Statistical Learning Algorithms Based on Bregman Distances

John D. Lafferty* Carnegie Mellon University

Stephen Della Pietra† Renaissance Technologies

Vincent Della Pietra† Renaissance Technologies

Abstract — We present a class of statistical learning algorithms formulated in terms of minimizing Bregman distances, a family of generalized entropy measures associated with convex functions. The inductive learning scheme is akin to growing a decision tree, with the Bregman distance filling the role of the impurity function in tree-based classifiers. Our approach is based on two components. In the feature selection step, each linear constraint in a pool of candidate features is evaluated by the reduction in Bregman distance that would result from adding it to the model. In the constraint satisfaction step, all of the parameters are adjusted to minimize the Bregman distance subject to the chosen constraints. We introduce a new iterative estimation algorithm for carrying out both the feature selection and constraint satisfaction steps, and outline a proof of the convergence of these algorithms.

1 Introduction

In this paper we present a class of statistical learning algorithms formulated in terms of minimizing *Bregman distances*, a class of generalized entropy measures associated with convex functions. The closest relatives of these algorithms are statistical methods for growing decision trees [2]. For decision trees, once a concave impurity function $f(p_1, \ldots, p_C)$ is chosen, the impurity of the tree is determined as $I_f(T) = \sum_t I_f(t) = \sum_t p(t) f(p(1\,|\,t), \ldots p(C\,|\,t))$, and a potential split q is evaluated according to the reduction in impurity that it yields: $\Delta I_f(q,t) = I_f(t) - I_f(t_L) - I_f(t_R)$. Popular impurity functions are the Shannon entropy $f(p_1, \ldots, p_C) = -\sum_j p_j \log p_j$ and the Gini criterion $f(p_1, \ldots, p_C) = \sum_{i \neq j} p_j p_i = 1 - \sum_j p_j^2$. We refer to [2] for an in-depth presentation of the art and science of growing decision trees.

In the algorithms for Bregman distances that we outline here, questions are formulated as linear constraints on the distribution to be inferred. Given a Bregman distance B_f and a distribution $\tilde{p}(x)$ on "histories" x, the distance between two measures $p(y\,|\,x)$ and $q(y\,|\,x)$ is taken to be

$$\bar{B}_f(p,q) = \sum_x \tilde{p}(x) B_f(p(\cdot\,|\,x), q(\cdot\,|\,x))$$

A procedure for approximating a reference distribution p is obtained by incrementally adding linear constraints, with each constraint evaluated according to the reduction in Bregman distance that it yields. This is analogous to the idea of greedily reducing the impurity of the classifier for decision trees, with the notable difference that the linear constraints do not in general form a partition of the event space.

The main technical result we announce here is the convergence and monotonicity of an iterative scaling algorithm for minimizing a Bregman distance subject to linear equality constraints, and we do not focus on the general or practical aspects of inference for Bregman distances. The details of this proof in the special case of the Kullback-Leibler divergence appear in [9]. The basic idea behind the algorithm is to make use of an auxiliary function which bounds the change in divergence from below after each iteration. This use of an auxiliary function is the standard means of justifying and implementing the EM algorithm [10], but the application of this technique to generalized maximum entropy problems appears to be new.

2 Bregman Distance

If f is a strictly convex real-valued function, the *f-entropy* of a discrete measure $p(x) \geq 0$ is defined by

$$H_f(p) = -\sum_x f(p(x))$$

and the Bregman distance $B_f(p,q)$ is given as

$$B_f(p,q) = \sum_x f(p(x)) - f(q(x)) - f'(q(x))(p(x) - q(x))\,.$$

When $f(x) = x \log x$, H_f is the Shannon entropy and $B_f(p,q)$ is the I-divergence, when $f(x) = -\log x$ we obtain the Burg entropy and discrete Itakura-Saito distortion

$$B_f(p,q) = \sum_x \left(\log \frac{q(x)}{p(x)} + \frac{p(x)}{q(x)} - 1 \right)$$

*School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA. E-mail: lafferty@cs.cmu.edu. Research supported in part by NSF and ARPA under grant IRI-9314969 and the ATR Interpreting Telecommunications Research Laboratories.

