Two-temperature logistic regression based on the Tsallis divergence

Ehsan Amid[*] **Manfred K. Warmuth**[*,†] **Sriram Srinivasan**[*]

∗ University of California, Santa Cruz
† Google Brain, Zürich
{eamid, manfred, ssriniv9}@ucsc.edu

Abstract

We develop a variant of multiclass logistic regression that is significantly more robust to noise. The algorithm has one weight vector per class and the surrogate loss is a function of the linear activations (one per class). The surrogate loss of an example with linear activation vector $\mathbf{a}$ and class c has the form $-\log_{t_1} \exp_{t_2}(a_c - G_{t_2}(\mathbf{a}))$ where the two temperatures t_1 and t_2 "temper" the log and exp, respectively, and $G_{t_2}(\mathbf{a})$ is a scalar value that generalizes the log-partition function. We motivate this loss using the Tsallis divergence. Our method allows transitioning between non-convex and convex losses by the choice of the temperature parameters. As the temperature t_1 of the logarithm becomes smaller than the temperature t_2 of the exponential, the surrogate loss becomes "quasi convex". Various tunings of the temperatures recover previous methods and tuning the degree of non-convexity is crucial in the experiments. In particular, quasi-convexity and boundedness of the loss provide significant robustness to the outliers. We explain this by showing that $t_1 < 1$ caps the surrogate loss and $t_2 > 1$ makes the predictive distribution have a heavy tail.

We show that the surrogate loss is Bayes-consistent, even in the non-convex case. Additionally, we provide efficient iterative algorithms for calculating the log-partition value only in a few number of iterations. Our compelling experimental results on large real-world datasets show the advantage of using the two-temperature variant in the noisy as well as the noise free case.

Proceedings of the 22[nd] International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89.

1 Introduction

Consider a classification problem where every instance $\mathbf{x} \in \mathbb{R}^d$ is labeled by one class $c \in \{1, \ldots, C\}$. The goal of learning algorithm is to develop a classifier, parameterized by $\mathbf{W}$, which correctly predicts the class label c of a given instance $\mathbf{x}$. In order to learn the optimal parameter $\mathbf{W}^*$ of the classifier, we minimize the *regularized empirical surrogate loss* of a set of i.i.d. examples $\{(\mathbf{x}_n, c_n)\}_{n=1}^N$ from the data distribution:

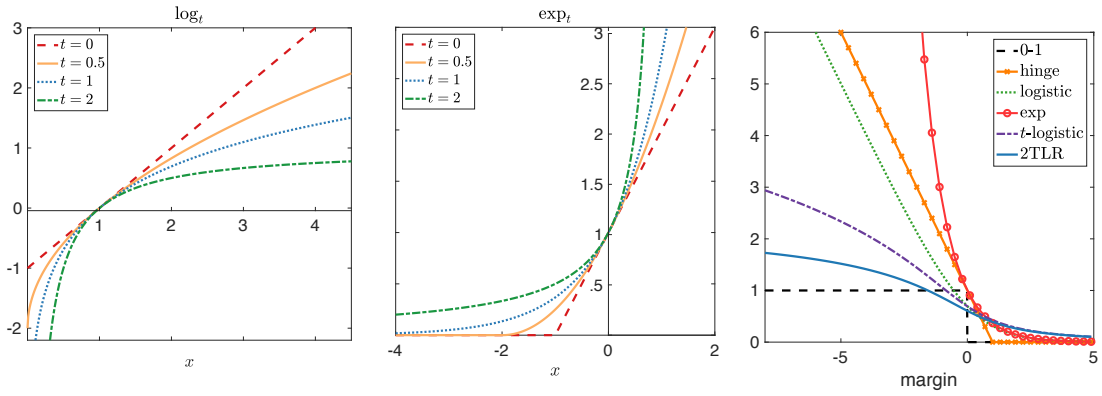
$$\mathbf{W}^* = \arg\min_{\mathbf{W}} \; \mathcal{L}(\mathbf{W}) + \mathcal{R}(\mathbf{W}) \,,$$

where

$$\mathcal{L}(\mathbf{W}) = \frac{1}{N} \sum_n \xi(\mathbf{x}_n, c_n \,|\, \mathbf{W}) \,.$$

Here, $\xi(\mathbf{x}_n, c_n \,|\, \mathbf{W})$ denotes the *surrogate loss*, which replaces the 0-1 loss associated with the example $(\mathbf{x}_n, c_n)$. Also, $\mathbf{W}$ is a $d \times C$ weight matrix and $\mathcal{R}(\mathbf{W})$ a regularizer. The c-th column $\mathbf{w}_c$ is the weight vector for class c. In this paper, we consider the *linear activation* models where both the parameterized classifier and the surrogate loss $\xi(\mathbf{x}, c \,|\, \mathbf{W})$ can be written as functions of the linear *activation* vector $\mathbf{a} = \mathbf{W}^\top \mathbf{x}$.

Among different properties of the surrogate functions used in practice, convexity plays an important role since it provides the convergence guarantee of the solution to a global minimum [11]. Additionally, there exist many convex optimization packages for solving the minimization problem efficiently [10, 22]. The main drawback of the convexity is that the loss of an individual example, e.g., for a highly misclassified outlier point, can grow indefinitely (at least with a linear rate) and dominate the objective function. Therefore, it has been shown that the convex functions are not robust to noise [13]. Specifically, Ben-David et al. [4] showed that among the convex surrogate loss functions for linear predictors, the hinge loss has the lowest expected misclassification error rate and any strongly convex loss has a qualitatively worse guarantee when compared to the hinge loss. To alleviate this problem, several strategies have been



(a) (b) (c)

Figure 1: Generalized logarithm and exponential functions: a) $\log_t$, b) $\exp_t$, and c) Different loss functions for classification of a single example $\mathbf{x}$ with label $c = +1$ as a function of the margin $a = \mathbf{w}^\top \mathbf{x}$. The t-logistic loss of [6] is non-convex (here t-logistic$(a) = -\log \exp_t (a/2 - G_t(a))$ with $t = 1.6$), but goes to $+\infty$ as margin $\to -\infty$. On the other hand, our proposed two-temperature logistic loss $-\log_{t_1} \exp_{t_2}(a_c - G_{t_2}(\mathbf{a}))$ (for e.g. $t_1 = 0.6$, $t_2 = 1.6$) is upper-bounded by $1/(1 - t_1) = 2.5$.

proposed to intorduce non-convexity into the loss function [15, 9, 8, 19, 5]. More recently, Ding et al. [6] used heavy-tailed properties of t-exponential distributions to define a robust loss function for logistic regression. The main idea behind these techniques is to eventually "bend down" the loss and give up on those points that are highly misclassified.

