Minimizing *convex* loss functions is the holy grail of Machine Learning. A plethora of models based on this paradigm have been proposed over the past several decades. A fundamental problem with all such models is that they are not robust to outliers. In contrast, we explore the consequences of building probabilistic models with a parametric family of distributions that have been proposed in statistical physics. This leads to loss functions which are *quasi-convex* and flatten out for misclassified points which are far away from the decision boundary. Consequently, the models are robust to outliers. In this proposal we outline a research agenda to make models based on these new distribution families practical, and to study their generality and applicability to machine learning.

Building upon our recent work on $t$-logistic regression, a generalization of logistic regression, we will show how conditional models based on this new family of distributions can be developed. The key challenge when working with these generalized distribution families, as in the case of the exponential family, is to compute the log-partition function and perform inference efficiently. We will address this challenge for two specific cases. For problems such as multiclass classification where the number of classes is fairly small, we will develop exact iterative algorithms. On the other hand, for problems such as sequence classification where the number of classes is exponentially large, we will develop approximate inference techniques by extending variational methods. We will also explore models that drop the normalization constraint. This sidesteps the issue of computing the log-partition function. We have done large scale experiments with models that employ convex loss functions and made all our implementations available. Building on this work, we will do systematic benchmark comparisons of the new algorithms with the previous ones and publicize our code.

**Intellectual Merits:** Much of machine learning is based on convex loss functions. Models which use quasi-convex losses are shunned because they lead to objective functions with a number of local minima. However, working with generalized exponential families yields probabilistic models which are well behaved and empirically exhibit a single globally optimal solution. We will develop theory to show this property rigorously. Our study, if successful, will lead to one of the first machine learning algorithms where globally useful properties can be proved even though the loss employed is not convex.

**Broader Impact:** By partnering with Google we will apply our methods to a challenging real-world problem of recognizing text in photos (the PhotoOCR problem), and release the core algorithms and their implementations as open source packages. Our partnership with Google will allow graduate students involved in this project to gain valuable hands on experience with real life problems via internships. Undergraduate students will be offered research opportunities throughout the duration of the project. The PIs are active in organizing workshops at international conferences and educational events such as the Machine Learning Summer School. We will use these fora to publicize our results and attract bright graduate students to further our research agenda.

**Keywords:** Machine Learning; Logistic Regression; Probabilistic Models; Conditional Random Fields; Boosting; Quasi-Convex Functions.
1 Motivation and Goals

Models based on the exponential family of distributions have been studied for decades in machine learning and statistics [1, 15]. Distributions from the exponential family also arise naturally in another context namely, as solutions to the problem of maximizing the Shannon Boltzmann Gibbs (SBG) entropy subject to linear constraints [12, 13]. Since the SBG entropy cannot adequately capture all physical phenomenon, a number of generalizations of the SBG entropy have been proposed in statistical physics. Recently, there has been an effort to unify these disparate entropies by using generalized exp and log functions [27, 28, 30]. The so-called $t$-logarithm and $t$-exponential functions, which are inverses of each other, are defined using a parameter $0 < t \leq 2$ (see Figure 2):

$$
\log_t(x) = \left( \frac{x^{1-t} - 1}{1-t} \right)^{1/(1-t)}.
$$

$$
\exp_t(x) = \left[ \max(0, 1 + (1-t)x) \right]^{1/(1-t)}.
$$

Note that in general $\exp_t(a + b) \neq \exp_t(a) \cdot \exp_t(b)$ and $\log_t(1/a) \neq -\log_t(a)$. The usual log, exp, and functions are recovered in the limit as $t \to 1$.

One can define a new family of distributions by replacing the exp in the exponential family by the more general $\exp_t$ function [28, 37]. In this proposal we aim to build conditional models based upon the $t$-exponential family and study their algorithmic and theoretical properties. To illustrate some of the possibilities, we start with logistic regression, where the loss of an example $(x, y) \in \mathbb{R}^n \times \{0, 1\}$ is defined as $\ln(1 + \exp(x \cdot \theta)) - y x \cdot \theta$. Here $\theta \in \mathbb{R}^n$ is the current parameter or weight vector. This loss is the negative logarithm of the distribution $p(y | x, \theta) = \exp(y x \cdot \theta - g(\theta | x))$. Clearly the sum of this distribution over the label domain $y \in \{0, 1\}$ is one. The most common class of distributions studied in statistics is the exponential family of distributions. In the case of classification, this family distributions always is the exponential of a linear term $y x \cdot \theta$ minus a log-partition function $g(\theta | x)$ that assures the distribution is normalized when summing over both labels, i.e.,

$$
p(y | x, \theta) = \exp(y x \cdot \theta - g(\theta | x)).
$$

Logistic regression clearly belongs to this family. In this case the log-partition function is $g(\theta | x) = \ln(1 + \exp(\theta \cdot x))$. In the $t$-exponential family, the exponential is replaced by the $t$-exponential, i.e.,

$$
p_t(y | x, \theta) := \exp_t(y x \cdot \theta - g_t(\theta | x)).
$$

The partition function $g_t(\theta | x)$ again assures that the two label probabilities sum to one, i.e.,

$$
\exp_t(\theta \cdot x - g_t(\theta | x)) + \exp_t(-\theta \cdot x - g_t(\theta | x)) = 1.
$$

