss reaches 76.87% accuracy, consistent with previous benchmarks (Appendix A.1). Perhaps surprisingly, we find that the α-divergence loss function surpasses the KL loss in validation accuracy. This improvement is achieved without adjusting any hyperparameters and therefore is notable. In addition, $\alpha = 1.5$ seems to be optimal within α-divergences as shown in Figure 5, where we see that validation accuracy is maximal for $\alpha = 1.5$ among 11 values for $\alpha \in [1, 2]$. However, the Jensen–Shannon, Squared Hellinger, and Chi-square (Pearson divergence) generated loss functions perform worse than the KL divergence in this experiment.
表 2 显示了使用不同的 f 散度生成损失训练 ResNet50 的验证准确率；粗体表示最高精度。回想一下，KL 生成的损失相当于标准交叉熵损失；因此，使用 KL 生成的损失应该与先前工作 (He et al., 2016) 的结果相匹配。我们的实验证实了这一点：KL 损失达到 76.87% 的准确率，与之前的基准一致（附录 A.1）。也许令人惊讶的是，我们发现 α 散度损失函数在验证准确率方面超过了 KL 损失。这种改进是在不调整任何超参数的情况下实现的，因此是值得注意的。此外，如图 5 所示，在 α 散度内，$\alpha = 1.5$ 似乎是最优的，其中我们看到，在 $\alpha \in [1, 2]$ 的 11 个值中，当 $\alpha = 1.5$ 时，验证准确率最高。然而，在本实验中，Jensen–Shannon、Squared Hellinger 和 Chi-square（皮尔逊散度）生成的损失函数表现不如 KL 散度。

4.2. Language modelling

We compare the performance of different f-divergence-based loss functions for training language models (LMs). To provide a thorough evaluation, we consider three common LM training strategies: (i) pretraining, (ii) supervised fine-tuning (SFT), and (iii) distillation-based fine-tuning. All three approaches optimize a next-token prediction loss:

$$\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\sum_{t=1}^{|\boldsymbol{y}|}\ell_f(h_{\boldsymbol{w}}(\boldsymbol{x},\boldsymbol{y}_{<t}),\phi(y_t);\boldsymbol{q})\right], \qquad (15)$$

where $h_{\boldsymbol{w}}$ is the LM that outputs logits with parameters $\boldsymbol{w}$, $|\boldsymbol{y}|$ is the length of the text sequence $\boldsymbol{y}$, and the reference measure $\boldsymbol{q} = \mathbf{1}$ is a unitary prior; cf. Equation (11). Pretraining and SFT use one-hot vectors $\phi(y_t) \in \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_k\}$ as the targets; distillation uses soft labels $\phi(y_t) \in \triangle^k$ obtained as the next-token probability of a teacher model.
我们比较了不同的基于 f 散度的损失函数在训练语言模型 (LM) 方面的表现。为了进行全面评估，我们考虑了三种常见的 LM 训练策略：(i) 预训练，(ii) 监督微调 (SFT)，以及 (iii) 基于蒸馏的微调。这三种方法都优化了下一个 token 预测损失：

$$\mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\sum_{t=1}^{|\boldsymbol{y}|}\ell_f(h_{\boldsymbol{w}}(\boldsymbol{x},\boldsymbol{y}_{<t}),\phi(y_t);\boldsymbol{q})\right], \qquad (16)$$

其中 $h_{\boldsymbol{w}}$ 是输出参数为 $\boldsymbol{w}$ 的 logits 的 LM，$|\boldsymbol{y}|$ 是文本序列 $\boldsymbol{y}$ 的长度，参考度量 $\boldsymbol{q} = \mathbf{1}$ 是单一先验；参见 Equation (11)。预训练和 SFT 使用独热向量 $\phi(y_t) \in \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_k\}$ 作为目标；蒸馏使用软标签 $\phi(y_t) \in \triangle^k$ 作为教师模型的下一个 token 概率。

Table 2. ImageNet classification results.

Divergence	Accuracy (%)
Kullback–Leibler	76.87
Chi-square	76.06
α-divergence ($\alpha = 1.5$)	77.56
Jensen–Shannon	72.24
Squared Hellinger	72.81

Pretraining with f-divergence generated losses. We used the same pretraining method as the NanoDO models (Liu et al., 2024; Wortsman et al., 2024), a set of well-tuned decoder-only transformer models trained on the public C4 dataset (Raffel et al., 2020). Specifically, we follow the training set up of the 1.2B parameter model from Wortsman et al. (2024); we use 250B tokens of C4 to train the model, so that the model is Chinchilla optimal (Hoffmann et al., 2022). Our pretraining results (see Appendix A.2) echo our main findings from the ImageNet experiments in Section 4.1. Specifically, losses based on α-divergence, with $\alpha = 1.5$, performed best for predicting the next word, slightly outperforming the standard KL divergence loss.
我们使用了与 NanoDO 模型 (Liu et al., 2024; Wortsman et al., 2024) 相同的预训练方法，这是一组在公共 C4 数据集 (Raffel et al., 2020) 上训练的经过良好调整的仅解码器转换器模型。具体来说，我们遵循 Wortsman et al. (2024) 中 1.2B 参数模型的训练设置；我们使用 C4 的 250B tokens 来训练模型，以便该模型达到 Chinchilla 最优 (Hoffmann et al., 2022)。我们的预训练结果（见附录 A.2）与我们在第 4.1 节中的 ImageNet 实验的主要发现相呼应。具体来说，基于 α-divergence 的损失，其中 $\alpha = 1.5$，在预测下一个单词时表现最佳，略优于标准 KL 散度损失。

Learning $\boldsymbol{q}$. Since our loss functions take a reference distribution $\boldsymbol{q}$ and we can compute gradients with respect to $\boldsymbol{q}$, a natural idea is to try to learn it from data. As a negative result, using the NanoDO implementation above on the C4 dataset, we found that learning $\boldsymbol{q}$ does not seem to help improve performance compared to $\boldsymbol{q} = \mathbf{1}$, both when using the KL divergence and when using the $\alpha = 1.5$ divergence.
由于我们的损失函数采用参考分布 $\boldsymbol{q}$，并且我们可以计算关于 $\boldsymbol{q}$ 的梯度，因此一个自然的想法是尝试从数据中学习它。结果显示，使用上述 NanoDO 实现在 C4 数据集上进行测试时，我们发现学习 $\boldsymbol{q}$ 似乎无助于提高性能，与 $\boldsymbol{q} = \mathbf{1}$ 相比，无论是使用 KL 散度还是使用 $\alpha = 1.5$ 散度。

SFT and distillation with f-divergence generated losses. Instead of pretraining LMs from scratch, practitioners often finetune readily available, open-weight models pretrained with standard cross-entropy. This prompts a question: Can we effectively finetune these cross-entropy pretrained models using different f-divergence generated losses? To explore this, we evaluated f-divergence generated losses with two common finetuning methods, SFT and distillation, on a text summarization task (Narayan et al., 2018). For SFT, we use a pretrained T5-base model (Raffel et al., 2020) with 250M parameters and train it on the XSum dataset (Narayan et al., 2018) with a next-token prediction loss (16). Traditionally, the cross-entropy loss is used as the next-token prediction loss, which is equivalent to KL-divergence. In our experiments, however, we evaluate more general f-divergence generated losses.
从业者通常不会从头开始对 LM 进行预训练，而是对现成的、使用标准交叉熵进行预训练的开放权重模型进行微调。这引发了一个问题：我们能否使用不同的 f 散度生成的损失有效地微调这些交叉熵预训练模型？为了探索这个问题，我们在文本摘要任务 (Narayan et al., 2018) 上使用两种常见的微调方法 SFT 和蒸馏评估了 f 散度生成的损失。对于 SFT，我们使用具有 2.5 亿个参数的预训练 T5 基础模型 (Raffel et al., 2020)，并在 XSum 数据集 (Narayan et al., 2018) 上对其进行训练，下一个 token 预测损失为 (16)。传统上，交叉熵损失用作下一个 token 预测损失，相当于 KL 散度。然而，在我们的实验中，我们评估了更一般的 f-divergence 产生的损失。

Distillation uses a large teacher model's class probabilities as soft labels to train a smaller student model (Hinton et al., 2015). In the LM context, such soft labels are the teacher model's next-token probabilities, which can be more effective than one-hot labels as they provide richer information about the likely next tokens. We use a T5-XL model (800M parameters) as the teacher and a T5-base model (250M parameters) as the student; prior to distillation, both models were SFT'ed on XSum as in Agarwal et al. (2024). We then fit the student model with the loss (16), where $\phi(y_t)$ in the loss are the soft labels produced by the teacher model.
蒸馏使用大型教师模型的类别概率作为软标签来训练较小的学生模型 (Hinton et al., 2015)。在 LM 上下文中，此类软标签是教师模型的下一个 token 概率，它们比独热标签更有效，因为它们提供了有关可能的下一个 token 的更丰富信息。我们使用 T5-XL 模型（800M 个参数）作为教师，使用 T5-base 模型（250M 个参数）作为学生；在蒸馏之前，这两个模型都在 XSum 上进行了 SFT，如 Agarwal et al. (2024) 中所示。然后，我们使用损失 (16) 来拟合学生模型，其中损失中的 $\phi(y_t)$ 是教师模型生成的软标签。

For both SFT and distillation, we adopted the XSum settings from Agarwal et al. (2024, Appendix A.3). Following Chowdhery et al. (2023) and Agarwal et al. (2024), we evaluated summarization quality using the ROUGE-2 score (Lin, 2004) with temperature-based sampling (temperature=1). Figure 3 shows results for SFT (unhatched) and distillation (hatched). Distillation consistently outperformed SFT. Critically, Chi-squared and α-divergence ($\alpha = 1.5$) losses outperformed KL divergence for both training paradigms.
对于 SFT 和蒸馏，我们采用了 Agarwal et al. (2024,

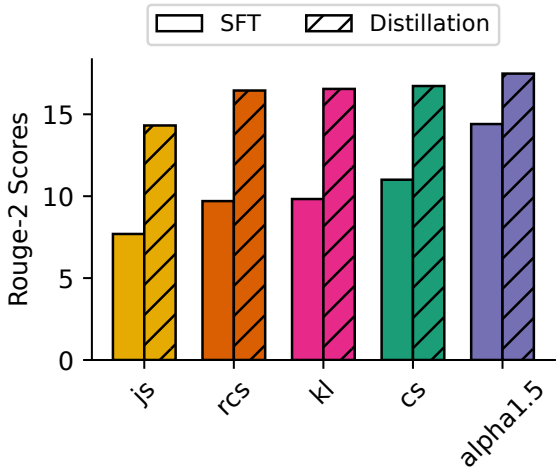

Figure 3. Comparing the effect of different f-divergence generated losses used for SFT and distillation training of LMs. The f-divergence generated losses are Jensen–Shannon (js), reverse Chi-square (rcs), Kullback–Leibler (kl), Chi-square (cs), and α-divergence with $\alpha = 1.5$ (alpha1.5)

比较用于 SFT 和 LM 蒸馏训练的不同 f 散度产生的损失的效果。f 散度产生的损失包括 Jensen-Shannon (js)、反向卡方 (rcs)、Kullback-Leibler (kl)、卡方 (cs) 和 α 散度，其中 $\alpha = 1.5$ (alpha1.5)

附录 A.3) 中的 XSum 设置。遵循 Chowdhery et al. (2023) 和 Agarwal et al. (2024)，我们使用基于温度采样（温度 = 1）的 ROUGE-2 分数 (Lin, 2004) 评估摘要质量。图 3 显示了 SFT（未阴影）和蒸馏（阴影）的结果。蒸馏始终优于 SFT。至关重要的是，对于两种训练范式，卡方和 α 散度（$\alpha = 1.5$）损失均优于 KL 散度。

Comparing f-softargmax variants for decoding. In addition to the loss function used at the training of LMs, another key factor that influences the performance of LMs is how one decode responses from them. While the training of a LM uses a fixed f-divergence loss, when decoding, one can choose different f-softargmax variants to turn LM logits into next-token probabilities for sampling. In our prior experiments of SFT and distillation (Figure 3), at decoding time we used the f-softargmax corresponding to each f-divergence loss function used at training time. But this raises a question: Do the performance differences primarily come from the f-divergence loss used during training or from the usage of different f-softargmax at decoding? To study this, we perform SFT with different f-divergence generated losses as before, but then decode from the trained LM using either the associated f-softargmax or the classical (KL-based) softargmax. Figure 4 shows the results. Decoding with the two types of f-softargmax variants yields nearly identical results. Indeed, note that the two bars of KL divergence are slightly different only due to the effect of random sampling; the observed performance differences in the case of other divergences are comparable to those seen in the KL case. This suggests that the f-divergence used during training primarily contributes to the performance differences of models.