In this paper, we generalize the ideas in [6] for constructing a non-convex surrogate loss as the negative log-likelihood of a t-exponential distribution. Our approach is based on the Tsallis divergence which is the natural choice of divergence for the family of t-exponential distributions [1]. Our definition of surrogate loss involves a generalized logarithm and a generalized exponential function. The generalization imbues each of these functions with a different temperature parameter. By varying the temperatures for the two functions, we transition between the convex and more robust quasi-convex loss functions. More importantly, the loss function becomes bounded for certain choices of the parameters. Figure 1 illustrates the different loss functions used for classification along with an example of our proposed surrogate loss. Even though our generalization of constructing non-convex surrogate losses is strikingly simple, our experiments clearly show that the tail-heaviness by itself (as introduced in [6]) is insufficient for handling the outliers and the label noise. More importantly, controlling the boundedness of the loss is an additional crucial property for obtaining robustness to both outliers and label noise. A similar bounded surrogate loss was recently developed for training deep neural networks in the presence of label noise [25]. Our contributions in this paper can be summarized as follows:

- We generalize the ideas in [6] and [25] by introducing the two-temperature logistic regression (2TRL) which lets us control both the tail-heaviness as well as boundedness of the non-convex surrogate loss.
- We provide fast efficient iterative algorithms for calculating the normalization constant in the t-exponential probabilities.
- We discuss the properties of the surrogate loss for different ranges of the two temperatures (the previous methods become special cases) and the implications of using the Tsallis divergence for parameter estimation. More specifically, we show that *properness* is achieved by switching to the *escort* probability of the optimizer.
- Finally, we show that our loss is Bayes-consistent, even in the non-convex case. While many convex surrogate losses enjoy Bayes-consistency, achieving Bayes-consistency for non-convex losses is a highly non-trivial property and thus, is an important consideration in designing the loss functions for classification [14].

2 Tsallis Entropy and Tsallis Divergence

The $\log_t$ function with *temperature* parameter $t > 0$ is defined as a generalization of the standard log function [17, 18][1],

$$\log_t x = \frac{1}{1-t}(x^{1-t} - 1)\,. \tag{1}$$

[1] Note that in this section, we use x as a scalar input and it should not be confused with the multivariate random variable $\mathbf{x}$.

The $\log_t$ function is monotonically increasing and recovers the standard log function in the limit $t \to 1$. However, some properties of the log function do not generalize to $\log_t$. For instance, $\log_t ab \neq \log_t a + \log_t b$ in general. Additionally, unlike the standard log function, the $\log_t$ function is lower bounded by $-1/(1-t)$ for $0 < t < 1$ and upper bounded by $1/(t-1)$ for $t > 1$ (See Figure 1a). This property has been used to design robust loss transformations for metric learning [2].

Using the $\log_t$ function, we can generalize the notion of the (Shannon) entropy of a probability distribution. For a probability distribution $p(\mathbf{x})$, the Tsallis entropy [21] is defined as

$$H_t(p) = \frac{\int p(\mathbf{x})^t \, d\mathbf{x} - 1}{1-t} = \int p(\mathbf{x}) \log_t \frac{1}{p(\mathbf{x})} d\mathbf{x}. \quad (2)$$

Note that the standard entropy is recovered when $t \to 1$. Similarly, the Tsallis divergence between the distributions $p(\mathbf{x})$ and $q(\mathbf{x})$ can be defined as a generalization of the Kullback-Leibler (KL) divergence, that is,

$$D_t(p\|q) = -\int p(\mathbf{x}) \log_t \frac{q(\mathbf{x})}{p(\mathbf{x})} \, d\mathbf{x}. \quad (3)$$

Note that the KL divergence is also recovered in the limit $t \to 1$. We also define the $\exp_t$ function as the inverse of $\log_t$ (See Figure 1b):

$$\exp_t(x) = [1 + (1-t)\, x]_+^{1/(1-t)} \,, \quad (4)$$

where $[\,\cdot\,]_+ = \max(\,\cdot\,, 0)$. Again the vanilla exp function is the $t \to 1$ limit. An important property of the $\exp_t$ function is its heavier tail compared to exp for values of $t > 1$ (see Figure 1b). This property leads to definition of a class of generalized distributions under the $\exp_t$ function, called the t-exponential family of distributions with vector of sufficient statistics $\mathbf{x}$,

$$p_t(\mathbf{x} \,|\, \theta) = \exp_t(\theta^\top \mathbf{x} - G_t(\theta)), \quad \text{for } t > 0\,. \quad (5)$$

Here θ is called the *canonical parameter* and the convex function $G_t(\theta)$, called the *log-partition function*, ensures that the distribution is normalized, that is,

$$\int \exp_t(\theta^\top \mathbf{x} - G_t(\theta)) \, d\mathbf{x} = 1\,. \quad (6)$$

An important distribution related to the t-exponential distribution (5) is called the *escort distribution* and is defined as

$$q_t(\mathbf{x} \,|\, \theta) = \frac{1}{\mathcal{Z}_t(\theta)} \exp_t(\theta^\top \mathbf{x} - G_t(\theta))^t, \quad (7)$$

where

$$\mathcal{Z}_t(\theta) = \int \exp_t(\theta^\top \mathbf{x} - G_t(\theta))^t \, d\mathbf{x}\,.$$

Here $\mathcal{Z}_t(\theta)$ is the normalization factor. It is easy to see that [1]

$$\nabla G_t(\theta) = \mathbb{E}_{q_t}[\mathbf{x}] = \frac{1}{\mathcal{Z}_t(\theta)} \int \mathbf{x} \exp_t(\mathbf{x}^\top \theta - G_t(\theta))^t \, d\mathbf{x}. \quad (8)$$

As (8) suggest, escort probabilities appear when calculating the gradient of the loss, as we will see in the later sections. When dealing with t-exponential distributions, the Tsallis entropy and divergence take the role of Shannon entropy and KL divergence respectively, for the vanilla exponential family (See e.g. [1]).