Unfortunately, $g_t(\theta | x)$ does not have a closed form any more (except for $t = 2$). Instead, we have to work with the above implicit equation. Furthermore, the negative log-likelihood $-\log p_t(y | \theta, x)$ is no longer convex in the parameter vector $\theta$. Note that we used the standard logarithm and not
the inverse log\textsubscript{\textsubscript{\textcircled{\textit{t}}}} of exp\textsubscript{\textsubscript{\textcircled{\textit{t}}}} to define this loss because adding the negative log-likelihood corresponds to assuming independence between examples. We call $-\log p_t(y|\theta, x)$ the $t$-logistic loss (in Figure 1 it is plotted for various values of $t$). Clearly, the $t$-logistic loss is quasi-convex and plateaus out for points with large negative margin. An advantage of this plateauing is that the “influence” of outliers (points with large negative margin) on the final solution is minimized. This gives rise to models which are robust to label noise.

One might wonder if robustness to outliers can be obtained by using some other convex loss. After all, convexity has become the holy grail of Machine Learning because minimizing the sum of convex loss functions leads to globally optimal solutions. However, the recent results of [23] rule out this possibility. Concretely, [23] show that any learning algorithm based on convex loss functions is not robust to noise, and therefore one has to move away from convex loss functions to design robust classifiers. However, this is problematic because summing quasi-convex loss functions gives rise to non-convex objective functions which suffer from the problem of multiple local minima. Somewhat surprisingly, we [5] empirically observe on a wide variety of datasets that the objective function of $t$-logistic regression does not suffer from this problem. Not only is the objective function smooth and continuous, but also seems to exhibit a single globally optimal solution. While we do not have a formal proof yet, we will present some compelling visual intuition as to why this is the case in Section 4.1.

Above we saw the simple case of binary logistic regression. More generally, conditional densities from the exponential family are defined as [1]

$$p(y|x; \theta) := \exp \left( \langle \Phi(x, y), \theta \rangle - g(\theta | x) \right) = \frac{1}{Z(\theta | x)} \exp \left( \langle \Phi(x, y), \theta \rangle \right).$$

Here, we used a joint feature map $\Phi(x, y)$ to define the density, and $\langle \cdot, \cdot \rangle$ denotes the Euclidean dot product. For binary logistic regression discussed above we used a linear feature map \textit{i.e.}, in this case $\Phi(x, y)$ equals $y x$. Fancier models (\textit{e.g.} for multiclass classification) are obtained by using other feature maps $\Phi(x, y)$. The $t$-exponential counterpart of the above conditional density is obtained by replacing exp with exp\textsubscript{\textsubscript{\textcircled{\textit{t}}}} [28]:

$$p(y|x; \theta) := \exp_t \left( \langle \Phi(x, y), \theta \rangle - g_t(\theta) \right).$$

Arguably, the major challenge in using the $t$-exponential families is in being able to efficient compute the log-partition function $g_t$. To address this issue for multiclass classification, where the number of classes is fairly small, we will develop exact iterative algorithms. On the other hand, for problems such as sequence classification where the number of classes is exponentially large, we will develop...
approximate inference techniques by extending variational methods. We will also explore models that drop the normalization constraint. This sidesteps the issue of computing the log-partition function.

2 Connections to Existing Work

The robustness of the models built using the $t$-exponential family can also be understood by visualizing the $\exp_t$ function for various values of $t$ (see Figure 2). Observe that for values of $t > 1$ $\exp_t$ tends to approach zero slowly, and this leads to “heavy-tailed” distributions. It is well known that heavy-tailed distributions lead to robust models. For instance, there is a rich tradition in statistics of using the Student’s-$t$ distribution instead of the Gaussian distribution to obtain robust models (see e.g. [10, 11, 22, 32, 34, 40] and many more). Working with the $t$-exponential family, which includes the Student’s-$t$ distribution as a special case, not only allows us to beyond what has been done in robust statistics, but also facilitates a unified analysis. The foundational literature about generalized exponential families appeared in statistical physics [8, 27–30, 43, 44] and statistics [9, 19]. Our contribution is in applying these ideas to machine learning problems.

Figure 2: Left: $\exp_t(x) := \left[\max (0, 1 + (1 - t)x)\right]^{1/(1-t)}$ for various values of $t$ indicated. Right: The shaded area in the left figure is zoomed in to better depict the tail behavior.

3 Our Recent Relevant Work

3.1 Establishing Connections

Tim Sears was a student of PI Vishwanathan and our initial work on designing probabilistic models using generalizations of the exponential family appeared in his Ph.D. thesis [37]. We introduced generalized relative entropies to the machine learning community and showed they arise as solutions to a generalized maximum entropy (maxent) problem. In fact, we showed that this problem has been studied under various names in different fields such as convex analysis [e.g. 2, 31], and information

\[\text{Conversely, for } t < 1, \exp_t \text{ can reach the value zero, and this leads to “truncated” distributions.}\]
geometry [e.g. 6, 26]. In each case we showed that the main theorems in these papers can be reconstructed simply by using basic concepts from convex analysis along with Fenchel duality.