除了在训练 LM 时使用的损失函数之外，影响 LM 性能的另一个关键因素是如何解码响应。虽然 LM 的训练使用固定的 f-divergence 损失，但在解码时，可以选择不同的 f-softargmax 变体将 LM logits 转换为下一个 token 概率以进行采样。在我们之前的 SFT 和蒸馏实验中（图 3），在解码时，我们使用了与训练时使用的每个 f-divergence 损失函数相对应的 f-softargmax。但这引发了一个问题：性能差异主要来自训练期间使用的 f-divergence 损失还是来自解码时使用不同的 f-softargmax？为了研究这个问题，我们像以前一样使用不同的 f-divergence 生成损失执行 SFT，然后使用相关的 f-softargmax 或经典（基于 KL）softargmax 从训练后的 LM 解码。图 4 显示了结果。使用两种类型的 f-softargmax 变体进行解码可获得几乎相同的结果。事实上，请注意，KL 散度的两个条形图仅由于随机采样的影响而略有不同；在其他散度的情况下观察到的性能差异与 KL 情况下的性能差异相当。这表明训练期间使用的 f-散度主要导致了模型的性能差异。

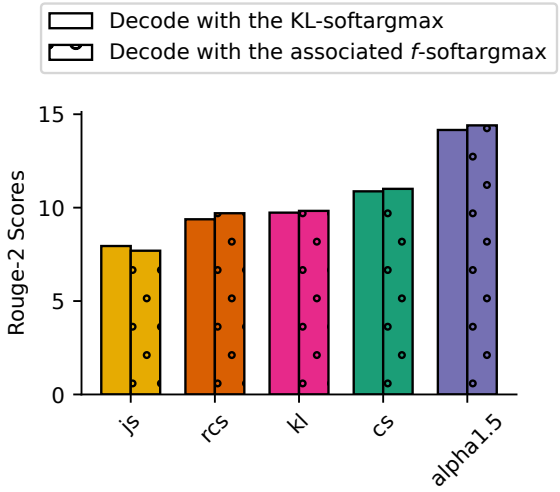

Figure 4. Decoding with the classical (KL-based) softargmax performs similarly to decoding with f-softargmax associated with the loss used for training. This suggests that the choice of f-divergence loss during training, not the decoding method, primarily drives performance differences.

使用经典（基于 KL）softargmax 进行解码的性能与使用与用于训练的损失相关的 f-softargmax 进行解码的性能类似。这表明，训练期间对 f-divergence 损失的选择，而不是解码方法，主要导致了性能差异。

4.3. Summary of empirical findings

We summarize the main take-aways from our experiments.
我们总结了实验的主要结论。

- We found that the loss function generated by the α-divergence with $\alpha = 1.5$ works well across all tasks we tried. While equivalent to Tsallis negentropy ($\alpha = 1.5$) with a uniform reference measure (Blondel et al., 2020; Peters et al., 2019), its effectiveness across several language modeling tasks is novel. Its good performance could intuitively come from the fact that it is a middle ground between the logistic loss ($\alpha = 1$) and the sparsemax loss ($\alpha = 2$). Figure 5 confirms that accuracy is maximized around $\alpha = 1.5$.
  我们发现，由 α 散度（$\alpha = 1.5$）生成的损失函数在我们尝试的所有任务中都表现良好。虽然它相当于具有统一参考度量 (Blondel et al., 2020; Peters et al., 2019) 的 Tsallis 负熵（$\alpha = 1.5$），但它在多个语言建模任务中的有效性是新颖的。它的良好表现可以直观地归因于这样一个事实：它是逻辑损失（$\alpha = 1$）和 sparsemax 损失（$\alpha = 2$）之间的中间地带。图 5 证实，准确率在 $\alpha = 1.5$ 左右达到最大化。

- We successfully fine-tuned LMs with various f-divergence generated losses, despite the fact the pretraining was carried out using the cross-entropy loss. This enables direct application of our f-divergence generated losses to pretrained, open-weight LMs.
  尽管预训练是使用交叉熵损失进行的，但我们成功地使用各种 f 发散度产生的损失对 LM 进行了微调。这使得我们能够将 f 发散度产生的损失直接应用于预训练的开放权重 LM。

- We show that, during text generation, standard softargmax yielded good performance even when the model is finetuned with f-divergence generated losses. This is surprising because (7) tells us that we should in principle use the f-softargmax operator associated with the loss. This empirical finding opens up the possibility to use our losses without changing the inference code, which is convenient when working with open-weight LMs.
  我们表明，在文本生成过程中，即使使用 f 散度产生的损失对模型进行微调，标准 softargmax 也能产生良好的性能。这令人惊讶，因为 (7) 告诉我们，原则上我们应该使用与损失相关的 f-softargmax 运算符。这一经验发现开辟了在不更改推理代码的情况下使用我们的损失的可能性，这在使用开放权重 LM 时非常方便。

- Loss functions generated by f-divergences not considered before (Jensen-Shannon, squared Hellinger) did not lead to better accuracy. The KL and $\alpha = 1.5$ generated losses consistently achieved the best results across tasks.
  之前未考虑的 f 散度生成的损失函数（Jensen-Shannon、平方 Hellinger）并未带来更好的准确度。KL 和 $\alpha = 1.5$ 生成的损失在各个任务中始终取得最佳结果。

5. Related work

Loss functions based on f-divergences. Nguyen et al. (2009) studied surrogate losses for binary classification based on f-divergences. However, this approach is not straightforward to generalize to the muticlass setting, as it requires multi-distribution extensions of f-divergences (Garcia-Garcia & Williamson, 2012; Duchi et al., 2018). Recently, f-divergences have been used for distillation in combination with a classical softargmax (Agarwal et al., 2024). In our notation, this defines a loss $\boldsymbol{\theta} \mapsto D_f(\boldsymbol{y}, \mathrm{softargmax}(\boldsymbol{\theta}))$. However, this does not result in a convex loss in $\boldsymbol{\theta}$. Belousov (2017) briefly studied the idea of generating Bregman divergences from f-divergences. However, Bregman divergences need to be explicitly composed with a softargmax and this composition is again usually not convex. In contrast, in our approach, each loss $\boldsymbol{\theta} \mapsto \ell_f(\boldsymbol{\theta}, \boldsymbol{y}; \boldsymbol{q})$ is convex and is implicitly associated with the corresponding $\mathrm{softargmax}_f$ operator.
Nguyen et al. (2009) 研究了基于 f-divergences 的二分类替代损失。然而，这种方法不能直接推广到多类设置，因为它需要 f-divergences 的多分布扩展 (Garcia-Garcia & Williamson, 2012; Duchi et al., 2018)。最近，f-divergences 已与经典的 softargmax (Agarwal et al., 2024) 结合用于蒸馏。在我们的符号中，这定义了损失 $\boldsymbol{\theta} \mapsto D_f(\boldsymbol{y}, \mathrm{softargmax}(\boldsymbol{\theta}))$。然而，这不会导致 $\boldsymbol{\theta}$ 中的凸损失。Belousov (2017) 简要研究了从 f-divergences 生成 Bregman 散度的想法。然而，Bregman 散度需要明确地与 softargmax 组合，而且这种组合通常也不是凸的。相反，在我们的方法中，每个损失 $\boldsymbol{\theta} \mapsto \ell_f(\boldsymbol{\theta}, \boldsymbol{y}; \boldsymbol{q})$ 都是凸的，并且隐式地与相应的 $\mathrm{softargmax}_f$ 运算符相关联。

LM alignment with f-divergences. In RLHF, given a learned reward model $r\colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, the optimal policy is
在 RLHF 中，给定一个学习到的奖励模型 $r\colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$，最佳策略是

$$\boldsymbol{p}^\star(\cdot|\boldsymbol{x}) \coloneqq \operatorname*{argmax}_{\boldsymbol{p} \in \triangle^{|\mathcal{Y}|}} \mathbb{E}_{\boldsymbol{y}\sim\boldsymbol{p}}[r(\boldsymbol{x}, \boldsymbol{y})] - \mathrm{KL}(\boldsymbol{p}, \boldsymbol{q}(\cdot|\boldsymbol{x})),$$

where $\boldsymbol{x} \in \mathcal{X}$ is a given prompt and $\boldsymbol{q}(\cdot|\boldsymbol{x}) \in \triangle^{|\mathcal{Y}|}$ is a reference conditional distribution, usually a pretrained model. This approach can be generalized to

f-divergences (Go et al., 2023). In our notation, this can be written as
我们建议使用 f 散度作为生成损失函数和相关运算符 (f-softmax、f-softargmax、f-softplus 和 f-sigmoid) 的正则化。我们的建议分别在 logistic、sparsemax 和 entmax 损失函数与 KL、卡方和 α 散度之间建立了联系。得益于这种观点，我们的建议将这些损失推广到非均匀类权重。其他 f 散度的选择使我们能够创建全新的损失函数。总体而言，我们发现由 $\alpha = 1.5$ 散度生成的损失函数在我们尝试的所有任务上的效果与交叉熵损失相当或更好。

$$\boldsymbol{p}^\star(\cdot|\boldsymbol{x}) = \operatorname{softargmax}_f(r(\boldsymbol{x}, \cdot), \boldsymbol{q}(\cdot|\boldsymbol{x})).$$

To avoid learning a separate reward model altogether, we can use direct preference optimization or DPO (Rafailov et al., 2024), which was generalized to f-divergences (Wang et al., 2024). Proposition 1 can be seen as generalization of (Wang et al., 2024, Theorem 1) that supports $0 \in \operatorname{dom}(f')$. In this paper, our focus was on the SFT and distillation steps, which usually precede RLHF. Unlike these works, we propose to use the f-softargmax operator and the corresponding Fenchel–Young loss for pretraining, SFT and distillation in the space of tokens, making SGD-based optimization easy.
为了避免完全学习单独的奖励模型，我们可以使用直接偏好优化或 DPO (Rafailov et al., 2024)，它被推广到 f-divergences (Wang et al., 2024)。命题 1 可以看作是 (Wang et al., 2024, Theorem 1) 的推广，它支持 $0 \in \operatorname{dom}(f')$。在本文中，我们的重点是 SFT 和蒸馏步骤，这些步骤通常在 RLHF 之前。与这些工作不同，我们建议在 token 空间中使用 f-softargmax 运算符和相应的 Fenchel–Young 损失进行预训练、SFT 和蒸馏，从而使基于 SGD 的优化变得容易。

6. Conclusion

We proposed to use f-divergences as regularization for generating loss functions and associated operators (f-softmax, f-softargmax, f-softplus and f-sigmoid). Our proposal establishes a link between the logistic, sparsemax and entmax loss functions and the KL, chi-square and α divergences, respectively. Thanks to this perspective, our proposal generalizes these losses to non-uniform class weights. Other choices of f-divergences allowed us to create entirely new loss functions. Overall, we found that the loss function generated by the $\alpha = 1.5$ divergence worked comparably or better to the cross-entropy loss on all tasks we tried.
我们建议使用 f 散度作为生成损失函数和相关运算符 (f-softmax、f-softargmax、f-softplus 和 f-sigmoid) 的正则化。我们的建议分别在 logistic、sparsemax 和 entmax 损失函数与 KL、卡方和 α 散度之间建立了联系。得益于这种观点，我们的建议将这些损失推广到非均匀类权重。其他 f 散度的选择使我们能够创建全新的损失函数。总体而言，我们发现由 $\alpha = 1.5$ 散度生成的损失函数在我们尝试的所有任务上的效果与交叉熵损失相当或更好。

Acknowledgements

We thank Quentin Berthet and Olivier Bousquet for feedback on this paper. We thank Lechao Xiao for discussions and help with the NanoDO experiments.

Impact Statement

This paper explores f-divergence generated loss functions for classification or language modelling. We do not foresee any specific ethical or societal implications arising directly from this work.

Author contributions

- Vincent Roulet: implementation, experiments, proofs, writing.
- Tianlin Liu: implementation, experiments, writing.
- Nino Vieillard: implementation.
- Michaël Sander: writing, experiments.
- Mathieu Blondel: initial project idea, implementation, proofs, writing.

References

Agarwal, R., Vieillard, N., Zhou, Y., Stanczyk, P., Garea, S. R., Geist, M., and Bachem, O. On-policy distillation of language models: learning from self-generated mistakes. In Proceedings of the International Conference on Learning Representations (ICLR), 2024.

Ali, S. M. and Silvey, S. D. A general class of coefficients of divergence of one distribution from another. Journal of the Royal Statistical Society: Series B (Methodological), 28(1):131–142, 1966.

Belousov, B. Bregman divergence of alpha-divergence, 2017. URL http://www.boris-belousov.net/2017/04/16/bregman-divergence/.

Belousov, B. and Peters, J. f-divergence constrained policy improvement. arXiv preprint arXiv:1801.00056, 2017.

Blondel, M. and Roulet, V. The elements of differentiable programming. arXiv preprint arXiv:2403.14606, 2024.

Blondel, M., Martins, A. F., and Niculae, V. Learning with Fenchel–Young losses. Journal of Machine Learning Research, 21(35):1–69, 2020.