3 Two-temperature Logistic Regression

Let $\mathbf{a} = \mathbf{W}^\top \mathbf{x}$. Following the discussion on the heavy-tail properties of the t-exponential family of distributions in [6], we model the conditional probability of the class c given input $\mathbf{x}$ with a t-exponential distribution with temperature t_2:

$$\begin{aligned} \hat{p}_{t_2}(c \,|\, \mathbf{x}, \mathbf{W}) &= \exp_{t_2}(\mathbf{w}_c^\top \mathbf{x} - G_{t_2}(\mathbf{W}^\top \mathbf{x})) \\ &= \exp_{t_2}\left(a_c - G_{t_2}(\mathbf{a})\right) \,, \end{aligned} \quad (9)$$

where the log-partition function $G_{t_2}(\mathbf{a})$ ensures that the probabilities sum up to 1, that is,

$$\sum_c \exp_{t_2}(a_c - G_{t_2}(\mathbf{a})) = 1\,. \quad (10)$$

This definition for the conditional probabilities is similar to the ones given in [6]. The definition (9) also includes the softmax probabilities as a special case when $t_2 = 1$:

$$\begin{aligned} \hat{p}_1(c \,|\, \mathbf{x}, \mathbf{W}) &= \exp(a_c - \overbrace{\log \sum_j \exp(a_j)}^{G_1(\mathbf{a})}) \\ &= \frac{\exp(a_c)}{\sum_j \exp(a_j)} \,. \end{aligned} \quad (11)$$

In order to adopt the heavy-tail properties of t-exponential distribution, we are mainly interested in the values of $t_2 > 1$. However, for values of $t_2 \neq 1$, the log-partition function $G_{t_2}(\mathbf{a})$ does not have a closed form solution in general and must be calculated numerically: We provide an iterative method for computing $G_{t_2}(\mathbf{a})$ efficiently (Algorithm 1).

Given the prediction probabilities (9) in the form of a t_2-exponential distribution, we can now define the loss between the empirical label distribution $p_e(c \,|\, \mathbf{x}_n) = \mathbb{I}_{c=c_n}$, and the prediction $\hat{p}_{t_2}(c \,|\, \mathbf{x}_n)$ using a sum of

Algorithm 1 Iterative algorithm for computing G_t for multiclass 2TLR.

Input: Vector of activations $\mathbf{a}$, temperature $t > 1$
Output: $G_t(\mathbf{a})$
$\mu \leftarrow \max(\mathbf{a})$
$\tilde{\mathbf{a}} \leftarrow \mathbf{a} - \mu$
while $\tilde{\mathbf{a}}$ not converged **do**
  $Z(\tilde{\mathbf{a}}) \leftarrow \sum_{c=1}^{C} \exp_t(\tilde{a}_c)$
  $\tilde{\mathbf{a}} \leftarrow Z(\tilde{\mathbf{a}})^{1-t}(\mathbf{a} - \mu)$
end while
$G_t(\mathbf{a}) \leftarrow -\log_t(1/Z(\tilde{\mathbf{a}})) + \mu$

Tsallis divergences with temperature t_1:

$$\mathcal{L}(\mathbf{W}) = -\frac{1}{N}\sum_n\sum_c p_e(c\,|\,\mathbf{x}_n)\log_{t_1}\frac{\hat{p}_{t_2}(c\,|\,\mathbf{x}_n,\mathbf{W})}{p_e(c\,|\,\mathbf{x}_n)} = -\frac{1}{N}\sum_n\sum_c \mathbb{I}_{c=c_n}\log_{t_1}\frac{\hat{p}_{t_2}(c\,|\,\mathbf{x}_n,\mathbf{W})}{\mathbb{I}_{c=c_n}}. \quad (12)$$

Justified by a limit argument, $0\times\log_t 0 = 0\times\log_t\infty = 0$, the loss (12) simplifies to

$$\mathcal{L}(\mathbf{W}) = -\frac{1}{N}\sum_n \log_{t_1}\hat{p}_{t_2}(c_n\,|\,\mathbf{x}_n,\mathbf{W}) = \frac{1}{N}\sum_n \underbrace{[-\log_{t_1}\exp_{t_2}(\mathbf{w}_{c_n}^\top\mathbf{x}_n - G_{t_2}(\mathbf{W}^\top\mathbf{x}_n))]}_{\xi_{t_1}^{t_2}(\mathbf{x}_n,c_n\,|\,\mathbf{W})}. \quad (13)$$

We refer to the classification algorithm with the loss defined in (13) as *Two-Temperature Logistic Regression* (2TLR). The gradient of the loss with respect to the c-th parameter $\mathbf{w}_c$ can be written as

$$\nabla_{\mathbf{w}_c}\mathcal{L}(\mathbf{W}) = -\sum_n \hat{p}_{t_2}(c_n\,|\,\mathbf{x}_n,\mathbf{W})^{t_2-t_1}\Big(\mathbb{I}_{c=c_n} - \hat{q}_{t_2}(c\,|\,\mathbf{x}_n,\mathbf{W})\Big)\mathbf{x}_n, \quad (14)$$

where

$$\hat{q}_{t_2}(c\,|\,\mathbf{x},\mathbf{W}) = \frac{\exp_{t_2}(a_c - G_{t_2}(\mathbf{a}))^{t_2}}{\sum_j \exp_{t_2}(a_j - G_{t_2}(\mathbf{a}))^{t_2}} \sim \hat{p}_{t_2}(c\,|\,\mathbf{x},\mathbf{W})^{t_2}$$

is the escort distribution of $\hat{p}_{t_2}(c\,|\,\mathbf{x},\mathbf{W})$.