Independently, Sears applied $t$-entropy which is defined using the log $t$ function$^2$ to the problem of non-negative matrix factorization (NMF). Results were presented on the ORL face data set$^3$ and the THOMAS database of legislative information$^4$. Some preliminary investigation on generalizing the Hammersley-Clifford theorem to different notions of decomposability and the corresponding generalized-Markov property were also presented (also see [38]).

### 3.2 $t$-Logistic Regression for Binary Classification

We recently explored the use of $t$-exponential families for binary logistic regression [5]. In order to tackle the non-convex optimization problem that arises, we note that adding the negative log-likelihood corresponds to assuming independence between the data points. Therefore, one can equivalently rewrite the objective function of $t$-logistic regression as the product of likelihoods:

$$\prod_{i=1}^{m} p(y_i | x_i; \theta) = \prod_{i=1}^{m} \exp_t \left( y_i \theta \cdot x_i - g_t(\theta | x_i) \right).$$

(2)

Products of convex functions are not convex, but have certain well defined analytical properties which makes optimizing them tractable. One way to rewrite the above objective function is to introduce non-negative variables $\xi_i$, which can be thought of as “influence” variables [17]

$$\min_{\theta, \xi} \sum_{i=1}^{m} \xi_i z_i(\theta), \quad \text{s.t.} \quad \xi > 0, \quad \prod_{i=1}^{m} \xi_i \geq 1,$$

(3)

where $z_i(\theta) = (1 + (1 - t)(y_i \theta \cdot x_i - g_t(\theta | x_i)))$. Analysis of the KKT conditions shows that at the optimal solution $z_i(\theta) \xi_i = \text{constant}$. If a point has large negative margin, then its corresponding $z_i(\theta)$ is large and consequently its influence $\xi_i$ is small. Conversely, points which are well classified have small $z_i(\theta)$ and their influence $\xi_i$ is large.

**Empirical Evaluation** To demonstrate how our algorithm performs empirically we consider the Mease-Wyner dataset$^5$, a synthetic dataset to test the effect of label noise [24]. As Figure 3 shows classical algorithms such as logistic regression and SVM break down in the presence of label noise. Even an algorithm which employs the non-convex probit loss, $L(u) = 1 - erf(2u)$, used in [7] is prone to the same problem. In contrast, $t$-logistic regression (especially with $t = 1.9$) is able to handle label noise and shows statistically significant improvement over the other algorithms. The noise tolerance of our algorithm can be explained by the $\xi$ coefficients which denote the influence of a point. We plot them in Figure 3 (right). One can observe that the $\xi$ of the noisy data is much smaller than that of the clean data, which indicates that the algorithm is able to effectively identify these points and cap their influence.

---

$^2$In [37] it is called the $q$-entropy. We use $t$-entropy to maintain notation compatibility with our later work.

$^3$http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

$^4$http://thomas.loc.gov/home/rollcallvotes.html

$^5$Results on a number of other publicly available datasets are similar and can be found in [5].
Figure 3: Left: The test error rate of various algorithms on the Mease-Wyner dataset with and without 10% label noise. We use 10-fold cross validation. Right: The distribution of $\xi$ obtained after training $t$-logistic regression with $t=1.9$ on the Mease-Wyner dataset with 10% label noise.

4 Proposed Research

4.1 Theoretical Considerations

Even though $t$-logistic regression employs a quasi-convex loss, the objective function empirically exhibits a single global minima. It is therefore important to establish this property of $t$-logistic regression rigorously. We now present some visual intuition to show why we believe this is possible. We used a two dimensional toy dataset which contains 50 points drawn uniformly from $[-2, 2] \times [-2, 2]$. Since the data is two dimensional, the algorithm has only two parameters. The objective function is plotted as a function of these parameters in Figure 4. For comparison we also plot the objective function on the same dataset obtained using the probit loss used in [7].

As can be seen, the probit loss yields a highly non-convex objective function with a large number of local optima$^6$. In contrast, even though we are averaging over quasi-convex loss functions the resulting function of $t$-logistic regression is smooth, with a single global optimum. This behavior persists when we use different random samples, change the sampling scheme, or vary the number of data points. Moving over to higher dimensional datasets such as Adult, USPS, and Web$^7$, we initialize the algorithm with different randomly chosen starting points and checked the solution obtained. It always arrives at the same solution (within numerical precision), that is, the single global minima of the objective function.

$^6$The probit loss has two tuning parameters, namely the mean and the variance of the Gaussian distribution used to compute the $erf(\cdot)$. For all reasonable values of these parameters, the local optima occurs.

$^7$All available from http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.
One Dimensional Case  If we restrict ourselves to one dimension, then we can visualize the gradient of the objective function. See Figure 5, where we depict the situation for a specific set of examples\textsuperscript{8}. Observe that the gradient of the objective function crosses zero at exactly one point, which implies that the function attains a unique global minimum. Our line of attack will be to formalize this observation and prove that the one dimensional objective function always has a globally optimal solution. To extend the result to higher dimensions, note that minimizing our objective function is equivalent to minimizing a product of $m$ convex functions (2). Although not convex, there are schemes for algorithmically finding all local minima [17]. Even though these schemes are not algorithmically attractive because they scale exponentially in $m$, they provide valuable insights into the properties of the objective function. In particular, we will study the one dimensional projections of such functions in order to lift our one dimensional results to higher dimensions.