Blondel, M., Berthet, Q., Cuturi, M., Frostig, R., Hoyer, S., Llinares-Lopez, F., Pedregosa, F., and Vert, J.-P. Efficient and modular implicit differentiation. In Proceedings of the Conference on Neural Information Processing Systems (NeurIPS), pp. 5230–5242. Curran Associates, Inc., 2022.

Bradley, R. A. and Terry, M. E. Rank analysis of incomplete block designs: I. the method of paired comparisons. Biometrika, 39(3/4):324–345, 1952.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. PaLM: Scaling language modeling with pathways. Journal of Machine Learning Research, 24(240):1–113, 2023.

Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., and Amodei, D. Deep reinforcement learning from human preferences. In Proceedings of the Conference on Neural Information Processing Systems (NeurIPS), 2017.

Cichocki, A. and Amari, S.-i. Families of alpha-beta- and gamma-divergences: Flexible and robust measures of similarities. Entropy, 12(6):1532–1568, 2010.

Csiszár, I. On information-type measure of difference of probability distributions and indirect observations. Studia Sci. Math. Hungar., 2:299–318, 1967.

Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A. P., Caron, M., Geirhos, R., Alabdulmohsin, I., Jenatton, R., Beyer, L., Tschannen, M., Arnab, A., Wang, X., Riquelme Ruiz, C., Minderer, M., Puigcerver, J., Evci, U., Kumar, M., Steenkiste, S. V., Elsayed, G. F., Mahendran, A., Yu, F., Oliver, A., Huot, F., Bastings, J., Collier, M., Gritsenko, A. A., Birodkar, V., Vasconcelos, C. N., Tay, Y., Mensink, T., Kolesnikov, A., Pavetic, F., Tran, D., Kipf, T., Lucic, M., Zhai, X., Keysers, D., Harmsen, J. J., and Houlsby, N. Scaling vision transformers to 22 billion parameters. In Proceedings of the International Conference on Machine Learning (ICML), pp. 7480–7512, 2023.

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

Garcia-Garcia, D. and Williamson, R. C. Divergences and risks for multiclass experiments. In Proceedings of the Annual Conference on Learning Theory, pp. 28–1. JMLR Workshop and Conference Proceedings, 2012.

Go, D., Korbak, T., Kruszewski, G., Rozen, J., Ryu, N., and Dymetman, M. Aligning language models with preferences through f-divergence minimization. In Proceedings of the International Conference on Machine Learning (ICML). PMLR, 2023.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531, 2015.

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., de Las Casas, D., Hendricks, L. A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., van den Driessche, G., Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., Vinyals, O., Rae, J., and Sifre, L. An empirical analysis of compute-optimal large language model training. In Proceedings of the Conference on Neural Information Processing Systems (NeurIPS), volume 35, pp. 30016–30030. Curran Associates, Inc., 2022.

Krantz, S. G. and Parks, H. R. The implicit function theorem: history, theory, and applications. Springer Science & Business Media, 2002.

Lin, C.-Y. Rouge: A package for automatic evaluation of summaries. In Text Summarization Branches Out, pp. 74–81, 2004.

Liu, P. J., Novak, R., Lee, J., Wortsman, M., Xiao, L., Everett, K., Alemi, A. A., Kurzeja, M., Marcenac, P., Gur, I., Kornblith, S., Xu, K., Elsayed, G., Fischer, I., Pennington, J., Adlam, B., and Dickstein, J.-S. Nanodo: A minimal transformer decoder-only language model implementation in JAX., 2024. URL http://github.com/google-deepmind/nanodo.

Martins, A. and Astudillo, R. From softmax to sparsemax: A sparse model of attention and multi-label classification. In Proceedings of the International Conference on Machine learning (ICML), pp. 1614–1623. PMLR, 2016.

Narayan, S., Cohen, S. B., and Lapata, M. Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1797–1807. Association for Computational Linguistics, 2018.

Nguyen, X., Wainwright, M. J., and Jordan, M. I. On surrogate loss functions and f-divergences. The Annals of Statistics, 37(2):876 – 904, 2009.

Peters, B., Niculae, V., and Martins, A. F. Sparse sequence-to-sequence models. In Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), 2019.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. OpenAI blog, 1(8):9, 2019.

Rafailov, R., Sharma, A., Mitchell, E., Manning, C. D., Ermon, S., and Finn, C. Direct preference optimization: Your language model is secretly a reward model. In Proceedings of the Conference on Neural Information Processing Systems (NeurIPS), 2024.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Rényi, A. On measures of entropy and information. In Proceedings of the fourth Berkeley symposium on mathematical statistics and probability, volume 1: contributions to the theory of statistics, volume 4, pp. 547–562. University of California Press, 1961.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. ImageNet large scale visual recognition challenge. International Journal of Computer Vision, 115:211–252, 2015.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. Neurocomputing, 568:127063, 2024.

Terjék, D. Moreau–Yosida f-divergences. In Proceedings of the International Conference on Machine Learning, pp. 10214–10224. PMLR, 2021.

Wang, C., Jiang, Y., Yang, C., Liu, H., and Chen, Y. Beyond reverse KL: Generalizing direct preference optimization with diverse divergence constraints. In Proceedings of the International Conference on Learning Representations (ICLR), 2024.

Wortsman, M., Liu, P. J., Xiao, L., Everett, K. E., Alemi, A. A., Adlam, B., Co-Reyes, J. D., Gur, I., Kumar, A., Novak, R., Pennington, J., Sohl-Dickstein, J., Xu, K., Lee, J., Gilmer, J., and Kornblith, S. Small-scale proxies for large-scale transformer training instabilities. In Proceedings of the International Conference on Learning Representations (ICLR), 2024.

A. Experimental details and additional experiments

A.1. ImageNet classification

When using the KL divergence, our training yields 76.87% accuracy on ImageNet. This result matches previous benchmarks of ResNet50 trained on ImageNet, such as https://github.com/google/flax/tree/main/examples/imagenet.
使用 KL 散度时，我们的训练在 ImageNet 上的准确率为 76.87%。此结果与之前在 ImageNet 上训练的 ResNet50 的基准测试相匹配，例如 https://github.com/google/flax/tree/main/examples/imagenet。

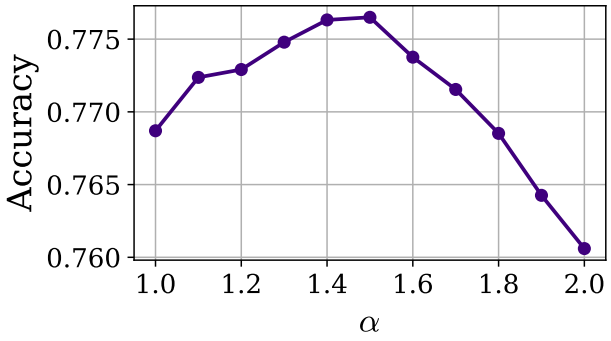

Figure 5. Validation accuracy on ImageNet when using the α-divergence generated loss, with $\alpha \in [1., 2.]$. Accuracy is maximized near $\alpha = 1.5$.
使用 α 散度产生的损失，在 ImageNet 上进行验证的准确率，其中 $\alpha \in [1., 2.]$。准确率在 $\alpha = 1.5$ 附近达到最大值。

A.2. Pretraining NanoDO models

The 1.2B NanoDO model's architecture and hyperparameters follow from the setup in Wortsman et al. (2024). The model is similar to GPT2 (Radford et al., 2019) but incorporates modern features like rotary positional embeddings (Su et al., 2024) and qk-layernorm (Dehghani et al., 2023). To evaluate the trained model, we use the next-token prediction accuracy on the validation dataset. Our results (Table 3) show that α-divergence is on par, and slightly more performant, than the classifical KL approach.
1.2B NanoDO 模型的架构和超参数遵循 Wortsman et al. (2024) 中的设置。该模型类似于 GPT2 (Radford et al., 2019)，但结合了旋转位置嵌入 (Su et al., 2024) 和 qk-layernorm (Dehghani et al., 2023) 等现代功能。为了评估训练后的模型，我们在验证数据集上使用下一个标记预测准确率。我们的结果（表格 3）表明，α-divergence 与经典 KL 方法相当，且性能略高。

Table 3. Next-token prediction accuracy

Divergence	Accuracy (%)
Kullback–Leibler	48.66
Chi-square	46.75
α-divergence ($\alpha = 1.5$)	48.70

A.3. Computational cost

Bisection convergence. We experimentally validate the exponential convergence of our proposed Algorithm 1. Our results are in Figure 6 and confirm the theoretical convergence rate.
我们通过实验验证了我们提出的算法 1 的指数收敛性。我们的结果如图 6 所示，并证实了理论收敛速度。

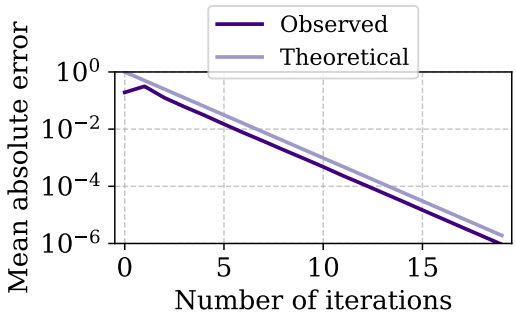

Figure 6. Error for computing the root in Propositon 1 using our bisection based Algorithm 1 as a function of the number of iterations. We use the output with 30 iterations as a proxy for the true root. We compare the measured error (dark purple) with the theoretical 2^{-t} error (light purple). This experiment is run on TPU.
使用基于二分法的算法 1 计算 Propositon 1 中的根的误差是迭代次数的函数。我们使用 30 次迭代的输出作为真实根的代理。我们将测量误差（深紫色）与理论 2^{-t} 误差（浅紫色）进行比较。此实验在 TPU 上运行。

Overall cost. We propose an experiment validating that our proposed losses generated by f-divergences and associated operators do not affect the runtime of standard learning pipelines. For this, we consider a Residual Network (He et al., 2016) ResNet18, with either the standard softargmax or the softargmax associated with the $\alpha = 1.5$ divergence. For different batch sizes, we profile the time needed for running both the model and the f-softargmax. Results in Figure 7 show that the runtimes are approximately equal.
我们提出了一个实验来验证我们提出的由 f 散度和相关运算符产生的损失不会影响标准学习管道的运行时间。为此，我们考虑使用残差网络 (He et al., 2016) ResNet18，其标准 softargmax 或与 $\alpha = 1.5$ 散度相关的 softargmax。对于不同的批次大小，我们分析了运行模型和 f-softargmax 所需的时间。图 7 中的结果显示运行时间大致相等。

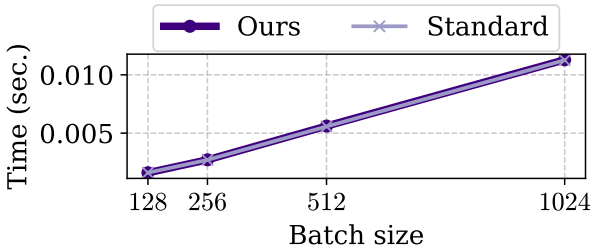

Figure 7. Profiled time for an input of shape $(b, 224, 224, 3)$ to be processed by a ResNet18 followed by f-softargmax coresponding the 1.5-divergence (dark purple) or the standard Kullback–Leibler divergence (light purple, in which case we use the standard JAX implementation) for different batches size b. This experiment is run on TPU.
形状为 $(b, 224, 224, 3)$ 的输入由 ResNet18 处理，然后是 f-softargmax，对应于不同批次大小 b 的 1.5 散度（深紫色）或标准 Kullback–Leibler 散度（浅紫色，在这种情况下我们使用标准 JAX 实现）。此实验在 TPU 上运行。

A.4. Impact of prior distribution on sigmoids

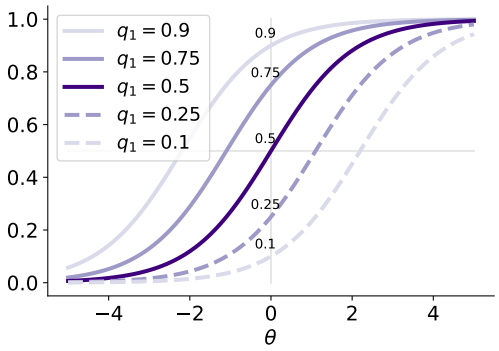

Figure 8. Illustration of $\theta \mapsto \text{sigmoid}_f(\theta; (1-q_1, q_1))$ for $q_1 \in \{0.1, 0.25, 0.5, 0.75, 0.9\}$ and for $f(u) = u \log u$, the generating function of the KL divergence. We see that $\text{sigmoid}_f(\theta; (1-q_1, q_1))$ at $\theta = 0$ is equal to q_1. Intuitively, if a model is uncertain and produces a value of $\theta = 0$, then the model outputs q_1, the prior probability of the positive class.
对于 $q_1 \in \{0.1, 0.25, 0.5, 0.75, 0.9\}$ 和 $f(u) = u \log u$（KL 散度的生成函数），$\theta \mapsto \text{sigmoid}_f(\theta; (1-q_1, q_1))$ 的图示。我们看到，在 $\theta = 0$ 时，$\text{sigmoid}_f(\theta; (1-q_1, q_1))$ 等于 q_1。直观地讲，如果模型不确定并产生 $\theta = 0$ 的值，则模型输出 q_1，即正类的先验概率。