We are mainly interested in $0 < t_1 < 1$ because for this range, the loss of each individual observation becomes capped by the constant $1/(1-t_1)$. As we show in the experiments, the boundedness of loss provides significant improvement in handling noisy observations. Note that the gradient of the loss of the n-th observation contains an *importance factor* of the form $\hat{p}_{t_2}(c_n\,|\,\mathbf{x}_n,\mathbf{W})^{t_2-t_1}$ that depends on the conditional probability of the n-th observation and the temperature gap $t_2 - t_1$. Note that for $t_2 > t_1$, the temperature gap is non-negative and the importance factors dampen the gradient of those observations that have small probabilities towards zero. Also the loss of each observation is bounded only for values of $0 < t_1 < 1$. On the other hand, the importance factors vanish when $t_1 = t_2$. In particular, it vanishes for standard logistic regression (i.e. when $t_1 = t_2 = 1$).

Next we focus on the binary classification and analyze the properties of the surrogate loss in this case.

4 Binary Classification

For $C = 2$, we use the classes $c \in \{\pm 1\}$ and denote the parameter vector as $\mathbf{W} = [\mathbf{w}_+, \mathbf{w}_-]$ and linear activations as $\mathbf{a} = [\mathbf{w}_+^\top\mathbf{x},\, \mathbf{w}_-^\top\mathbf{x}]^\top = [a_+, a_-]^\top$. Similar to (9), we can define the probabilities as

$$\hat{p}_{t_2}(c = \pm 1|\mathbf{x}) = \exp_{t_2}(\mathbf{w}_\pm^\top\mathbf{x} - G_{t_2}(\mathbf{W}^\top\mathbf{x})) = \exp_{t_2}(a_\pm - G_{t_2}(\mathbf{a}))\,. \quad (15)$$

The log-partition function $G_{t_2}(\mathbf{a})$ ensures that the two probabilities sum to 1. It is easy to see that for any constant b, $G_{t_2}(\mathbf{a} + b\,\mathbf{1}) = G_{t_2}(\mathbf{a}) + b\,\mathbf{1}$. Therefore we can simplify the margin vector $\mathbf{a}$ by subtracting the mean of the inner-products $\frac{\mathbf{w}_+^\top\mathbf{x}+\mathbf{w}_-^\top\mathbf{x}}{2}$, that is, $\mathbf{a} = [\frac{(\mathbf{w}_+-\mathbf{w}_-)^\top\mathbf{x}}{2}, -\frac{(\mathbf{w}_+-\mathbf{w}_-)^\top\mathbf{x}}{2}]^\top = [\frac{\mathbf{w}^\top\mathbf{x}}{2}, -\frac{\mathbf{w}^\top\mathbf{x}}{2}]^\top = [\frac{a}{2}, -\frac{a}{2}]^\top$, where we define $\mathbf{w} = \mathbf{w}_+ - \mathbf{w}_-$. Thus, we can write the probabilities in the following compact form

$$\hat{p}_{t_2}(c\,|\,\mathbf{x},\mathbf{w}) = \exp_{t_2}(\frac{c}{2}\overbrace{\mathbf{w}^\top\mathbf{x}}^{a} - G_{t_2}(\mathbf{w}^\top\mathbf{x}))\,.$$

This definition contains the logistic probabilities as the special case when $t_2 = 1$:

$$\hat{p}_1(c\,|\,\mathbf{x}) = \frac{\exp(\frac{c}{2}\,a)}{\exp(\frac{c}{2}\,a) + \exp_{t_2}(\frac{-c}{2}\,a)} = \frac{1}{1+\exp(-c\,a)}\,,$$

since $G_1(a) = \log\left(\exp\frac{a}{2} + \exp\frac{-a}{2}\right)$. For $t_2 \neq 1$, $G_{t_2}(a)$ does not have a closed form solution[2] and we provide a variant of the iterative algorithm for calculating $G_t(a)$ for the binary case (Algorithm 2).

Following similar steps as in (12), we can write the loss for the binary case as

$$\mathcal{L}(\mathbf{w}) = \sum_n \underbrace{-\log_{t_1}\exp_{t_2}(\frac{c_n}{2}\,a_n - G_{t_2}(a_n))}_{\xi_{t_1}^{t_2}(\mathbf{x}_n,y_n\,|\,\mathbf{w})}\,. \quad (16)$$

where $a_n = \mathbf{w}^\top\mathbf{x}_n$. For $t_1 = t_2 = 1$, the above loss is the standard logistic regression loss. Also for $t_1 = 1$ and $t_2 = t > 1$, the above becomes the t-logistic loss

[2] Except for $t_2 = 2$.

Algorithm 2 Iterative algorithm for computing G_t for binary 2TLR.

Input: Activation $a > 0$, temperature $t > 1$
Output: $G_t(a)$
if $t == 2$ **then**
  $G_t(a) \leftarrow \sqrt{a^2/4 + 1}$
  return
end if
$\tilde{a} \leftarrow a$
while $\tilde{a}$ not converged **do**
  $Z(\tilde{a}) \leftarrow 1 + \exp_t(-\tilde{a})$
  $\tilde{a} \leftarrow Z(\tilde{a})^{1-t} a$
end while
$G_t(a) \leftarrow -\log_t(1/Z(\tilde{a})) + a/2$

of [6]. The gradient of the loss (16) wrt $\mathbf{w}$ is

$$\nabla \mathcal{L}(\mathbf{w}) = -\frac{1}{2} \sum_n \hat{p}_{t_2}(c_n \,|\, \mathbf{x}_n, \mathbf{w})^{t_2 - t_1} \Big(c_n - \sum_c c\, \hat{q}_{t_2}(c \,|\, \mathbf{x}, \mathbf{w}) \Big) \mathbf{x}_n \,,$$

where $\hat{q}_{t_2}(c \,|\, \mathbf{x}, \mathbf{w}) \sim \hat{p}_{t_2}(c \,|\, \mathbf{x}, \mathbf{w})^{t_2}$ is the escort distribution.