\textsuperscript{8}Identical behavior is observed for other sets of examples.
Table 1: Average time (in milliseconds) spent by our iterative scheme for solving (6).

<table>
<thead>
<tr>
<th>$k$</th>
<th>10</th>
<th>20</th>
<th>30</th>
<th>40</th>
<th>50</th>
<th>60</th>
<th>70</th>
<th>80</th>
<th>90</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>iterative</td>
<td>0.3</td>
<td>0.3</td>
<td>0.3</td>
<td>0.4</td>
<td>0.4</td>
<td>0.4</td>
<td>0.3</td>
<td>0.3</td>
<td>0.4</td>
<td>0.5</td>
</tr>
</tbody>
</table>

4.2 $t$-logistic Regression for Multiclass Classification

In the case of binary classification, $y \in \{0, 1\}$ and therefore log-partition function $g_t$ can be computed by solving the implicit equation (1). Furthermore, since $g_t$ is smooth and convex, it can be approximated with very high numerical precision using the method of splines. For multiclass problems the labels $y$ are drawn from $\{1, \ldots, k\}$, and the implicit equation (1) now becomes

$$\sum_{i=1}^{k} \exp_t \left( \langle \Phi(x, y), \theta \rangle - g_t(\theta | x) \right) = 1.$$  (4)

If we can compute $g_t$ efficiently, then the resulting multiclass classifier will share the same robustness properties of binary $t$-logistic regression. Solving for $g_t$ using standard root finding routines such as `fsolve` in MATLAB is possible, but excruciatingly slow. Instead, we propose to use an iterative scheme. For this, we show that there exists a $\tilde{\theta}$ and a function $Z(\tilde{\theta} | x)$ such that

$$Z(\tilde{\theta} | x) = \sum_i \exp_t \left( \langle \Phi(x, y), \tilde{\theta} \rangle \right)$$

and

$$\exp_t \left( \langle \Phi(x, y), \theta \rangle - g_t(\theta | x) \right) = \frac{1}{Z(\tilde{\theta} | x)} \exp_t \left( \langle \Phi(x, y), \tilde{\theta} \rangle \right).$$  (5)

Instead of trying to compute $g_t$ we try to iteratively estimate $\tilde{\theta}$ and $Z(\tilde{\theta} | x)$. Initial experiments show that $g_t$ can be obtained with high accuracy in approximately 5-10 iterations.

To illustrate, consider the following toy example: Let $x = 1$ and $\langle \Phi(x, y), \theta \rangle = \theta_i$, in which case (4) simplifies to

$$\sum_{i=1}^{k} \exp_t (\theta_i - g_t(\theta)) = 1.$$  (6)

We let $k \in \{10, 20, \ldots, 100\}$, and for each $k$ we randomly generate $\theta \in [-10, 10]^k$, and compute the corresponding $g_t(\theta)$. We compare the time spent in estimating $g_t(\theta)$ by the iterative scheme averaged over 100 runs and present the results in Table 1. Our method scales almost constantly with $k$ the number of classes, thus making it useful for problems involving a large number of classes.

PhotoOCR  PI Vishwanathan has an ongoing collaboration with an image recognition team at Google that is working toward solving the so-called PhotoOCR problem (see letter of collaboration). The data consists of approximately 150,000 images that are obtained from smart phones, and the problem is to recognize characters in these photos. While optical character recognition (OCR) systems have been mostly perfected for high-quality images obtained by scanning documents under carefully controlled conditions, the results obtained from existing commercial OCR packages when
applied to photos are far from satisfactory. Many factors make OCR on photos (PhotoOCR) significantly more difficult than OCR on scanned documents: low resolution, blur, occlusion, poor light conditions, arbitrary angles from which the image is captured, etc. In a typical application pipeline, the data is labeled semi-automatically. The photos are then processed by a segmenter in order to detect individual characters in the images. The sub-images containing individual characters are then passed on to a feature extractor and a classifier. However, the training data that is available to the classifier is noisy because of the errors in labeling and segmentation phases.

In our preliminary experiments we posed this as an one-vs-rest binary classification problem, and used binary $t$-logistic regression. This resulted in approximately 5.5% improvement in overall classification accuracy as compared to the non-robust method that is in use internally at Google. Further gains in classification accuracy can be obtained by replacing binary $t$-logistic regression with multiclass $t$-logistic regression. However, this will require developing better optimization procedures. This is primarily because the application involves a large number of classes (26 characters, 10 digits, and a number of special symbols) as well as a large number of training data points. However, we can show that objective function that is minimized by multiclass $t$-logistic regression is biconvex. This endows the problem with significant structure, that can be exploited to design efficient optimizers. PI Vishwanathan has significant experience in designing optimizers for large scale learning problems [42, 45, 46, 48]. Building upon this expertise we will design practically efficient optimization algorithms. Google is committing resources to help with this endeavor. By leveraging our partnership with Google we will not only make an important contribution to a practical application of immense interest to industry, but also release the core algorithms and their implementations under a permissive open source license for wider use by the research community.