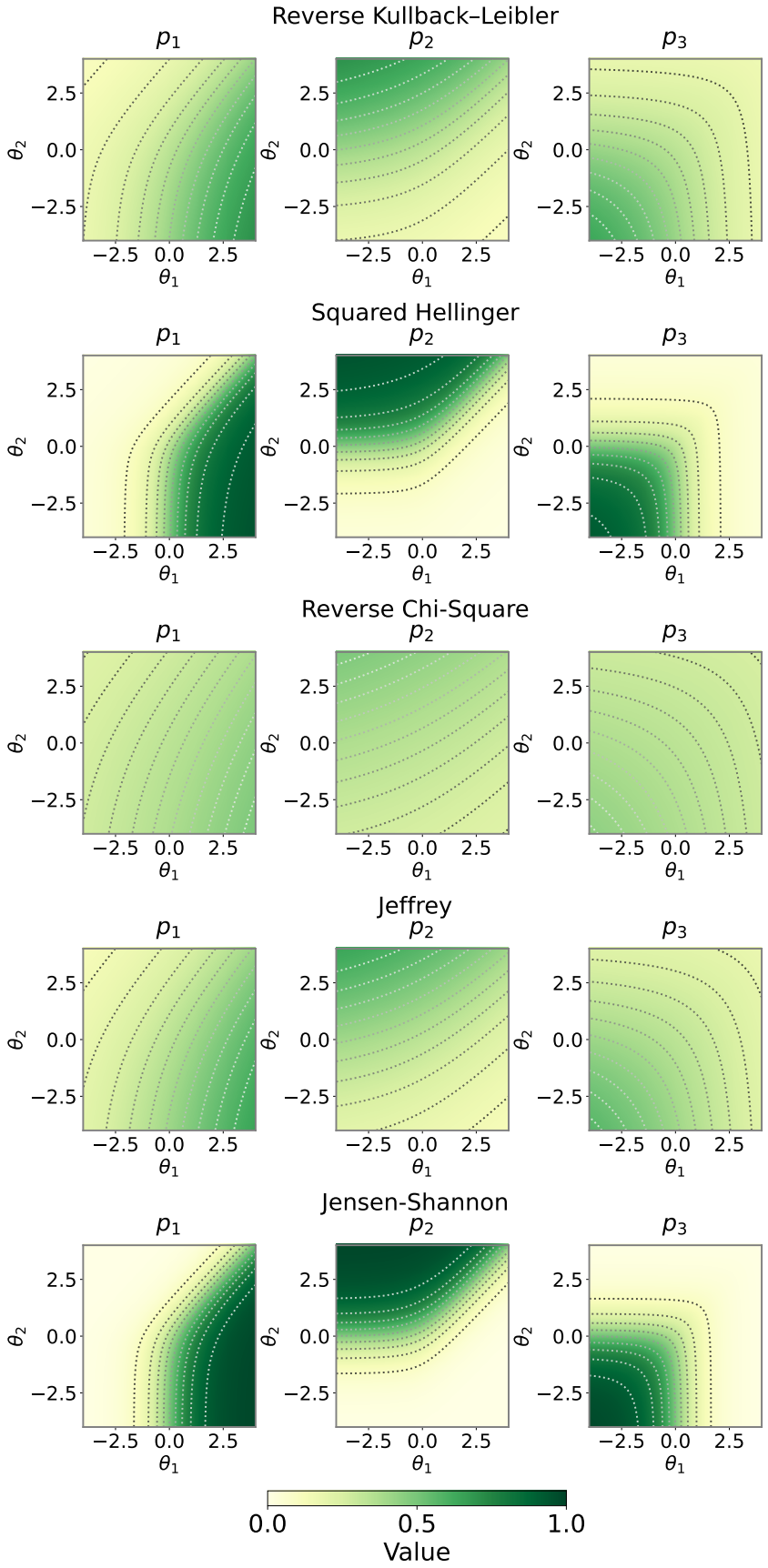

Figure 9. Illustration of $(p_1, p_2, p_3) = \text{softargmax}_f(\theta_1, \theta_2, 0; \boldsymbol{q})$ when varying $\theta_1, \theta_2 \in \mathbb{R}$ for diverse f-divergences and with $\boldsymbol{q} = (1, 1, 1)$.

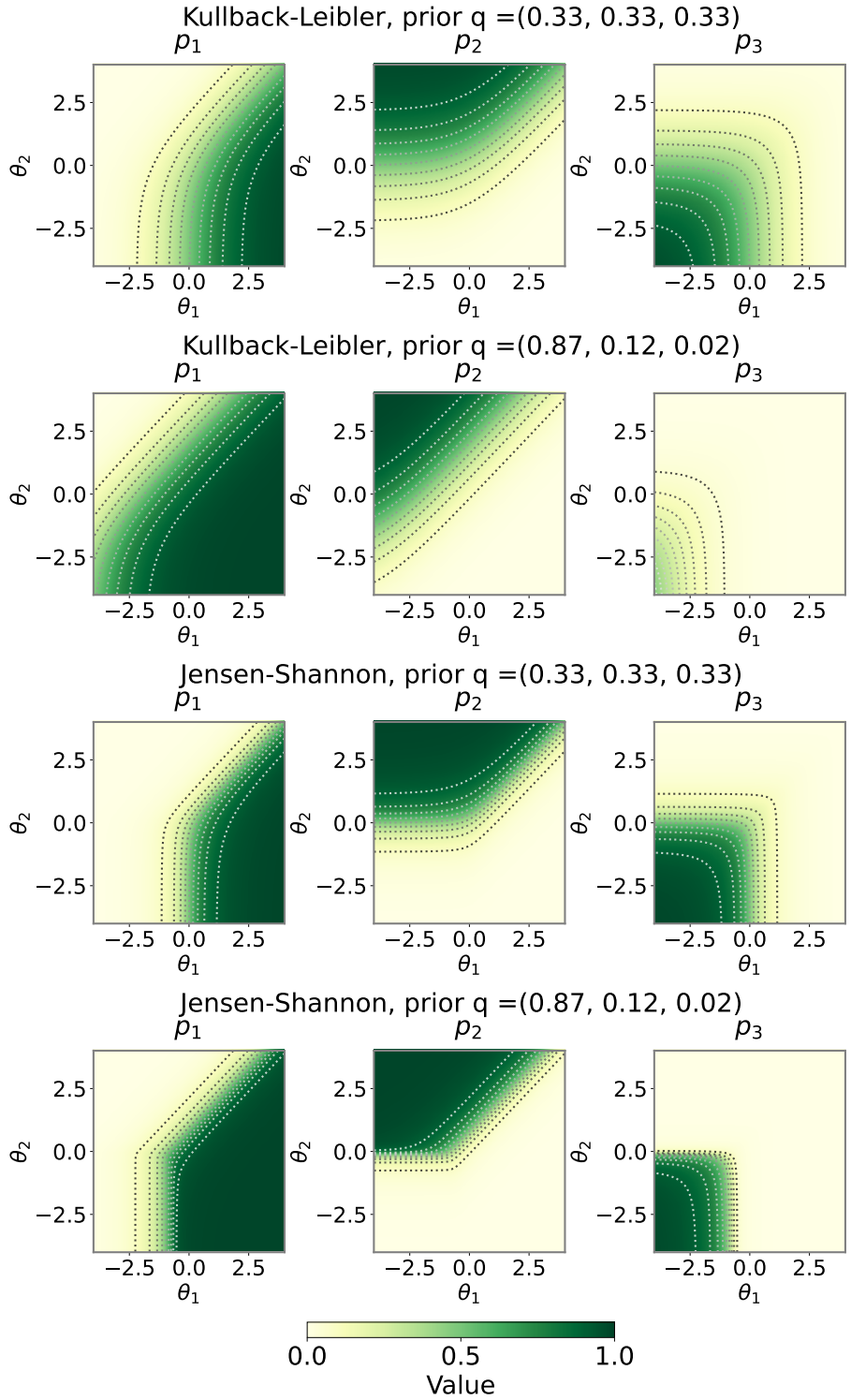

Figure 10. Illustration of $(p_1, p_2, p_3) = \text{softargmax}_f(\theta_1, \theta_2, 0; \boldsymbol{q})$ when varying $\theta_1, \theta_2 \in \mathbb{R}$ for diverse f-divergences and priors $\boldsymbol{q}$.

B. Examples of f-divergences

- KL
  - $f(u) = u \log u$, $\mathrm{dom}(f) = \mathbb{R}_+$
  - $f'(u) = \log u + 1$
  - $f_*(v) = \exp(v-1)$, $\mathrm{dom}(f_*) = \mathbb{R}$
  - $f'_*(v) = \exp(v-1)$
  - $D_f(p,q) = \mathrm{KL}(p,q) \coloneqq \langle p, \log p\rangle - \langle p, \log q\rangle$
- Generalized KL
  - $f(u) = u \log u - (u-1)$, $\mathrm{dom}(f) = \mathbb{R}_+$
  - $f'(u) = \log u$
  - $f_*(v) = \exp(v) - 1$, $\mathrm{dom}(f_*) = \mathbb{R}$
  - $f'_*(v) = \exp(v)$
  - $D_f(p,q) = \mathrm{GKL}(p,q) \coloneqq \langle p, \log p\rangle - \langle p, \log q\rangle - \langle p, 1\rangle + \langle q, 1\rangle$
- Reverse KL
  - $f(u) = -\log u$, $\mathrm{dom}(f) = \mathbb{R}_+$
  - $f'(u) = -1/u$
  - $f_*(v) = -1 - \log(-v)$, $\mathrm{dom}(f_*) = \mathbb{R}_-$ (c.f. Proposition 2)
  - $f'_*(v) = -1/v$
  - $D_f(p,q) = \mathrm{KL}(q,p)$
- Jeffrey
  - $f(u) = (u-1)\log u$, $\mathrm{dom}(f) = \mathbb{R}_+$
  - $f'(u) = \log u + 1 - \frac{1}{u}$
  - $f_*(v) = \frac{1}{W(\exp(1-v))} + \log \frac{1}{W(\exp(1-v))} - 1$, $\mathrm{dom}(f_*) = \mathbb{R}$ (c.f. Proposition 3)
  - $f'_*(v) = \frac{1}{W(\exp(1-v))}$
  - $D_f(p,q) = \mathrm{KL}(p,q) + \mathrm{KL}(q,p) = \langle \log p - \log q, p - q\rangle$
  - Remark: W is the Lambert function
- Jensen-Shannon
  - $f(u) = u \log u - (u+1)\log\left(\frac{u+1}{2}\right)$, $\mathrm{dom}(f) = \mathbb{R}_+$
  - $f'(u) = \log u - \log\left(\frac{u+1}{2}\right) = \log\left(\frac{2u}{u+1}\right)$
  - $f_*(v) = -\log(2 - \exp(v))$, $\mathrm{dom}(f_*) = (-\infty, \log 2)$ (c.f. Proposition 4)
  - $f'_*(v) = \frac{\exp(v)}{2-\exp(v)} = \frac{1}{2\exp(-v)-1}$
  - $D_f(p,q) = 2 \cdot \mathrm{JS}(p,q) \coloneqq \mathrm{KL}(p,m) + \mathrm{KL}(q,m) = \left\langle p, \log\left(\frac{2p}{p+q}\right)\right\rangle + \left\langle q, \log\left(\frac{2q}{p+q}\right)\right\rangle$, where $m \coloneqq \frac{1}{2}(p+q)$.
- Squared Hellinger
  - $f(u) = (\sqrt{u}-1)^2 = u - 2\sqrt{u} + 1$, $\mathrm{dom}(f) = \mathbb{R}_+$
  - $f'(u) = 1 - \frac{1}{\sqrt{u}}$
  - $f_*(v) = \frac{v}{1-v}$, $\mathrm{dom}(f_*) = (-\infty, 1)$ (c.f. Proposition 5)
  - $f'_*(v) = \frac{1}{(1-v)^2}$
  - $D_f(p,q) = 2 \cdot \mathrm{SH}(p,q) = \sum_{j=1}^k (\sqrt{p_j} - \sqrt{q_j})^2$
- Chi-squared divergence (a.k.a. Pearson divergence)
  - $f(u) = \frac{1}{2}(u^2 - 1)$, $\mathrm{dom}(f) = \mathbb{R}$

- $f'(u) = u$
- $f_*(v) = \frac{1}{2}(v^2+1)$, $\mathrm{dom}(f_*) = \mathbb{R}$ (c.f. Proposition 6)
- $f'_*(v) = v$
- $D_f(p,q) = \frac{1}{2}\chi^2(p,q) \coloneqq \frac{1}{2}\sum_{j=1}^k \frac{(p_j-q_j)^2}{q_j}$
- Remark: other possible generating functions are $f(u) = \frac{1}{2}(u-1)^2$ or $f(u) = \frac{1}{2}(u^2-u)$. We choose $f(u) = \frac{1}{2}(u^2-1)$ as it satisfies $f'(0) = 0$.