4.1 Properties

The curvature of the two-temperature loss function $\xi_{t_1}^{t_2}(\mathbf{x}, y \,|\, \theta)$ depends on the choice of the temperature parameters t_1 and t_2. For certain choices, we still have convex losses while for the others, the loss function shows a quasi-convex behavior. The properties of the loss function are summarized below. Without loss of generality, we assume $c = +1$.

Remark 1. The loss function $\xi_{t_1}^{t_2}(\mathbf{x}, c \,|\, \mathbf{w}) = -\log_{t_1} \exp_{t_2}(\frac{a}{2} - G_{t_2}(a))$ has the following properties:

1. For values of $t_1 \geq t_2$ and $t_1 \geq 1$, the loss function is convex. Specifically, for $t_1 = t_2 = t \geq 1$, we have the convex loss
$$\xi_t^t(\mathbf{x}, c \,|\, \theta) = G_t(a) - \frac{a}{2} \,. \tag{17}$$
Moreover, the curvature of the function increases with the temperature gap $t_1 - t_2 > 0$.
2. The function is quasi-convex for $t_1 < t_2$ or for any $t_2 \geq 0$ when $t_1 < 1$.

The proof is provided in the Appendix B.

5 Implications of Using the Tsallis Divergence

We briefly discuss the implicit assumptions behind using the Tsallis divergence for parameter estimation. Consider modeling the (unknown) posterior distribution $p(c \,|\, \mathbf{x})$ for the set of random variables $(\mathbf{x}, c) \in \mathbb{R}^d \times \{1, \ldots, C\}$ using a discriminative model $\hat{p}_{\mathcal{M}}(c \,|\, \mathbf{x})$. For this purpose, consider minimizing the expected Tsallis divergence between the class posterior distribution of the data and the predicted posterior probabilities, that is,

$$\mathbb{E}_{\mathbf{x}} \left[-\sum_c p(c \,|\, \mathbf{x}) \log_t \frac{\hat{p}_{\mathcal{M}}(c \,|\, \mathbf{x})}{p(c \,|\, \mathbf{x})} \right] \tag{18a}$$

$$= \mathbb{E}_{\mathbf{x}} \Big[\sum_c p(c \,|\, \mathbf{x})^t \Big[\log_t p(c \,|\, \mathbf{x}) - \log_t \hat{p}_{\mathcal{M}}(c \,|\, \mathbf{x}) \Big] \Big] \tag{18b}$$

$$= -H_t - \int \sum_c p(c \,|\, \mathbf{x})^t \log_t \hat{p}_{\mathcal{M}}(c \,|\, \mathbf{x})\, p(\mathbf{x})\, d\mathbf{x} \tag{18c}$$

$$\approx -H_t - \sum_n \sum_c \mathbb{I}_{c=c_n} \log_t \hat{p}_{\mathcal{M}}(c_n \,|\, \mathbf{x}_n) \tag{18d}$$

$$= -H_t - \sum_n \log_t \hat{p}_{\mathcal{M}}(c_n \,|\, \mathbf{x}_n) \,, \tag{18e}$$

in which $H_t = -\int \sum_c p(c \,|\, \mathbf{x})^t \log_t p(c \,|\, \mathbf{x})\, p(\mathbf{x})\, d\mathbf{x} = \mathbb{E}_{\mathbf{x}} \left[\sum_c p(c \,|\, \mathbf{x}) \log_t \frac{1}{p(c \,|\, \mathbf{x})} \right]$ is the expected Tsallis entropy of the posterior distribution $p(c \,|\, \mathbf{x})$ and is a constant. Note that from (18a) to (18b) we use the property $\log_t(u/v) = u^t(\log_t u - \log_t v)$ of the $\log_t$ function and from (18c) to (18d) we perform a Monte Carlo approximation of the integral and sum using a set of samples $\{\mathbf{x}_n, c_n\}$. Therefore, we can eliminate the second sum in (18d) and only keep the terms corresponding to the observed labels, as in (18e). However, indeed, minimizing the sum in (18e) involves the implicit assumption that the c samples are drawn from the tempered conditional distribution $\sim p(c \,|\, \mathbf{x})^t$ and therefore, the minimizer solves $\hat{p}^*_{\mathcal{M}}(c \,|\, \mathbf{x}) \sim p(c \,|\, \mathbf{x})^{1/t}$. Thus, as a consequence of using the Tsallis divergence in (18e), the surrogate loss $\xi_{t_1}^{t_2}(\mathbf{x}_n, c_n \,|\, \mathbf{w})$ is not *proper* [23], i.e., $\hat{p}_{t_2}(c \,|\, \mathbf{x}, \mathbf{W}^*) \neq p(c \,|\, \mathbf{x})$. However, simply enough, the *escort* probabilities $\sim \hat{p}_{t_2}(c \,|\, \mathbf{x}, \mathbf{W}^*)^{t_1}$ match to the correct conditional probabilities. In the case of $t = 1$, the Tsallis divergence reduces to the KL-divergence and we recover the maximum-likelihood estimation $-\sum_n \log \hat{p}_{\mathcal{M}}(c_n | \mathbf{x}_n) = -\log \prod_n \hat{p}_{\mathcal{M}}(c_n | \mathbf{x}_n)$ and $\hat{p}^*_{\mathcal{M}}(c | \mathbf{x}) = p(c | \mathbf{x})$.

Although the properness of the loss function may be important in density estimation applications, for the classification problem, the estimated posterior probabilities are irrelevant as long as the class label is predicted correctly. Thus, we are mainly interested in the Bayes-consistency property of the loss [3, 20], which guarantees that at the solution, the correct label can be predicted using the arg max of the margin vector $\mathbf{a}$.