**Beyond Multiclass Classification** A large number of commonly used loss functions in machine learning can be viewed as generalizations of the hinge loss, $l(x, y, \theta) = \max(0, 1 - \langle \Phi(x, y), \theta \rangle)$, used in Support Vector Machines. Examples include the loss functions used in novelty detection, ordinal regression, and multi-label classification. We have worked with these loss functions in the past [3], and presented a unifying view [42]. If we replace $\max(\cdot)$ in the loss by the softmax function, $\log \sum \exp(\cdot)$, then the loss functions can be interpreted as the log-likelihood in a conditional exponential family [42]. We will generalize these loss functions to the conditional $t$-exponential family and derive efficient training algorithms. We conjecture that (5) holds whenever the label space $\mathcal{Y}$ is finite. Verifying this conjecture will readily yield an iterative scheme for computing the log-partition function and lead to robust classifiers for a large number of problems.

### 4.3 $t$-Maximum Entropy Markov Model ($t$-MEMM)

In many practical applications, we are given a sequence of observations $X = \{x_1, \ldots, x_n\} \in \mathbb{R}^d$. The general problem is to infer a sequence of discrete hidden states $Y = \{y_1, \ldots, y_n\} \in \{1, \ldots, k\}^n$. One approach to solve this problem is to use a maximum entropy Markov model (MEMM), which is a graphical model that combines features of hidden Markov models (HMMs) and Maximum Entropy (maxent) models. Unlike a standard maxent classifier which assumes that the $y_i$’s are conditionally independent of each other, MEMMs make a standard Markov assumption: $p(y_t|y_1, \ldots, y_{t-1}) = p(y_t|y_{t-1})$. As such, the MEMM is fully specified by two parameters namely the $k \times k$ (unnormalized)
transition matrix $A$ and a parameter vector $\theta$. Given these, we can write

$$p(Y \mid X, A, \theta) = p(y_1 \mid x_1, \theta) \prod_{s=2}^{n} p(y_s \mid x_s, y_{s-1}, A, \theta)$$

where,

$$p(y_1 = y \mid x_1, \theta) = \exp(\langle \Phi(x_1, y), \theta \rangle - g_t(\theta \mid x_1))$$

$$p(y_s = y \mid x_s, y_{s-1} = y', A, \theta) = \exp(\langle \Phi(x_s, y), \theta \rangle + a_{y'y} - g_t(\theta, A \mid x_s, y')).$$

In order to estimate $A$ and $\theta$ given $X$ and $Y$, we minimize the negative entropy of $p(Y \mid X, A, \theta)$, which can be written as

$$-H(\theta, A) := \sum_y q(y) \log q(y) + \sum_{s=2}^{n} \sum_{y,y'} q_s(y, y') \log p(y_s = y \mid x_s, y_{s-1} = y', A, \theta).$$

where we used the shorthand $q(y) = p(y_1 = y \mid X, A, \theta)$ and $q_s(y, y') = p(y_s = y, y_{s-1} = y' \mid X, A, \theta)$. If we define $\alpha_s(y) = p(y_s = y \mid X, A, \theta)$ then $q_s(y, y') = \alpha_s(y')p(y_s = y \mid x_s, y_{s-1} = y', A, \theta)$ while $\alpha_s$ can be computed recursively using

$$\alpha_s(y) = \sum_{y'} \alpha_{s-1}(y')p(y_s = y \mid x_s, y_{s-1} = y', A, \theta) \sum_{y'} q(y, y').$$

We propose to build an MEMM model based on $t$-exponential family by replacing $p(y_1 = y \mid x_1, \theta)$ and $p(y_k = i \mid x_k, y_{k-1} = j, A, \theta)$ with distributions from the $t$-exponential family.

$$p(y_1 = y \mid x_1, \theta) = \exp_t(\langle \Phi(x_1, y), \theta \rangle - g_t(\theta \mid x_1))$$

$$p(y_s = y \mid x_s, y_{s-1} = y', A, \theta) = \exp_t(\langle \Phi(x_s, y), \theta \rangle + a_{y'y} - g_t(\theta, A \mid x_s, y')).$$

The objective function (7) in our case becomes

$$-H(\theta, A) := \sum_y q(y) \log q(y) + \sum_{s=2}^{n} \sum_{y,y'} q_s(y, y') \log \exp_t(\langle \Phi(x_s, y), \theta \rangle + a_{y'y} - g_t(\theta, A \mid x_s, y')).$$

The normalization constants $g_t(\theta \mid x_1)$ and $g_t(\theta, A \mid x_s, y')$ can be computed by solving the following implicit equations, both of which involve a summation of $k$ terms.

$$1 = \sum_y \exp_t(\langle \Phi(x_1, y), \theta \rangle - g_t(\theta \mid x_1))$$

$$1 = \sum_y \exp_t(\langle \Phi(x_s, y), \theta \rangle + a_{y'y} - g_t(\theta, A \mid x_s, y')).$$

The iterative techniques proposed for the multiclass case can be used to compute the above normalization constants.