- Reverse chi-squared divergence (a.k.a. Neyman divergence)
  - $f(u) = \frac{1}{2}(\frac{1}{u}-1)$, $\mathrm{dom}(f) = \mathbb{R}_+$
  - $f'(u) = -\frac{1}{2u^2}$
  - $f_*(v) = -\sqrt{-2v} + \frac{1}{2}$, $\mathrm{dom}(f_*) = \mathbb{R}_-$ (c.f. Proposition 7)
  - $f'_*(v) = \frac{1}{\sqrt{-2v}}$
  - $D_f(p,q) = \frac{1}{2}\chi^2(q,p)$

- α-divergences
  - $f(u) = \begin{cases} \frac{(u^\alpha-1)-\alpha(u-1)}{\alpha(\alpha-1)} + \iota_{\mathbb{R}_+}(u) & \text{if } \alpha \neq 1 \\ u\log u - (u-1) & \text{if } \alpha = 1 \end{cases}$, $\mathrm{dom}(f) = \mathbb{R}_+$
  - $f'(u) = \log_\alpha(u) \coloneqq \begin{cases} \frac{u^{\alpha-1}-1}{\alpha-1} & \text{if } \alpha \neq 1 \\ \log u & \text{if } \alpha = 1 \end{cases}$
  - $f_*(v) = \begin{cases} \frac{(1+(\alpha-1)v)^{\frac{\alpha}{\alpha-1}}-1}{\alpha} & \text{if } \alpha \in (0,1) \\ \exp(v) & \text{if } \alpha = 1 \\ \frac{(1+(\alpha-1)v)_+^{\frac{\alpha}{\alpha-1}}-1}{\alpha} & \text{if } \alpha \in (1,+\infty), \text{ where } x_+ = \max(x,0) \end{cases}$, (c.f. Proposition 8)
  - $\mathrm{dom}\, f_* = \begin{cases} (-\infty, -\frac{1}{\alpha-1}] & \text{if } \alpha \in (0,1) \\ \mathbb{R} & \text{if } \alpha \geq 1 \end{cases}$
  - $f'_*(v) = \exp_\alpha(v) \coloneqq \begin{cases} (1+(\alpha-1)v)^{\frac{1}{\alpha-1}} & \text{if } \alpha \in (0,1) \\ \exp v & \text{if } \alpha = 1 \\ (1+(\alpha-1)v)_+^{\frac{1}{\alpha-1}} & \text{if } \alpha \in (1,+\infty) \end{cases}$
  - $D_f(p,q) = \frac{1}{\alpha}(\langle p, \log_\alpha(p/q)\rangle - \langle p,1\rangle + \langle q,1\rangle) = \frac{1}{\alpha(\alpha-1)}\left(\langle p^\alpha, q^{\alpha-1}\rangle - \alpha\langle p,1\rangle + (\alpha-1)\langle q,1\rangle\right)$
  - Remark: By analogy with the KL divergence case, the α-divergence can be derived by setting $f'(u) \coloneqq \log_\alpha(u)$ and by integrating. The constant of integration is chosen so as to satisfy $f(1) = 0$.

All functions f above can easily be shown to be strictly convex and differentiable. Note that both properties (strict convexity and differentiability) are defined on the interior of the domain of definition of the function, and therefore non-differentiability on its border (typically 0 on the left) does not matter to assess these properties.
上述所有函数 f 都可轻松证明为严格凸且可微的。请注意，这两个属性（严格凸性和可微性）都是在函数定义域的内部定义的，因此其边界上的非可微性（通常左侧为 0）对于评估这些属性无关紧要。

C. Proofs

C.1. Conjugates

In this section, we derive the conjugate of f,

$$f_*(v) \coloneqq \max_u uv - f(u)$$

for the divergences listed in Appendix B.

Reverse KL divergence.

> Proposition 2. The conjugate of $f(u) = -\log(u)$ for $u \in \operatorname{dom} f = (0, +\infty)$ is $f_*(v) = -1 - \log(-v)$ for $v \in \operatorname{dom} f_* = (-\infty, 0)$.

Proof. For $v > 0$, the function $u \mapsto uv - f(u)$ is unbounded above by taking $u \to +\infty$. For $v < 0$, we have $\lim_{u\to 0}\{uv - f(u)\} = \lim_{u\to+\infty}\{uv - f(u)\} = -\infty$. Hence a maximizer $u^\star$ of $u \mapsto uv - f(u)$ exists and is characterized by $v = f'(u^\star)$. Since $f'(u) = -1/u$, we have

$$v = f'(u) \iff v = -1/u \iff u = -1/v = f'_*(v).$$

We then get

$$f_*(v) = -1 + \log(-1/v) = -1 - \log(-v).$$

□

Jeffrey divergence.

> Proposition 3. The conjugate of $f(u) = (u-1)\log u$ for $u \in \operatorname{dom} f = (0, +\infty)$ is $f_*(v) = \frac{1}{W(\exp(1-v))} + \log \frac{1}{W(\exp(1-v))} - 1$ for $v \in \operatorname{dom} f_* = \mathbb{R}$, where W is the Lambert W function.

Proof. For any $v \in \mathbb{R}$, we have $\lim_{u\to 0}\{uv - f(u)\} = \lim_{u\to+\infty}\{uv - f(u)\} = -\infty$. Therefore, for any $v \in \mathbb{R}$, the function $u \mapsto uv - f(u)$ has a maximizer $u^\star$ characterized by $v = f'(u^\star)$.

Using $f'(u) = \log u + 1 - \frac{1}{u}$, we get

$$v = f'(u) = \log u + 1 - \frac{1}{u} \iff 1 - v = 1/u + \log(1/u) \iff \exp(1-v) = \exp(1/u)1/u.$$

We recall the definition of the Lambert W function.

For $z \geq 0$, $W(z)$ is the inverse of the function $g(w) = w\exp(w)$, i.e., $W(z) = g^{-1}(z)$ and $W^{-1}(w) = g(w)$. We then obtain

$$u = \frac{1}{W(\exp(1-v))} = f'_*(v).$$

Furthermore, we have

$$uv = u\log u + u - 1$$

and therefore, we get

$$\begin{aligned} f_*(v) &= uv - u\log u + \log u \\ &= u + \log u - 1 \\ &= \frac{1}{W(\exp(1-v))} + \log \frac{1}{W(\exp(1-v))} - 1. \end{aligned}$$

□

Jensen-Shannon divergence.

Proposition 4. The conjugate of $f(u) = u\log u - (u+1)\log\left(\frac{u+1}{2}\right)$ for $u \in \operatorname{dom} f = (0, +\infty)$ is $f_*(v) = \log u - v = -\log(2 - \exp(v))$, for $v \in \operatorname{dom} f_* = (-\infty, \log 2)$.

Proof. We have for $u \to +\infty$, $uv - u\log u + (u+1)\log((u+1)/2) \sim uv - u\log u + u\log u/2 \sim uv - u\log 2$, where $\sim$ denotes asymptotic equivalence. Hence for $v > \log 2$, we have $\lim_{u\to+\infty} uv - f(u) = +\infty$. For $v < \log 2$, we have $\lim_{u\to+\infty}\{uv - f(u)\} = \lim_{u\to 0}\{uv - f(u)\} = -\infty$. Hence, for $v < \log 2$, a maximizer $u^\star$ of $u \mapsto uv - f(u)$ exists and is characterized by $v = f'(u^\star)$.

Using $f'(u) = \log\left(\frac{2u}{u+1}\right)$, we get

$$v = f'(u) \iff \exp(v) = \frac{2u}{u+1} \iff u = \frac{\exp(v)}{2 - \exp(v)} = \frac{\exp(0)}{2\exp(-v) - \exp(0)} = f_*'(v).$$

Let us now compute $f_*(v) = uv - u\log u + (u+1)\log\left(\frac{u+1}{2}\right)$. We observe that

$$u + 1 = \frac{2}{2 - \exp(v)} \iff \frac{u+1}{2} = \frac{u}{\exp(v)} \iff \log\left(\frac{u+1}{2}\right) = \log u - v.$$

Therefore

$$(u+1)\log\left(\frac{u+1}{2}\right) = u(\log u - v) + \log u - v = u\log u - uv + \log u - v.$$

Therefore

$$f_*(v) = \log u - v = -\log(2 - \exp(v)).$$

□

Squared Hellinger divergence.

Proposition 5. The conjugate of $f(u) = (\sqrt{u} - 1)^2$ for $u \in \operatorname{dom} f = [0, +\infty)$ is $f_*(v) = \frac{v}{1-v}$ for $v \in \operatorname{dom} f_* = (-\infty, 1)$.

Proof. We have $uv - f(u) = u(v-1) + 2\sqrt{u} - 1$. Hence, for $v \geq 1$, $u \to uv - f(u)$ is unbounded above. For $v < 1$, $\lim_{u\to+\infty}\{uv - f(u)\} = -\infty$ and $uv - f(u)$ has then a maximizer $u^\star$ in $[0, +\infty)$ characterized by $f'(u^\star) = v$. We have $f(u) = (\sqrt{u} - 1)^2 = u - 2\sqrt{u} + 1$. Using

$$v = f'(u) = 1 - \frac{1}{\sqrt{u}} \iff \sqrt{u} = \frac{1}{1-v} \iff u = \frac{1}{(1-v)^2},$$

we get

$$\begin{aligned} uv &= \frac{v}{(1-v)^2} \\ \sqrt{u} - 1 &= \frac{v}{1-v} \\ (\sqrt{u} - 1)^2 &= \frac{v^2}{(1-v)^2} \end{aligned}$$

so that

$$f^*(v) = \frac{v}{(1-v)^2} - \frac{v^2}{(1-v)^2} = \frac{v}{1-v}.$$

□

Chi-squared divergence (Pearson divergence).

Proposition 6. The conjugate of $f(u) = \frac{1}{2}(u^2-1)$ for $u \in \operatorname{dom} f = \mathbb{R}$ is $f_*(v) = \frac{1}{2}(v^2+1)$ for $v \in \operatorname{dom} f_* = \mathbb{R}$.
对于 $u \in \operatorname{dom} f = \mathbb{R}$，$f(u) = \frac{1}{2}(u^2-1)$ 的共轭是对于 $v \in \operatorname{dom} f_* = \mathbb{R}$，$f_*(v) = \frac{1}{2}(v^2+1)$。

Proof. f is a strongly convex function on $\mathbb{R}$. Hence the maximizer $u^\star$ of $u \mapsto uv - f(u)$ exists for any $v \in \mathbb{R}$ and is characterized by $f'(u^\star) = v$. Using $f'(u) = u$, we get $v = f'(u) = u$. Therefore,
f 是 $\mathbb{R}$ 上的强凸函数。因此，对于任何 $v \in \mathbb{R}$，$u \mapsto uv - f(u)$ 的最大化函数 $u^\star$ 都存在并且以 $f'(u^\star) = v$ 为特征。使用 $f'(u) = u$，我们得到 $v = f'(u) = u$。因此，

$$f_*(v) = uv - f(u) = v^2 - \frac{1}{2}(v^2-1) = \frac{1}{2}(v^2+1).$$

□

Reverse chi-squared divergence (Neyman divergence).

Proposition 7. The conjugate of $f(u) = \frac{1}{2}(\frac{1}{u} - 1)$ for $u \in \operatorname{dom} f = (0, +\infty)$ is $f_*(v) = -\sqrt{-2v} + \frac{1}{2}$ for $v \in \operatorname{dom} f_* = (-\infty, 0)$.

Proof. For $v > 0$, $\lim_{u\to+\infty}\{uv - f(u)\} = +\infty$. For $v < 0$, $\lim_{u\to 0}\{uv - f(u)\} = \lim_{u\to+\infty}\{uv - f(u)\} = -\infty$. Hence the maximizer $u^\star$ of $u \mapsto uv - f(u)$ exists and is characterized by $f'(u^\star) = v$. Using $v = f'(u) = -\frac{1}{2u^2}$, we get

$$u^2 = -\frac{1}{2v} \iff u = \frac{1}{\sqrt{-2v}}.$$

Therefore,

$$f_*(v) = uv - \frac{1}{2}\left(\frac{1}{u} - 1\right) = \frac{v}{\sqrt{-2v}} - \frac{\sqrt{-2v}}{2} + \frac{1}{2} = -\sqrt{-2v} + \frac{1}{2}$$

□

α-divergences.