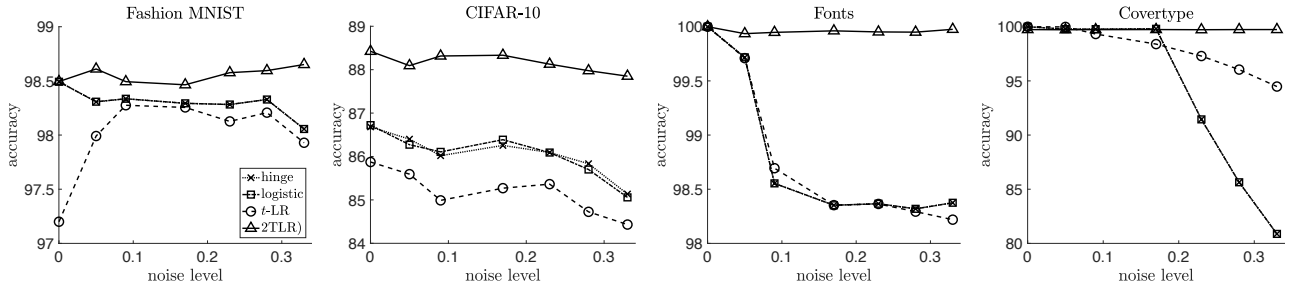


Figure 2: The classification accuracy in the presence of instance noise. The errorbars are small and not shown to avoid clutter.

Dataset (#instances, #dim)	**Noise Type**	**Classification Accuracy (%)**			
		hinge	**logistic**	***t*-LR**	**2TLR**
Fashion MNIST (20K, 784)	random	96.42 ± 0.59	96.42 ± 0.59	94.09 ± 0.48	$\mathbf{99.80 \pm 0.12}$
	small-margin	98.50 ± 0.26	98.50 ± 0.26	97.35 ± 0.42	$\mathbf{99.13 \pm 0.37}$
	large-margin	96.42 ± 0.59	96.42 ± 0.59	94.09 ± 0.48	$\mathbf{99.80 \pm 0.12}$
CIFAR-10 (10.8K, 1024)	random	84.27 ± 1.12	84.39 ± 1.17	82.11 ± 1.01	$\mathbf{87.75 \pm 1.40}$
	small-margin	84.94 ± 0.97	84.94 ± 0.99	84.22 ± 0.79	$\mathbf{86.28 \pm 1.18}$
	large-margin	77.79 ± 1.20	77.77 ± 1.20	72.58 ± 1.44	$\mathbf{88.56 \pm 1.20}$
Fonts (143K, 411)	random	83.78 ± 0.28	83.78 ± 0.28	$\mathbf{84.14 \pm 0.27}$	$\mathbf{84.14 \pm 0.27}$
	small-margin	$\mathbf{83.60 \pm 0.34}$	$\mathbf{83.60 \pm 0.34}$	83.38 ± 0.36	$\mathbf{83.60 \pm 0.34}$
	large-margin	72.39 ± 0.32	72.39 ± 0.32	$\mathbf{72.61 \pm 0.30}$	$\mathbf{72.61 \pm 0.30}$
Covertype (287K, 54)	random	97.52 ± 0.88	97.52 ± 0.88	$\mathbf{99.26 \pm 0.05}$	$\mathbf{99.26 \pm 0.05}$
	small-margin	96.79 ± 0.11	96.79 ± 0.11	$\mathbf{97.25 \pm 0.05}$	$\mathbf{97.25 \pm 0.05}$
	large-margin	83.59 ± 0.24	83.59 ± 0.24	84.79 ± 0.20	$\mathbf{94.03 \pm 0.13}$

Table 1: Classification accuracy with 10% label noise. The noise is added by selecting the points in three different manners: 1) Random: points are selected uniformly at random, 2) Small-Margin (SM): the points having smallest margin are selected, 3) Large-Margin (LM): the points having largest margin are selected.

6 Bayes-consistency

We use the results from Zhang et al. [24] to show the Bayes-consistency of the multiclass class case.

Definition 2 (Zhang et al. [24])**.** A surrogate loss $\xi(\mathbf{a}, c)$ w.r.t. a margin $\mathbf{a} = [a_1, \ldots, a_m]^\top$ with the additional constraint $\sum_c a_c = 0$ is said to be Bayes-consistent if for all possible label probability distributions $p(c\,|\,\mathbf{x})$ the following conditions are satisfied:

1. The minimization problem $\mathbf{a}^* = \arg\min_{\mathbf{a}} \sum_c p(c|\mathbf{x})\, \xi(\mathbf{a}, c)$ has a unique solution for all $\mathbf{x} \in \mathbb{R}^d$, and
2. $\arg\max_c a_c^* = \arg\max_c p(c\,|\,\mathbf{x})$ for all $\mathbf{x} \in \mathbb{R}^d$.

We now prove the following.

Theorem 3. *The multiclass surrogate loss* $\xi_{t_1}^{t_2}(\mathbf{x}, c|\,\mathbf{W}) = -\log_{t_1} \exp_{t_2}(a_c - G_{t_2}(\mathbf{a}))$ *is Bayes-consistent.*

Proof. The minimizer of the expectation

$$-\sum_c p(c\,|\,\mathbf{x})\, \log_{t_1} \exp_{t_2}(a_c - G_{t_2}(\mathbf{a})) \tag{19}$$

has the unique solution $\mathbf{a}^*$ such that $\exp_{t_2}(a_c^* - G_{t_2}(\mathbf{a}^*)) \propto p(c\,|\,\mathbf{x})^{1/t_1}$. Note that the minimizer is unique because $\exp_{t_2}$ is an injective function and therefore any other minimizer $\mathbf{a}^{**}$ must satisfy the following: $a_c^* - G_{t_2}(\mathbf{a}^*) = a_c^{**} - G_{t_2}(\mathbf{a}^{**})$ for all $c \in \{1, \ldots, C\}$. Enforcing the constraint[3] $\sum_c a_c^* = \sum_c a_c^{**} = 0$ yields $\mathbf{a}^* = \mathbf{a}^{**}$. Finally, monotonicity of $\exp_{t_2}$ function implies

$$\begin{aligned}\arg\max_c a_c^* &= \arg\max_c \exp_{t_2}(a_c^* - G_{t_2}(\mathbf{a}^*)) \\ &= \arg\max_c p(c\,|\mathbf{x})^{1/t_1} = \arg\max_c p(c\,|\mathbf{x})\,. \;\square\end{aligned}$$

[3]Note that we can always enforce the constraint $\sum_c a_c = 0$ by adding and subtracting the constant vector of mean value $\left(\frac{1}{C}\sum_c a_c\right)\mathbf{1}$ without changing the probabilities since $G_{t_2}(\mathbf{a} + b\,\mathbf{1}) = G_{t_2}(\mathbf{a}) + b\,\mathbf{1}$ for any constant b.