**Label Bias Problem** MEMMs suffer from the well known label-bias problem [20, 41]. This is usually explained as an artifact of the per-state normalization that is enforced by requiring $\sum_y p(y_s = y \mid x_s, y_{s-1} = y', A, \theta) = 1$. A factor that might play an important role is the convexity of the log-likelihood of the exponential family used to model $p(y_s \mid y_{s-1})$. Using a $t$-exponential
family distribution means that the log-likelihood is quasi-convex; if the chain contains mislabeled observations, then their “influence” will be capped. Furthermore, the distributions can either be heavy tailed or truncated. We will carefully study both theoretically and empirically if the label-bias problem can be overcome by using distributions from the \( t \)-exponential family. Our findings will also be important to understand the role that the \( \xi \) coefficients play in such models.

4.4 \( t \)-Conditional Random Fields (\( t \)-CRFs)

Compared to MEMMs which enforce a per-state normalization, Conditional Random Fields (CRFs) enforce global normalization \([20, 41]\). In the case of a simple chain structured model \(^9\)

\[
p(Y | X, \theta) = \exp \left( \sum_{i=2}^{n} \langle \Phi(y_i, y_{i-1}, X), \theta \rangle - g(\theta | X) \right). \tag{8}\]

The log-partition function \( g \) of the above model can be computed by a single pass of the forward-backward algorithm \([41]\).

**The Key Difficulty** If we replace the \( \exp \) in (8) by \( \exp_t \), then one can no longer use the forward-backward algorithm to compute the log-partition function \( g_t \). Intuitively, this is because \( \exp(a+b) = \exp(a) \cdot \exp(b) \) while \( \exp_t(a+b) \neq \exp_t(a) \cdot \exp_t(b) \). For the very same reason, inference techniques such as loopy belief propagation which exploit the factorization properties of the \( \exp \) function are also not admissible. To understand this further, note that a distribution from the exponential family which factorizes over the cliques of a graph corresponds to conditional independence between random variables \([14]\). This is a consequence of the Hammersely-Clifford theorem, and underlies the development of undirected graphical models. Efficient inference algorithms usually exploit this factorization property in order to compute the log-partition function. If one moves to the \( t \)-exponential family this correspondence between conditional independence and cliques of a graph is lost. Therefore, only a very small number of existing inference schemes can even be considered as candidates for extension to the \( t \)-exponential family.

**Exploiting Duality** Variational methods, on the other hand, take a completely different approach to the problem of computing the log-partition function. They rely on the convexity of the log-partition function and the duality with the negative entropy \([47]\). While the convexity of the log-partition function was established in \([28]\), we have shown \([4]\) (also see \([8, 44]\)) that the following negative \( t \)-entropy

\[
-S_t(p) := \sum_{i=1}^{n} q_i \log_t p_i \tag{9}
\]

is the Fenchel dual of \( g_t \). This opens up the possibility that some of the variational inference methods such as structured mean field updates can be extended to the \( t \)-exponential family. This will allow us to work with \( t \)-CRFs. We will perform careful experiments to study the properties of \( t \)-CRFs and compare them with traditional CRFs.

\(^9\)Of course, more complicated graphical models are admissible. Our aim however is to merely illustrate the issues involved in generalizing these models to the \( t \)-exponential family. For this, the chain structured models suffice.
4.5 Relaxing the Normalization Constraints

Maximum Entropy (maxent) and Maximum likelihood estimation are dual problems [21]. Maximum likelihood models work with distributions which need to be normalized. In contrast, one can drop the normalization constraint from maxent problems and derive novel algorithms. Although this does not lead to probabilistic models in the strict sense, one can sidestep the computation of the log-partition function which can be advantageous in some cases. As the first step towards deriving these models, we partially dualize $t$-logistic regression to derive the following maxent problem:

$$\max_{p, \xi} \left\{ -\sum_{i=1}^{m} \xi_i \sum_y p^t(y|x_i) \log_t p(y|x_i) \right\}$$

s.t. $$\sum_{i=1}^{m} \xi_i \mathbb{E}_{q(y|x_i)}[\Phi(x_i, y)] = \sum_{i=1}^{m} \xi_i \Phi(x_i, y_i)$$ and $\sum_y p(y|x_i) = 1$

$$\xi_i \geq 0, \prod_{i=1}^{m} \xi_i \geq 1,$$

where $q(y|x) = p(y|x)^t / \sum_y p(y|x)^t$ is the escort distribution, and $\xi_i$ determines the influence of a point $x_i$. One can drop the normalization constraints from the above problem and work with:

$$\max_{p, \xi} \left\{ -\sum_{i=1}^{m} \xi_i \sum_y (p^t(y|x_i) \log_t p(y|x_i) - p(y|x_i)) \right\}$$

s.t. $$\sum_{i=1}^{m} \xi_i \mathbb{E}_{q(y|x_i)}[\Phi(x_i, y)] = \sum_{i=1}^{m} \xi_i \Phi(x_i, y_i)$$

$$\xi_i \geq 0, \prod_{i=1}^{m} \xi_i \geq 1.$$ 

Note that since $p$ is no longer normalized, we need to add $\sum_i p(y_i|x_i)$ to the objective function to avoid degenerate solutions. Passing again to the dual with $\xi$ gives us

$$\min_{\theta, \xi} \sum_{i=1}^{m} \xi_i (1 + (1 - t) \exp_t (-\langle \Phi(x_i, y_i), \theta \rangle))$$

s.t. $$\xi_i \geq 0, \prod_{i=1}^{m} \xi_i \geq 1.$$ 

As [21] show, dualizing classical maxent after dropping the normalization constraints yields AdaBoost. Heavily inspired by AdaBoost, we introduce a distribution $d$ on the training points and propose the following iterative scheme to minimize (10):