Proposition 8. The conjugate of $f(u) = \frac{(u^\alpha-1)-\alpha(u-1)}{\alpha(\alpha-1)} + \iota_{\mathbb{R}_+}(u)$ is

$$f_*(v) = \begin{cases} \frac{(1+(\alpha-1)v)^{\frac{\alpha}{\alpha-1}}-1}{\alpha} & \text{if } \alpha \in (0,1) \\ \exp(v) & \text{if } \alpha = 1 \\ \frac{(1+(\alpha-1)v)_+^{\frac{\alpha}{\alpha-1}}-1}{\alpha} & \text{if } \alpha \in (1,+\infty), \text{ where } x_+ = \max(x,0) \end{cases}$$

for

$$v \in \operatorname{dom} f^* = \begin{cases} (-\infty, -\frac{1}{\alpha-1}] & \text{if } \alpha \in (0,1) \\ \mathbb{R} & \text{if } \alpha \geq 1 \end{cases}$$

Proof. Case $\alpha \in (0,1)$. As $u \to +\infty$, we have $vu - \frac{(u^\alpha-1)-\alpha(u-1)}{\alpha(\alpha-1)} \sim \left(v + \frac{1}{\alpha-1}\right)u$.

Hence for $v > \frac{1}{1-\alpha}$, $\lim_{u\to+\infty} vu - \frac{(u^\alpha-1)-\alpha(u-1)}{\alpha(\alpha-1)} = +\infty$.

For $v \leq \frac{1}{1-\alpha}$, $\lim_{u\to+\infty}\{vu - \frac{(u^\alpha-1)-\alpha(u-1)}{\alpha(\alpha-1)}\} = \lim_{u\to 0}\{vu - \frac{(u^\alpha-1)-\alpha(u-1)}{\alpha(\alpha-1)}\} = -\infty$. Hence the maximizer $u^\star$ of $u \mapsto uv - f(u)$ exists and is characterized by $f'(u^\star) = v$.

We have

$$v = f'(u) = \frac{u^{\alpha-1}-1}{\alpha-1} \iff u = (1+(\alpha-1)v)^{\frac{1}{\alpha-1}} = f_*'(v)$$

Therefore, denoting $z = (1 + (\alpha - 1)v)$, we have, for $v \leq \frac{1}{1-\alpha}$

$$\begin{aligned} f^*(v) &= uv - f(u) \\ &= vz^{\frac{1}{\alpha-1}} - \frac{(z^{\frac{\alpha}{\alpha-1}} - 1) - \alpha(z^{\frac{1}{\alpha-1}} - 1)}{\alpha(\alpha-1)} \\ &= \frac{\alpha((\alpha-1)v+1)z^{\frac{1}{\alpha-1}} - z^{\frac{\alpha}{\alpha-1}} - (\alpha-1)}{\alpha(\alpha-1)} \\ &= \frac{z^{\frac{\alpha}{\alpha-1}} - 1}{\alpha} \\ &= \frac{(1+(\alpha-1)v)^{\frac{\alpha}{\alpha-1}} - 1}{\alpha}. \end{aligned}$$

Case $\alpha \in (1, +\infty)$. For any $v \in \mathbb{R}$, $\lim_{u\to+\infty} vu - \frac{(u^\alpha-1)-\alpha(u-1)}{\alpha(\alpha-1)} = -\infty$. Denote $g(u) = vu - \frac{(u^\alpha-1)-\alpha(u-1)}{\alpha(\alpha-1)}$. Note that g' is decreasing on $[0, +\infty]$.

If $v \leq \frac{1}{1-\alpha}$, we have $g'(0) = v + \frac{1}{\alpha-1} \leq 0$, hence $g'(u) < 0$ for any $u > 0$, that is g is decreasing on $[0, +\infty)$ and $\sup_{u\geq 0} g(u) = g(0) = -f(0)$ in this case.

If $v > \frac{1}{1-\alpha}$, $g'(0) > 0$, $\lim_{u\to+\infty} g'(u) = -\infty$ and a maximizer $u^\star$ of g can be found by the necessary condition $f'(u^\star) = v$. Following the computations done in the case $\alpha \in (0, 1)$, for $v > \frac{1}{1-\alpha}$ we get

$$\begin{aligned} f^*(v) &= \frac{(1+(\alpha-1)v)^{\frac{\alpha}{\alpha-1}}}{\alpha} \\ f'_*(v) &= (1+(\alpha-1)v)^{\frac{1}{\alpha-1}}. \end{aligned}$$

And for any $v \in \mathbb{R}$, we get

$$\begin{aligned} f^*(v) &= \frac{(1+(\alpha-1)v)_+^{\frac{\alpha}{\alpha-1}} - 1}{\alpha} \\ f'_*(v) &= (1+(\alpha-1)v)_+^{\frac{1}{\alpha-1}}. \end{aligned}$$

where $x_+ = \max\{x, 0\}$.

Case $\alpha = 1$ Follows similar computations as for the KL case. □

C.2. f-softargmaxes

KL divergence. With $f_*(v) = f'_*(v) = \exp(v-1)$, we obtain

$$\tau^\star + 1 = \log \sum_{j=1}^k q_j \exp(\theta_j).$$

Plugging $\tau^\star$ back in (13), we obtain

$$p_j^\star = \frac{q_j \exp(\theta_j)}{\sum_{j=1}^k q_j \exp(\theta_j)},$$

Other choices of f do not lead to closed-form expressions.

C.3. f-softplus and f-sigmoid

In this section, we derive closed-form expressions for some instances of f-softplus and f-sigmoid. In this binary classification setting, we define $\boldsymbol{\theta} \coloneqq (0, s)$, $\boldsymbol{q} = (q_0, q_1)$ and $\boldsymbol{p}^\star = (1 - \pi^\star, \pi^\star)$. By Proposition 1, we seek for

$$\tau^\star = \operatorname*{argmin}_{\tau\in\mathbb{R}} \tau + q_0 f'_*(-\tau) + q_1 f'_*(s - \tau).$$

Since f_* differentiable and convex, such a scalar $\tau^\star$ is characterized by

$$-\tau^\star \in \operatorname{dom} f_*, \tag{17}$$
$$s - \tau^\star \in \operatorname{dom} f_*, \tag{18}$$
$$q_0 f_*'(-\tau^\star) + q_1 f_*'(s - \tau^\star) = 1. \tag{19}$$

KL divergence.

Proposition 9. Let $f(u) = u \log u$. Then,

$$\operatorname{sigmoid}_f(s; \boldsymbol{q}) = \frac{q_1 \exp(s)}{q_0 + q_1 \exp(s)}.$$

Proof. Using (19) and $f_*'(v) = \exp(v - 1)$, we obtain

$$\tau^\star + 1 = \log(q_0 + q_1 \exp(s)).$$

so that

$$\pi^\star = \frac{q_1 \exp(s)}{q_0 + q_1 \exp(s)}.$$

□

Reverse KL divergence.

Proposition 10. Let $f(u) = -\log(u)$. Then,

$$\operatorname{softplus}_f(s; \boldsymbol{q}) = \tau^\star - q_0 \log(\tau^\star) - q_1 \log(\tau^\star - s) - (q_0 + q_1)$$
$$\operatorname{sigmoid}_f(s; \boldsymbol{q}) = \frac{q_1}{\tau^\star - s}$$

where

$$\tau^\star \coloneqq \frac{1}{2}\left(q_0 + q_1 + s + \sqrt{(q_0 + q_1 + s)^2 - 4 q_0 s}\right).$$

Proof. Using $f_*'(v) = -1/v$, the root condition (19) can be written as

$$\begin{aligned}
& q_0 f_*'(-\tau) + q_1 f_*'(s - \tau) = 1 \\
\iff & \frac{q_0}{\tau} + \frac{q_1}{\tau - s} = 1 \\
\iff & \frac{q_0(\tau - s) + q_1 \tau}{\tau(\tau - s)} = 1 \\
\iff & a\tau^2 + b\tau + c = 0,
\end{aligned}$$

for

$$\begin{aligned}
a &\coloneqq 1 \\
b &\coloneqq -(q_0 + q_1 + s) \\
c &\coloneqq q_0 s.
\end{aligned}$$

Let us define the discrimant

$$\begin{aligned}
\Delta &\coloneqq b^2 - 4ac \\
&= (q_0 + q_1 + s)^2 - 4 q_0 s \\
&= (q_1 - q_0 + s)^2 + 4 q_0 q_1 \\
&> 0.
\end{aligned}$$

Therefore, the root condition (19) reads

$$\begin{aligned}\tau^\star &\in \{\tau_1, \tau_2\} \\ &\coloneqq \left\{\frac{-b+\sqrt{\Delta}}{2a}, \frac{-b-\sqrt{\Delta}}{2a}\right\}.\end{aligned}$$

Since $\operatorname{dom} f_* = \mathbb{R}_-$, we have

$$(-\tau^\star \in \operatorname{dom} f_*) \text{ and } (s - \tau^\star \in \operatorname{dom} f_*) \iff \tau^\star \geq \max\{0, s\}.$$

We have

$$\begin{aligned}\tau_2 &= \frac{1}{2}\left(q_0 + q_1 + s - \sqrt{(q_0+q_1+s)^2 - 4q_0 s}\right) \\ \tau_2 - s &= \frac{1}{2}\left(q_0 + q_1 - s - \sqrt{(q_0+q_1-s)^2 + 4q_1 s}\right).\end{aligned}$$

Hence for $s < 0$, $\tau_2 < 0$, and for $s \geq 0$, $\tau_2 - s < 0$. So the unique solution to the set of conditions (17), (18) and (19) is

$$\tau^\star = \tau_1 = \frac{1}{2}\left(q_0 + q_1 + s + \sqrt{(q_0+q_1+s)^2 - 4q_0 s}\right)$$

Plugging this value in the formulas for the softplus and the sigmoid in Proposition 1 gives the result. □

Numerically stable implementation.

For very large positive or negative s we may get numerical issues of the form $as - bs$. For a numerically stable implementation, one could write
对于非常大的正或负 s，我们可能会得到形式为 $as - bs$ 的数值问题。对于数值稳定的实现，可以写成

$$\begin{aligned}\tau^\star &= q_0 + h(g(s)) \\ \tau^\star - s &= q_1 + h(-g(s))\end{aligned}$$

where

$$\begin{aligned}g(s) &\coloneqq q_1 - q_0 + s \\ h(g) &\coloneqq \frac{1}{2} \cdot \begin{cases} |g|\left(\sqrt{1 + \frac{4q_0 q_1}{g^2}} - 1\right) & \text{if } g < -1 \\ g + \sqrt{g^2 + 4q_0 q_1} & \text{otherwise} \end{cases}\end{aligned}$$

We can then further rewrite

$$\begin{aligned}|g|\left(\sqrt{1 + \frac{4q_0 q_1}{g^2}} - 1\right) &= |g| \frac{4q_0 q_1 / g^2}{\sqrt{1 + \frac{4q_0 q_1}{g^2}} + 1} \\ &= \frac{4q_0 q_1}{\sqrt{g^2 + 4q_0 q_1} + |g|}.\end{aligned}$$

Jensen-Shannon divergence.

Proposition 11. Let $f(u) = u \log u - (u+1)\log\left(\frac{u+1}{2}\right)$. Then,

$$\begin{aligned}\operatorname{softplus}_f(s; \boldsymbol{q}) &= \log(x^\star) - q_0 \log(2 - 1/x^\star) - q_1 \log(2 - 1/(x^\star \cdot y)) \\ \operatorname{sigmoid}_f(s; \boldsymbol{q}) &= \frac{q_1}{2 \cdot x^\star \cdot y - 1},\end{aligned}$$

where

$$
\begin{aligned}
x^\star &\coloneqq \frac{-b+\sqrt{\Delta}}{2a} \\
y &\coloneqq \exp(-s) \\
a &\coloneqq 4y \\
b &\coloneqq -2(1+y+yq_0+q_1) \\
c &\coloneqq 1+q_0+q_1 \\
\Delta &\coloneqq b^2-4ac.
\end{aligned}
$$

Proof. Using $f_*'(v) = \frac{1}{2\exp(-v)-1}$, the root equation (19)

$$\frac{q_0}{2\exp(\tau)-1} + \frac{q_1}{2\exp(\tau-s)-1} = 1.$$

Using the change of variables $x \coloneqq \exp(\tau)$ and $y \coloneqq \exp(-s)$, we obtain

$$
\begin{aligned}
&\frac{q_0}{2x-1} + \frac{q_1}{2xy-1} = 1 \\
\iff & q_0(2xy-1) + q_1(2x-1) = (2x-1)(2xy-1) \\
\iff & ax^2+bx+c = 0,
\end{aligned}
$$

where we defined

$$
\begin{aligned}
a &\coloneqq 4y \\
b &\coloneqq -2(1+y+yq_0+q_1) \\
c &\coloneqq 1+q_0+q_1.
\end{aligned}
$$

Let us define the discriminant

$$\Delta \coloneqq b^2-4ac.$$

We have

$$
\begin{aligned}
\Delta &= 4y\left[\left((1+q_0)y^{1/2}+(1+q_1)y^{-1/2}\right)^2 - 4(1+q_0+q_1)\right] \\
&\geq 4y\left[4(1+q_0)(1+q_1) - 4(1+q_0+q_1)\right] \\
&= 16yq_0q_1 \\
&> 0,
\end{aligned}
$$

where in the second line we used that $y > 0$ and $\min_{x>0} \alpha x + \beta x^{-1} = 2\sqrt{\alpha\beta}$ for any $\alpha > 0, \beta > 0$. Therefore, we have

$$x^\star = \exp(\tau^\star) = \frac{-b+\sqrt{\Delta}}{2a}.$$

Using (12) with

$$
\begin{aligned}
f_*(-\tau^\star) &= -\log(2-\exp(-\tau^\star)) = -\log(2-1/x^\star) \\
f_*(s-\tau^\star) &= -\log(2-\exp(s-\tau^\star)) = -\log(2-1/(x^\star\cdot y)),
\end{aligned}
$$

we obtain

$$\operatorname{softplus}_f(s) = \log(x^\star) - q_0\log(2-1/x^\star) - q_1\log(2-1/(x^\star\cdot y)).$$

Similarly, using (13), we obtain

$$\operatorname{sigmoid}_f(s) = \frac{q_1}{2\exp(\tau^\star-s)-1} = \frac{q_1}{2\cdot x^\star\cdot y-1}.$$

□

Numerically stable implementation.