The result of Theorem 3, i.e. $\hat{p}_{t_2}(\mathbf{x}, c|\mathbf{W}^*) \propto p(c|\mathbf{x})^{1/t_1}$, is the direct consequence of using the sum of Tsallis divergences between the observed class distributions and the predicted class probabilities, as discussed in the previous section. However, the arg max operator is invariant with respect to the positive powers and thus, we still achieve Bayes-consistency.

Corollary 4. *The binary surrogate loss* $\xi_{t_1}^{t_2}(\mathbf{x}, c|\,\mathbf{w})$ *is Bayes-consistent.*

Note that because of the form of the margin vector $\mathbf{a} = [a, -a]^\top$ in the binary case, the arg max operator is equivalent to sign(a). Therefore, the given new points can simply be classified using the sign of the activation, without explicitly calculating the probabilities.

7 Experiments

We compare the binary classification accuracy when minimizing the following losses: our two-temperature surrogate loss (2TLR), vanilla logistic regression (LR), hinge loss, and t-logistic regression (t-LR). We do not compare our results to the method recently proposed by Feng et al. [7] which is based on detecting and removing the outliers in the dataset. The method in [7] makes strict assumptions about the type of the generative distribution, the availability of the noise variance and requires an upper-bound on the number of outliers. These assumptions make their method impractical for real-world applications.

Our experiments are for the following data sets: 1) Fashion MNIST [4], 2) CIFAR-10 [5], 3) Character Font Images [6], and 4) Covertype[6]. For each dataset, we randomly pick two classes such that the number instances from each class are roughly the same. The size and number of dimensions of each dataset is shown in the first column of Table 1.

For each dataset, we randomly consider 10% of the instances for test and perform 10-fold cross validation on the remaining part to find the optimal set of parameters for each method. These parameters include the L_2-regularizer values for all methods and temperature values for t-LR and 2TLR. The regularizer values are selected from the range $[10^{-5}, 10^{-1}]$. The range of temperature values for t-LR is chosen to be $[1.12, 1.9]$ and the range for t_1 and t_2 temperatures are set to $[0.1, 1]$ and $[1.12, 1.9]$, respectively. The values of all parameters are chosen using cross-validation. More specifically, the value of temperature for t-LR is set to 1.12 for the CIFAR-10, Covertype, and Fashion MNIST, and 1.3 for the Fonts dataset. For 2TLR, we set t_2 to be the same as in t-LR and for t_1, we use 0.1 for CIFAR-10, Covertype and Fashion MNIST, and 0.9 for the Fonts dataset. The results are averaged over 10 random train-test splits. We perform experiments in the presence of instance and label noise. All experiments are done on a 24 core cluster with 128 GB of RAM. We use a parallel implementation which utilizes all the cores in a machine.

We use the L-BFGS method for minimizing the losses. The initial weights are set to values sampled from a zero-mean Gaussian distribution with std $= 0.001$. In general, t-LR and our 2TLR method are non-convex and converge to a local minimum. However, the results are consistent over multiple random initializations. This can be verified by the std of the accuracy results in Table 1. Note that we observed the method to converge to bad local minima for std > 0.01.

7.1 Instance Noise

For the instance noise experiments, we consider the case where a subset of the training instances, chosen uniformly at random are replaced by instances from the remaining set of classes (i.e. those classes other than the two selected classes for the binary classification). This resembles the case of a multiclass dataset where a subset of the instances from each class are mislabled as instances of other classes. Therefore these mislabled instances often become extreme outliers for the class they are wrongly labeled with.

Figure 2 shows the results in the presence of different amounts of this type of instance noise. The new 2TLR method is significantly more robust to this noise than all the other methods and its performance is not considerably affected by up to 33% noise. The main reason for robustness of our method is the fact that by capping the surrogate loss, the total loss of the method is not affected much by the loss of each individual instance. This also validates our claim that tail-heaviness of the distribution by itself (as used in t-LR) cannot handle the outliers as well: In some case t-LR provides even worse results than LR and all are beaten by 2TLR.

7.2 Label Noise

We consider the label noise experiments where the labels of a subset of the training instances is flipped. Note that unlike the instance noise which alters the input distribution $p(\mathbf{x})$, the labels noise targets the distribution of the labels $p(c\,|\,\mathbf{x})$. Therefore, the label noise is generally handled by first approximating the label inversion rates and then, correcting the data distribution by reweighting the loss of individual instances

[4] Available at: https://github.com/zalandoresearch/fashion-mnist

[5] Available at: https://www.cs.toronto.edu/~kriz/cifar.html

[6] From the UCI repository.

Dataset (#instances, #dim)	Runtime (s)			
	hinge	**logistic**	***t*-LR**	**2TLR**
Fashion MNIST (20K, 784)	4.40 ± 0.28	4.57 ± 0.12	7.02 ± 0.29	7.35 ± 1.21
CIFAR-10 (10.8K, 1024)	31.90 ± 0.22	31.86 ± 0.29	35.08 ± 0.81	28.34 ± 10.56
Fonts (143K, 411)	49.47 ± 4.92	49.75 ± 4.99	82.85 ± 4.98	58.78 ± 6.14
Covertype (287K, 54)	6.45 ± 0.17	6.42 ± 0.18	66.66 ± 1.20	24.53 ± 1.24

Table 2: Runtime of the different algorithms in seconds.

or considering a label-dependent surrogate loss [12, 16]. Nevertheless, the noise can be alleviated to some extent by the tail-heaviness of the modeling distribution [6]. In addition to tail-heaviness, we show that in some cases, tuning the level of non-convexity and bounding the loss function also improves the performance.