**Algorithm 1: t-boosting**

1: repeat
2: \textbf{ξ-Step:} Update $\xi$ such that $\xi_i \propto (1 + (1 - t) \exp_t (-\langle \Phi(x_i, y_i), \theta \rangle))^{-1}$;
3: \textbf{θ-Step:} Update $\theta$ such that $\sum_i d_i \xi_i \exp_t (-\langle \Phi(x_i, y_i), \theta \rangle) \Phi(x_i, y_i) = 0$;
4: \textbf{d-Step:} Update $d$ such that $d_i \propto d_i \exp_t (-\langle \Phi(x_i, y_i), \theta \rangle)$.
5: until convergence
**Noise Tolerance** Just like AdaBoost, the weight $d_i$ assigned by our algorithm to the $i$-th data point depends on the negative margin $-\langle \Phi(x_i, y_i), \theta \rangle$. However, when $0 < t < 1$, the influence $\xi_i$ that is assigned to the $i$-th point is inversely proportional to its negative margin. The product of $d_i$ and $\xi_i$ determines the importance of a point in the $\theta$ step. This, we conjecture, fixes one of the well known weaknesses of AdaBoost namely intolerance to noise [35]. AdaBoost is particularly prone to this problem because whenever an outlier is misclassified, the weight $d_i$ assigned to it is very high. In other words, AdaBoost never “gives up” on extremely hard points, that is, the outliers. In our case, even if the $d_i$ is very large, the corresponding $\xi_i$ will be small thus counterbalancing the impact of outliers.

**Iteration Bounds** An attractive property of Boosting algorithms is that they converge to an $\epsilon$ accurate solution within $O(\log n/\epsilon^2)$ iterations. Since existing Boosting algorithms are based on the SBG entropy, such bounds are established by using the strong convexity of the SBG entropy with respect to the $\ell_1$ norm [39]. This gives rise to Pinkser’s inequality, which in turn is used to show that sufficient progress is made at every iteration. To develop a similar reasoning, we will establish the strong convexity of the $t$-entropy and derive a generalization of the Pinkser’s inequality.

## 5 Longer Range Challenges

**Kernels** One can incorporate kernels into our framework, by replacing the dot product $\langle \cdot, \cdot \rangle$ by a kernel $k(\cdot, \cdot)$ in a Reproducing Kernel Hilbert Space (RKHS). Kernels lead to a much richer class of models, and allow us to incorporate prior knowledge in a flexible manner [36]. The downside however is that optimization can no longer be carried out directly in an RKHS. To overcome this problem, we will develop an analog of the celebrated representer theorem [16] and show that a small number of coefficients are sufficient to recover the model parameters. This will allow us to derive a finite dimensional dual problem like in the case of Support Vector Machines. To optimize the dual we will devise an efficient coordinate descent procedure along the lines of the SMO (sequential minimal optimization) algorithm of Platt [33].

**$\phi$-Exponential Family** One can generalize the exp and the log functions beyond exp$_t$ and log$_t$ and define entropies based on them. exp$_\phi$ is the inverse to log$_\phi$ which in turn is defined by using a strictly positive and non-decreasing function $\phi : [0, +\infty) \rightarrow [0, +\infty)$ and letting [27, 29, 30]

$$
\log_\phi(x) := \int_1^x \frac{1}{\phi(y)} \, dy.
$$

Using $\phi(x) = x$ recovers the familiar log and $\phi(x) = x^t$ recovers the log$_t$ function.

Working with $\phi$-exponential families, which are defined using the exp$_\phi$ function, is more challenging. This is primarily because some of the algebraic transformations which allow us to transform the objective function of $t$-logistic regression into the form (3) do not directly apply. Furthermore, existing results do not establish a duality relationship between the log-partition function and the $\phi$-entropy. In fact, some preliminary results that we obtained [4] show that the $\phi$-entropy as defined by [28] is not the Fenchel dual of the log-partition function. Therefore, completely new techniques need to be developed in order to perform inference in these models.
6 Broader Impacts

We will make probabilistic models based on generalized exponential families practical. Our models will be robust to noise and yet have attractive theoretical properties. Such noise tolerant models can find applications in a number of disciplines, including finance, econometrics, astrophysics, imaging, philosophy, natural language processing, and species modeling. Although our core emphasis is on the underlying algorithms, we firmly believe that our results are likely to impact all these application areas. We will also train a number of students in this new area.

Graduate Classes The PIs regularly teach introductory courses on statistical machine learning wherein concepts such as logistic regression, maxent, and graphical models are routinely covered. The research proposed here can be viewed as a logical extension of these classical concepts, and can therefore be easily incorporated into the curriculum. As part of their course projects, students will be offered a chance to work on small sub-problems based on the research proposed here. This will not only pique their interest but also help attract bright graduate students who would like to work on this project.