Working with exponential requires careful handling of any formula like $uy - vy$ or $(uy)/(vy)$ that can easily reduce numerically to $\infty - \infty = \text{NaN}$ or $\infty/\infty = \text{NaN}$. We therefore derive the expressions of $x^\star$ and $x^\star y$ in terms of numerically stable operations.
使用指数需要小心处理任何公式，如 $uy - vy$ 或 $(uy)/(vy)$ 这些公式可以很容易地在数值上简化为 $\infty - \infty = \text{NaN}$ 或 $\infty/\infty = \text{NaN}$。因此，我们根据数值稳定的运算推导出 $x^\star$ 和 $x^\star y$ 的表达式。

We have, denoting $\alpha \coloneqq 1 + q_0$ and $\beta \coloneqq 1 + q_1$,

$$\begin{aligned}
x^\star &= \frac{\alpha + \beta y^{-1}}{4} + \frac{1}{4}\sqrt{(\alpha + \beta y^{-1})^2 - 4cy^{-1}} \\
&= \frac{y^{-1/2}}{4}\left(\alpha y^{1/2} + \beta y^{-1/2} + \sqrt{(\alpha y^{1/2} + \beta y^{-1/2})^2 - 4c}\right) \\
&= \frac{y^{-1/2}}{4}(\alpha y^{1/2} + \beta y^{-1/2})\left(1 + \sqrt{1 - \frac{4(1 + q_0 + q_1)}{(\alpha y^{1/2} + \beta y^{-1/2})^2}}\right) \\
&= \exp(s/2 + h(s/2)),
\end{aligned}$$

where

$$\begin{aligned}
h(s) &\coloneqq g(s) + \log\left(1 + \sqrt{1 - \frac{4(1 + q_0 + q_1)}{\exp(2g(s))}}\right) - 2\log(2) \\
g(s) &\coloneqq \log\left(\alpha e^{-s} + \beta e^{s}\right) \\
&\quad = \text{logsumexp}(-s + \log(1 + q_0), s + \log(1 + q_1)).
\end{aligned}$$

To summarize, we have

$$\begin{aligned}
x^\star &= \exp(s/2 + h(s/2)) \\
x^\star y &= \exp(-s/2 + h(s/2))
\end{aligned}$$

and therefore

$$\begin{aligned}
\text{softplus}_f(s) &= s/2 + h(s/2) - q_0 \log\left[2 - \exp(-s/2 - h(s/2))\right] - q_1 \log\left[2 - \exp(s/2 - h(s/2))\right] \\
\text{sigmoid}_f(s) &= \frac{q_1}{2\exp(-s/2 + h(s/2)) - 1}.
\end{aligned}$$

Squared Hellinger divergence. Using $f_*'(v) = \frac{1}{(1-v)^2}$ and using the change of variable $x \coloneqq \tau + 1$, the root equation (19) becomes
使用 $f_*'(v) = \frac{1}{(1-v)^2}$ 并使用变量变换 $x \coloneqq \tau + 1$，根方程 (19) 变为

$$\begin{aligned}
\frac{q_0}{(1+\tau)^2} + \frac{q_1}{(1+\tau-s)^2} = 1 &\iff \frac{q_0}{x^2} + \frac{q_1}{(x-s)^2} = 1 \\
&\iff q_0(x-s)^2 + q_1 x^2 = x^2(x-s)^2 \\
&\iff q_0(x^2 - 2sx + s^2) + q_1 x^2 = x^2(x^2 - 2sx + s^2) \\
&\iff q_0(x^2 - 2sx + s^2) + q_1 x^2 = x^4 - 2sx^3 + s^2x^2 \\
&\iff ax^4 + bx^3 + cx^2 + dx + e = 0,
\end{aligned}$$

where

$$\begin{aligned}
a &\coloneqq 1 \\
b &\coloneqq -2s \\
c &\coloneqq s^2 - q0 - q1 \\
d &\coloneqq 2sq0 \\
e &\coloneqq -q0s^2.
\end{aligned}$$

This is a quartic equation, which can be solved in closed form.

C.4. Proof of Proposition 1

First, the maximum defining the softmax and softargmax is well defined since it is a strictly concave problem on a non-empty bounded set. Denoting $\Omega_j(p) \coloneqq q_j f(p/q_j)$, such that $\Omega_j'(p) = f'(p/q_j)$ and $\Omega_j^*(\theta) = q_j f^*(\theta)$, we can apply Lemma 1 and get
首先，定义 softmax 和 softargmax 的最大值定义明确，因为它是非空有界集上的严格凹问题。表示 $\Omega_j(p) \coloneqq q_j f(p/q_j)$，使得 $\Omega_j'(p) = f'(p/q_j)$ 和 $\Omega_j^*(\theta) = q_j f^*(\theta)$，我们可以应用引理 1 并得到

$$\max_{\boldsymbol{p}\in\triangle^k} \langle \boldsymbol{p}, \boldsymbol{\theta}\rangle - \sum_{j=1}^k q_j f(p_j/q_j) = \inf_{\tau\in\mathbb{R}} \tau + \sum_{j=1}^k q_j f^*(\max\{\theta_j - \tau, f'(0)\}), \tag{20}$$

with $f'(0) \coloneqq \lim_{x\to 0, x\geq 0} f'(x) \in \mathbb{R}\cup\{-\infty\}$.

Since $(0,+\infty) \subseteq \operatorname{dom} f'$, and $\boldsymbol{q} > 0$, $f'\left(\left(\sum_{j=1}^k q_j\right)^{-1}\right)$ and $f'(q_{j_{\max}}^{-1})$ for $j_{\max} \in \operatorname{argmax}_{j\in\{1,\dots,k\}} \theta_j$ are well defined. We can then define

$$\begin{aligned}\tau_{\min} &\coloneqq \theta_{\max} - f'(q_{j_{\max}}^{-1})\\ \tau_{\max} &\coloneqq \theta_{\max} - f'\left(\left(\textstyle\sum_{j=1}^k q_j\right)^{-1}\right)\end{aligned}$$

where $\theta_{\max} = \theta_{j_{\max}}$. Since $\boldsymbol{q} > 0$, $q_{j_{\max}} < \sum_{j=1}^k q_j$, and since f' is increasing, we have $\tau_{\min} < \tau_{\max}$.

We can then analyze the following function on $[\tau_{\min}, \tau_{\max}]$:

$$h(\tau) \coloneqq \tau + \sum_{j=1}^k q_j f^*(\max\{\theta_j - \tau, f'(0)\}).$$

First, we need to ensure that we can compute derivatives of this function in $[\tau_{\min}, \tau_{\max}]$. From Lemma 2, we have $f^*(\max\{\theta_j - \tau, f'(0)\}) = (f + \iota_{\mathbb{R}_+})^*(\theta_j - \tau)$ and $\operatorname{dom}(f+\iota_{\mathbb{R}_+})_*' = \operatorname{dom} f_*' \cup (-\infty, f'(0)]$. Therefore, if $f'(0) > -\infty$, the domain of $(f+\iota_{\mathbb{R}_+})_*'$ is unbounded below. Otherwise, if $f'(0) = -\infty$, since $\operatorname{im} f' \subseteq \operatorname{dom} f_*'$, the domain of f_*', and so of $(f+\iota_{\mathbb{R}_+})_*'$, are unbounded below. Denoting $\alpha \coloneqq \sup \operatorname{dom}(f+\iota_{\mathbb{R}_+})_*'$, since $\theta_{\max} - \tau_{\min} = f'(q_{j_{\max}}^{-1}) \in \operatorname{dom} f_*'$, we then have $\theta_{\max} - \tau_{\min} < \alpha$ and therefore
首先，我们需要确保能够计算该函数在 $[\tau_{\min}, \tau_{\max}]$ 中的导数。从引理 2 可知 $f^*(\max\{\theta_j - \tau, f'(0)\}) = (f + \iota_{\mathbb{R}_+})^*(\theta_j - \tau)$ 和 $\operatorname{dom}(f+\iota_{\mathbb{R}_+})_*' = \operatorname{dom} f_*' \cup (-\infty, f'(0)]$。因此，如果 $f'(0) > -\infty$，则 $(f+\iota_{\mathbb{R}_+})_*'$ 的定义域在下方无界。否则，如果 $f'(0) = -\infty$，由于 $\operatorname{im} f' \subseteq \operatorname{dom} f_*'$，$f_*'$ 的定义域，因此 $(f+\iota_{\mathbb{R}_+})_*'$，在下方无界。记 $\alpha \coloneqq \sup \operatorname{dom}(f+\iota_{\mathbb{R}_+})_*'$，因为 $\theta_{\max} - \tau_{\min} = f'(q_{j_{\max}}^{-1}) \in \operatorname{dom} f_*'$，然后我们有 $\theta_{\max} - \tau_{\min} < \alpha$，因此

$$\begin{aligned}&\tau \geq \tau_{\min}\\ \implies &\theta_{\max} - \tau < \alpha\\ \iff &\theta_j - \tau < \alpha, \text{for all } j \in \{1,\dots,k\}\\ \implies &\tau \in \operatorname{dom} h'.\end{aligned}$$

We can then show that $h'(\tau_{\min}) \leq 0$ and $h'(\tau_{\max}) \geq 0$. Indeed, we have

$$\begin{aligned}h'(\tau_{\min}) &= 1 - \sum_{j=1}^k q_j f_*'(\max\{\theta_j - \tau_{\min}, f'(0)\})\\ &\overset{(i)}{\leq} 1 - q_{j_{\max}} f_*'(\max\{\theta_{\max} - \tau_{\min}, f'(0)\})\\ &= 1 - q_{j_{\max}} f_*'(\max\{f'(q_{j_{\max}}^{-1}), f'(0)\})\\ &\overset{(ii)}{=} 0,\end{aligned}$$

where in (i) we used that $\boldsymbol{q} > 0$ and $f'_*(\max\{y, f'(0)\}) \geq 0$ for any $y \in \operatorname{dom}(f + \iota_{\mathbb{R}_+})'_*$ as per Lemma 2, and in (ii), we used that $q_{j_{\max}}^{-1} > 0$, f' is increasing and $f'_*(f'(p)) = p$ for any $p \in \operatorname{dom} f'$.
其中在 (i) 中我们利用了 $\boldsymbol{q} > 0$ 且对于任何 $y \in \operatorname{dom}(f + \iota_{\mathbb{R}_+})'_*$，$f'_*(\max\{y, f'(0)\}) \geq 0$，正如引理 2 所言；在 (ii) 中，我们利用了 $q_{j_{\max}}^{-1} > 0$，f' 是递增的，且对于任何 $p \in \operatorname{dom} f'$，$f'_*(f'(p)) = p$。

Similarly, we have

$$\begin{aligned} h'(\tau_{\max}) &= 1 - \textstyle\sum_{j=1}^k q_j f'_*(\max\{\theta_j - \tau_{\max}, f'(0)\}) \\ &\overset{(i)}{\geq} 1 - \left(\textstyle\sum_{j=1}^k q_j\right) f'_*(\max\{\theta_{\max} - \tau_{\max}, f'(0)\}) \\ &= 1 - \left(\textstyle\sum_{j=1}^k q_j\right) f'_* \left(\max\left\{f'\left(\left(\textstyle\sum_{j=1}^k q_j\right)^{-1}\right), f'(0)\right\}\right) \\ &\overset{(ii)}{=} 0, \end{aligned}$$

where in (i) we used that $\sum_j a_j b_j \leq (\sum_j a_j) \max_j b_j$ if $a_j \geq 0$ with here $a_j = q_j > 0$ and $b_j = f'_*(\max\{\theta_j - \tau_{\max}, f'(0)\})$, and in (ii) we used the same reasoning as for $h'(\tau_{\min})$.
其中在 (i) 中我们使用了 $\sum_j a_j b_j \leq (\sum_j a_j) \max_j b_j$ 如果 $a_j \geq 0$，其中 $a_j = q_j > 0$ 且 $b_j = f'_*(\max\{\theta_j - \tau_{\max}, f'(0)\})$，而在 (ii) 中我们使用了与 $h'(\tau_{\min})$ 相同的推理。

Finally, we show that h' is increasing on $[\tau_{\min}, \tau_{\max}]$. For $j = j_{\max}$, we have that

$$\begin{aligned} &\tau \leq \tau_{\max} \\ \iff &\theta_{\max} - \tau \geq \theta_{\max} - \tau_{\max} = f'(q_{j_{\max}}) > f'(0). \end{aligned}$$

Since $f'_*(\max\{\theta, f'(0)\}) = (f + \iota_{\mathbb{R}_+})^*$ is increasing on $\operatorname{dom} f^* \setminus (-\infty, f(0)]$ (see Proposition 2), $\tau \mapsto -f'_*(\max\{\theta_{\max} - \tau, f'(0)\})$ is increasing on $[\tau_{\min}, \tau_{\max}]$ and so $h'(\tau) = 1 - \sum_{j=1}^k q_j f'_*(\max\{\theta_j - \tau, f'(0)\})$ is increasing on $[\tau_{\min}, \tau_{\max}]$ (as a sum of an increasing and non decreasing function).