We consider the "random" label noise where the label of a uniformly sampled subset of points is flipped. The subset of the noisy instances can also be selected by an adversarial mechanism that targets the training instances based on a certain notion of "importance". We also consider "small-margin" and "large-margin" label noise in which we first train a LR classifier on the noise free data and calculate the margin $c \cdot (\mathbf{w}^\top \mathbf{x})$ of each datapoint. Next, we select the desired portion of the correctly classified datapoints that receptively have the smallest and largest margins. Therefore, these two noise mechanisms target different type of instances, i.e. those closer to the decision boundary and those that are far away. Table 1 shows the results under 10% noise. 2TLR consistently has superior performance in all cases on all datasets. In some cases, the optimal value of the temperatures coincides with the values for LR ($t_1 = t_2 = 1$) and t-LR ($t_1 = 1$, $t_2 > 1$). However, in most cases, the optimal performance is achieved when $0 < t_1 < 1$.

7.3 Runtime

Table 2 shows the runtime of the optimization step of the methods. In general, the runtime of the 2TLR is comparable to the other methods, and in some cases the convergence time is faster than the vanilla logistic regression (LR). However, in some cases (e.g. Covertype dataset), 2TLR takes considerably longer time to converge. In particular, the overhead from calculating the G_t values is negligible[7]; the iterative algorithm takes around 20 iterations to converge to an accuracy of 10^{-10}.

[7] This was validated empirically by comparing to the `fzero` function in MATLAB, but the results are omitted.

8 Conclusions

We developed a generalized loss function for logistic regression which provides two temperatures to tune the properties of the loss. The first temperature tunes the level of non-convexity and the boundedness of the loss while the second one controls the tail-heaviness of the probabilities. Our experiments indicate that tuning the level of the non-convexity and boundedness is a crucial property for obtaining robustness to both instance and label noise while the computation time is comparable to logistic regression.

Acknowledgement

The authors would like to thank Nan Ding for his help with the iterative algorithms for calculating the normalization constants.

References

[1] Shun-ichi Amari, Atsumi Ohara, and Hiroshi Matsuzoe. Geometry of deformed exponential families: Invariant, dually-flat and conformal geometries. *Physica A: Statistical Mechanics and its Applications*, 391(18):4308–4319, 2012.

[2] E. Amid and M. K. Warmuth. A more globally accurate dimensionality reduction method using triplets. *arXiv preprint arXiv:1803.00854*, 2018.

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

[4] Shai Ben-David, David Loker, Nathan Srebro, and Karthik Sridharan. Minimizing the misclassifica-

tion error rate using a surrogate convex loss. *arXiv preprint arXiv:1206.6442*, 2012.

[5] Christophe Croux and Gentiane Haesbroeck. Implementing the bianco and yohai estimator for logistic regression. *Computational statistics & data analysis*, 44(1):273–295, 2003.

[6] Nan Ding and S. V. N. Vishwanathan. t-logistic regression. In *Proceedings of the 23th International Conference on Neural Information Processing Systems*, NIPS'10, pages 514–522, Cambridge, MA, USA, 2010.

[7] Jiashi Feng, Huan Xu, Shie Mannor, and Shuicheng Yan. Robust logistic regression and classification. In *Advances in neural information processing systems*, pages 253–261, 2014.

[8] Yoav Freund. An adaptive version of the boost by majority algorithm. *Machine learning*, 43(3):293–318, 2001.

[9] Yoav Freund. A more robust boosting algorithm. *arXiv preprint arXiv:0905.2138*, 2009.

[10] Michael Grant and Stephen Boyd. Cvx: Matlab software for disciplined convex programming, 2008.

[11] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Fundamentals of convex analysis*. Springer Science & Business Media, 2012.

[12] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2016.

[13] Philip M Long and Rocco A Servedio. Random classification noise defeats all convex potential boosters. In *Proceedings of the 25th international conference on Machine learning*, pages 608–615. ACM, 2008.

[14] Hamed Masnadi-Shirazi. *The design of Bayes consistent loss functions for classification*. PhD thesis, University of California, San Diego, 2011.

[15] Robert Cameron Mitchell and Richard T Carson. *Using surveys to value public goods: the contingent valuation method*. Resources for the Future, 1989.

[16] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in neural information processing systems*, pages 1196–1204, 2013.

[17] Jan Naudts. Deformed exponentials and logarithms in generalized thermostatistics. *Physica A*, 316:323–334, 2002.

[18] Jan Naudts. Generalized thermostatistics and mean-field theory. *Physica A*, 332:279–300, 2004.

[19] Seo Young Park and Yufeng Liu. Robust penalized logistic regression with truncated loss functions. *Canadian Journal of Statistics*, 39(2):300–323, 2011.

[20] Ambuj Tewari and Peter L Bartlett. On the consistency of multiclass classification methods. *Journal of Machine Learning Research*, 8(May):1007–1025, 2007.

[21] Constantino Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of statistical physics*, 52(1):479–487, 1988.

[22] Madeleine Udell, Karanveer Mohan, David Zeng, Jenny Hong, Steven Diamond, and Stephen Boyd. Convex optimization in Julia. *SC14 Workshop on High Performance Technical Computing in Dynamic Languages*, 2014.

[23] Robert C. Williamson, Elodie Vernet, and Mark D. Reid. Composite multiclass losses. *Journal of Machine Learning Research*, 17(223):1–52, 2016.

[24] Zhihua Zhang, Michael Jordan, Wu-Jun Li, and Dit Yan Yeung. Coherence functions for multicategory margin-based classification methods. In *Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS), Clearwater Beach, Florida, USA*, 2009.

[25] Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in Neural Information Processing Systems*, pages 8792–8802, 2018.