Workshops and Summer School The PIs are active in organizing workshops at leading venues such as the annual Neural Information Processing Systems (NIPS) conference. To publicize the proposed research and to receive feedback from the machine learning community we will organize workshops at leading international conferences. Furthermore, we will deliver mini-courses on this topic at the Machine Learning Summer Schools\(^\text{10}\) (MLSS). These summer schools are a series of two week intensive courses designed to introduce, amongst others, advanced undergraduates and graduate students to core concepts in machine learning. While both PIs have lectured at the past summer schools, PI Vishwanathan has also organized an MLSS in Canberra, Australia in 2005, and will organize a MLSS at Purdue in 2011.

Open-Source Software We will build upon open source code we have released in the past including efficient implementations of a bundle method solver (BMRM), quasi-Newton Solver (subBFGS) and Entropy Regularized LPBoost (ERLPboost). All code developed as a part of this project will be released under a permissive open source license on the Machine Learning and Open Source Software (MLOSS) website http://mloss.org. An open development model will be adapted with the aim of building a vibrant community of developers and users. Impact will be measured in terms of the number of downloads. PI Vishwanathan has experience in software design and architecture, having worked as a commercial software developer for three years.

Enhancing Research Training The ideas in this proposal have significant potential to be applied to real-world problems. As such, we have strong support from collaborators in Google. Our partnership with Google will give graduate students on this project access to resources such as large datasets, real-world problems, and large clusters of computers. Furthermore, they will get a chance to take up competitive internships at Google. This experience will enrich their research training and make them better prepared to become future knowledge workers.

\(^{10}\)http://mlss.cc
Involving Undergraduate Students in Research  During the summer we will employ an undergraduate student to work on specific sub-tasks under the guidance of the PIs and the graduate students involved in the project. We will also use this opportunity to recruit women and underrepresented minorities. This experience is particularly useful for students who are considering graduate school, because it provides them with research experience and makes them competitive applicants for admission to a graduate program. The graduate students, on the other hand, learn how to supervise students, and this experience is useful if they decide to pursue an academic career in the future.

7  Project Plan

The two members of the team have complementary expertise, which is crucial for the success of the project. Prof. Warmuth is widely cited for his work on computational learning theory, while Dr. Vishwanathan brings to the table extensive experience in structured prediction and optimization. Both PIs already have done over half a decade of collaborative research. Furthermore, we will frequently travel and visit each other (at least once a year) to collaborate. To ensure effective supervision, the PIs will serve as committee members for students funded by this project. Furthermore, students will be encouraged to split their time between Purdue University and UCSC.

The research outlined in this proposal will be carried out by the PIs and two graduate students. One graduate student will mainly focus on the theoretical aspects of the project while the other one will work on algorithmic and implementation details. We break down the various tasks and estimate the time needed to complete them in the chart below. Shading indicates dependencies between tasks.

- **A1** Multiclass $t$-logistic regression (section 4.2)
- **A2** Application to PhotoOCR (section 4.2)
- **A3** Extensions to multi-label and other problems (section 4.2)
- **A4** $t$-MEMMs (section 4.3)
- **A5** $t$-CRFs (section 4.4)
- **A6** $t$-Boosting (section 4.5)
- **T1** Unique global minima of the objective function (section 4.1)
- **T2** Efficiently computing log-partition function for $t$-CRFs (section 4.4)
- **T3** Extensions to kernels (section 5)
• T4 Extensions to the $\phi$-exponential family (section 5)
• T5 Convergence theory for $t$-Boosting (section 4.5)

8 Prior NSF Funding

PI Warmuth was supported by an ITR grant (Co-Pis: Kearns, Littmann and Schapire): “Representation and Learning in Computational Game Theory - NSF grant IIS 0325363”. The grant ended in Summer 2008. Essentially all of PI Warmuth’s work on recent boosting algorithms culminating in Entropy Regularized LPBoost were done with the support of this grant. Many additional findings discovered under this grant were not discussed in this proposal, such as Warmuth’s extensive work on compression schemes (see e.g. [18]).

PI Vishwanathan was supported by an IIS grant (Co-PIs: Neville and Kirshner) “RI: Small: Algorithms for Generation of Similar Graphs Using Subgraph Signatures - NSF grant IIS 0916686” (2009 - 2010). We investigated Kronecker Product Graph generation models for modeling populations of graphs and showed that they cannot model natural variability in the data. We explained this limitation theoretically, and used the insights gained to propose a new variant which provably can model the variance found in the input graphs. This result has been accepted for publication at the Alletron Conference on Communications, Control, and Computing [25]. We are currently investigating generalizations of the above model, and Monte Carlo approximation techniques for inferring the parameters of our new models.

PI Warmuth is currently supported by an IIS grant “RI: Small: Kernelization with Outer Product Instances - NSF grant IIS 0917397” (2009-2011). So far the kernel algorithms, such as SVMs, have been mainly applied to the case when the instances are vectors. The goal of this research is to lift kernel methods to the matrix domain, where the instances are outer products of two feature vectors. The matrix parameters can model interactions between the features. We discovered that in the matrix setting a much larger class of algorithms based on any spectrally invariant regularization can be kernelized. One of the main goals of the grant is to show how to kernelize the matrix versions of the multiplicative updates. These updates are motivated by using the quantum relative entropy as a regularization. The goal is to prove generalization bounds for these updates that grow logarithmic in the feature dimension.
References