Overall, h is well defined and strictly convex on $[\tau_{\min}, \tau_{\max}]$ such that $h'(\tau_{\min}) \leq 0$ and $h'(\tau_{\max}) \geq 0$. Hence, we have
总体而言，h 定义明确，且在 $[\tau_{\min}, \tau_{\max}]$ 上严格凸，使得 $h'(\tau_{\min}) \leq 0$ 和 $h'(\tau_{\max}) \geq 0$。因此，我们有

$$\inf_{\tau \in \mathbb{R}} h(\tau) = \min_{\tau_{\min} \leq \tau \leq \tau_{\max}} h(\tau).$$

and the unique minimizer $\tau^\star$ can then be found by solving the first order optimality condition $h'(\tau^\star) = 0$ in $[\tau_{\min}, \tau_{\max}]$. This gives the expression of $\tau^\star$ in the claim and plugging $\tau^\star$ back into (20) gives the expression of the softmax. The expression of the softargmax follows from Lemma 1.
然后可以通过求解 $[\tau_{\min}, \tau_{\max}]$ 中的一阶最优条件 $h'(\tau^\star) = 0$ 来找到唯一最小化器 $\tau^\star$。这给出了声明中的 $\tau^\star$ 的表达式，将 $\tau^\star$ 代回 (20) 可得到 softmax 的表达式。softargmax 的表达式遵循引理 1。

C.5. Lemmas

Lemma 1. Given k strictly convex differentiable univariate scalar functions Ω_j such that $(0, +\infty) \subseteq \operatorname{dom} \Omega_j$, for any $\boldsymbol{\theta} \in \mathbb{R}^k$, we have

$$\sup_{\boldsymbol{p} \in \triangle^k} \langle \boldsymbol{p}, \boldsymbol{\theta} \rangle - \sum_{j=1}^k \Omega_j(p_j) = \inf_{\tau \in \mathbb{R}} \tau + \sum_{j=1}^k (\Omega_j + \iota_{\mathbb{R}_+})^*(\theta_j - \tau),$$

where

$$(\Omega_j + \iota_{\mathbb{R}_+})^*(z) = \Omega_j^*(\max\{\theta_j - \tau, \Omega'_j(0)\})$$

and $\Omega'_j(0) \coloneqq \lim_{x \to 0, x \geq 0} \Omega'_j(x) \in \mathbb{R} \cup \{-\infty\}$. Given a minimizer $\tau^\star$ of the right hand-side, the maximizer $\boldsymbol{p}^\star$ of the left hand side is given by

$$p_j^\star = (\Omega_j + \iota_{\mathbb{R}_+})'_*(\theta_j - \tau^\star) = (\Omega_j^*)'(\max\{\theta_j - \tau^\star, \Omega'_j(0)\}).$$

Proof. We have

$$\begin{aligned}\sup_{\boldsymbol{p}\in\triangle^k}\langle \boldsymbol{p},\boldsymbol{\theta}\rangle-\sum_{j=1}^k\Omega_j(p_j) &= \sup_{\boldsymbol{p}\in\mathbb{R}_+^k}\inf_{\tau\in\mathbb{R}}\langle \boldsymbol{p},\boldsymbol{\theta}\rangle-\sum_{j=1}^k\Omega_j(p_j)+(1-\langle \boldsymbol{p},\mathbf{1}\rangle)\tau\\ &= \inf_{\tau\in\mathbb{R}}\tau+\max_{\boldsymbol{p}\in\mathbb{R}_+^k}\langle \boldsymbol{p},\boldsymbol{\theta}-\tau\mathbf{1}\rangle-\sum_{j=1}^k\Omega_j(p_j)\\ &= \inf_{\tau\in\mathbb{R}}\tau+\sum_{j=1}^k(\Omega_j+\iota_{\mathbb{R}_+})^*(\theta_j-\tau),\end{aligned}$$

where the second equality stands from strong duality, using that the simplex is convex, that the Ω_j are convex functions, and that the maximization problem is strictly feasible since $(0,+\infty)\subseteq\operatorname{dom}\Omega_j$.
其中第二个等式来自强对偶性，使用单纯形是凸的，Ω_j 是凸函数，并且最大化问题是严格可行的，因为 $(0,+\infty)\subseteq\operatorname{dom}\Omega_j$。

We can then apply Lemma 2, and get

$$\max_{\boldsymbol{p}\in\triangle^k}\langle \boldsymbol{p},\boldsymbol{\theta}\rangle-\sum_{j=1}^k\Omega_j(p_j)=\inf_{\tau\in\mathbb{R}}\tau+\sum_{j=1}^k\Omega_j^*(\max\{\theta_j-\tau,\Omega_j'(0)\}),$$

with $\Omega_j'(0)=\lim_{x\to0,x\geq0}\Omega_j'(x)\in\mathbb{R}\cup\{-\infty\}$. Given a minimizer $\tau^\star$ of the right hand-side, the maximizer $\boldsymbol{p}^\star$ of the left hand side is given by strong duality as

$$p_j^\star=\operatorname*{argmax}_{p_j\geq0}p_j(\theta_j-\tau)-\Omega_j(p_j),$$

whose full expression follows from Lemma 2. □

> Lemma 2. Given a strictly convex differentiable univariate scalar function f such that $(0,+\infty)\subseteq\operatorname{dom}f$, we have, for any $y\in\mathbb{R}$,
>
> $$(f+\iota_{\mathbb{R}_+})^*(y)\coloneqq\sup_{x\geq0}xy-f(x)=f^*(\max\{y,f'(0)\})$$
>
> with $f'(0)\coloneqq\lim_{x\to0,x\geq0}f'(x)\in\mathbb{R}\cup\{-\infty\}$. For $y\in\operatorname{dom}(f+\iota_{\mathbb{R}_+})_*'=\operatorname{dom}f_*'\cup(-\infty,f'(0)]$, we have
>
> $$(f+\iota_{\mathbb{R}_+})_*'(y)=f_*'(\max\{y,f'(0)\})\geq0.$$
>
> Finally, $(f+\iota_{\mathbb{R}_+})_*'$ is increasing on $(\operatorname{dom}f+\iota_{\mathbb{R}_+})_*'\setminus(-\infty,f'(0)]$.

Proof. Fix some $y\in\mathbb{R}$, denote $f'(0)=\lim_{x\to0,x\geq0}f'(x)\in\mathbb{R}\cup\{-\infty\}$ and

$$h(x)\coloneqq xy-f(x).$$

If $y\leq f'(0)$ (so provided that $f'(0)>-\infty$), then $h'(x)=y-f'(x)>0$ for all $x>0$, since f' is increasing. So h is decreasing on $(0,+\infty)$ and the maximum of h on $\mathbb{R}_+$ is reached at 0, that is

$$\sup_{x\geq0}xy-f(x)=-f(0).$$

If $y\geq f'(0)$, then $h'(x)=y-f'(x)>0$ for all $x<0$ since f' is increasing. Therefore $h(x)<h(0)$ for all $x<0$. Finally, since $h'(0)\leq0$, the supremum of h on $\mathbb{R}$ is necessarily greater or equal than 0, that is,

$$\sup_{x\geq0}xy-f(x)=\sup_{x\in\mathbb{R}}xy-f(x)=f^*(y).$$

Note that for $y = f'(0)$ (provided that $f'(0) > -\infty$), we then get

$$\sup_{x \geq 0} xf'(0) - f(x) = -f(0) = f^*(f'(0)),$$

and

$$f'_*(f'(0)) = \operatorname*{argmax}_{x \in \mathbb{R}} \{xf'(0) - f(x)\} = 0.$$

Combining the two cases above, we get

$$\begin{aligned} \sup_{x \geq 0} xy - f(x) &= \begin{cases} f^*(y) & \text{if } y > f'(0) \\ -f(0) & \text{if } y \leq f'(0) \end{cases} \\ &= f^*(\max\{y, f'(0)\}). \end{aligned}$$

For the derivative, first note that since $f + \iota_{\mathbb{R}_+}$ is strictly convex, its convex conjugate is differentiable on its domain of definition. Then, given $y \in \operatorname{dom}(f + \iota_{\mathbb{R}_+})'_*$, we have from the previous considerations,

$$(f + \iota_{\mathbb{R}_+})'_*(y) = \operatorname*{argmax}_{x \geq 0} \{xy - f(x)\} = \begin{cases} f'_*(y) & \text{if } y > f'(0) \\ 0 & \text{if } y \leq f'(0) \end{cases}.$$

Using that $f'_*(f'(0)) = 0$, we get

$$(f + \iota_{\mathbb{R}_+})'_*(y) = f'_*(\max\{y, f'(0)\}).$$

The expressions above also show that $\operatorname{dom}(f + \iota_{\mathbb{R}_+})'_* = \operatorname{dom} f'_* \cup (-\infty, f'(0)]$.

In any case, for $y \in \operatorname{dom}((f + \iota_{\mathbb{R}_+})^*)'$,

$$(f + \iota_{\mathbb{R}_+})'_*(y) = \operatorname*{argmax}_{x \geq 0} xy - f(x) \geq 0,$$

by definition of the maximization set.

Finally, since f is strictly convex and differentiable on $(0, +\infty)$, f' is invertible on $(f'(0), f'(\infty))$ where $f'(0) = \lim_{x \to 0, x \geq 0} f'(x)$ and $f'(\infty) = \lim_{x \to +\infty} f'(x)$. Moreover, for any $y \in (f'(0), f'(\infty))$, we then have $(f + \iota_{\mathbb{R}_+})'_*(y) = f'_*(y) = (f')^{-1}(y)$, where $(f')^{-1}$ is the inverse of f' on $(f'(0), f'(\infty))$. Since f' is increasing on $(0, +\infty)$, $(f')^{-1}$ is also increasing on $(f'(0), f'(\infty))$, and so are f'_* and $(f + \iota_{\mathbb{R}_+})'_*$. Noting that $(f'(0), f'(\infty)) = \operatorname{dom} f'_* \setminus (-\infty, f'(0)] = (\operatorname{dom} f + \iota_{\mathbb{R}_+})'_* \setminus (-\infty, f'(0)]$ concludes the claim.
最后，由于 f 在 $(0, +\infty)$ 上严格凸且可微，f' 在 $(f'(0), f'(\infty))$ 上可逆，其中 $f'(0) = \lim_{x \to 0, x \geq 0} f'(x)$ 且 $f'(\infty) = \lim_{x \to +\infty} f'(x)$。此外，对于任何 $y \in (f'(0), f'(\infty))$，我们有 $(f + \iota_{\mathbb{R}_+})'_*(y) = f'_*(y) = (f')^{-1}(y)$，其中 $(f')^{-1}$ 是 f' 在 $(f'(0), f'(\infty))$ 上的逆。由于 f' 在 $(0, +\infty)$ 上递增，因此 $(f')^{-1}$ 也在 $(f'(0), f'(\infty))$ 上递增，并且 f'_* 和 $(f + \iota_{\mathbb{R}_+})'_*$ 也递增。注意到 $(f'(0), f'(\infty)) = \operatorname{dom} f'_* \setminus (-\infty, f'(0)] = (\operatorname{dom} f + \iota_{\mathbb{R}_+})'_* \setminus (-\infty, f'(0)]$ 可得出上述结论。 □