†Renaissance Technologies, 25 East Loop Road, Stony Brook, NY 11709, USA. E-mail: [sdella,vdella]@rentec.com. Part of this work was completed while the authors were with the IBM Watson Research Center in Yorktown Heights, NY. Research supported in part by ARPA under grant N00014-91-C-0135.

which arises in the spectral analysis of speech signals. When $f(x) = x^2$ we obtain the mean-squared distance

$$B_f(p, q) = \sum_x (p(x) - q(x))^2$$

A graphical representation of Bregman distance, as a measure of the convexity of f, is shown in Figure 1.

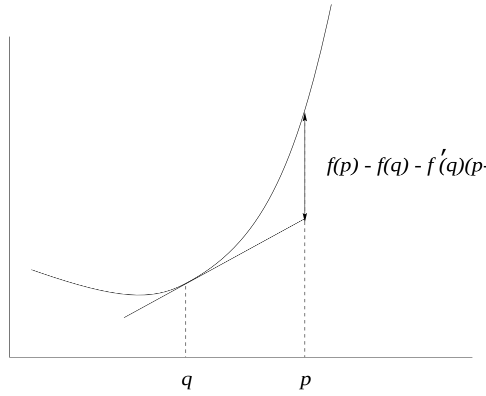

Figure 1: The Bregman distance $B_f(p, q)$ is an indication of the increase in $f(p)$ over $f(q)$ above linear growth with slope $f'(q)$.

These distances are naturally placed within continuous family of convex functions

$$f_\alpha(x) = \begin{cases} -x^\alpha + \alpha x - \alpha + 1 & \text{for } 0 < \alpha < 1 \\ x^\alpha - \alpha x + \alpha - 1 & \text{for } \alpha < 0 \\ x \log x - x + 1 & \text{if } \alpha = 1 \\ x - \log x - 1 & \text{if } \alpha = 0. \end{cases}$$

The significance of this family lies in Csiszár's results [6] showing that scale-invariant inference for linear inverse problems can be characterized in terms of minimizing a Bregman distance of the form B_{f_α}. A sample of these curves is shown graphically in Figure 2.

More generally, if $F : \mathbf{R}^r \longrightarrow \mathbf{R}$ is strictly convex and C^1, the Bregman distance $B_F(p, q)$ is defined as

$$B_F(p, q) = F(p) - F(q) - \nabla F(q) \cdot (p - q)$$

where $p = (p_1, \ldots, p_r)$ and $q = (q_1, \ldots, q_r)$. The distances B_F were introduced in [1] along with an interative algorithm for minimizing B_F subject to linear constraints. Another such algorithm is given in [4]. Other closely related "distances" associated with convex functions are Csiszár's *f-divergences*

$$D_f(p \,\|\, q) = \sum_x q(x) f\left(\frac{p(x)}{q(x)}\right)$$

We do not consider algorithms for these divergences in this paper, but note that they are investigated as error measures in backpropagation training of neural networks in [11].

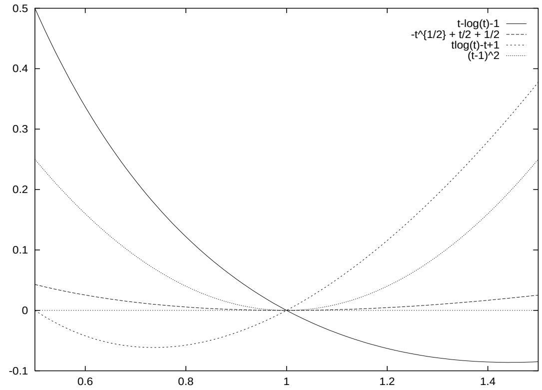

Figure 2: Four convex functions $f_\alpha(t)$ giving rise to Bregman distances: $f_2(t) = (t-1)^2$, $f_0 = t - \log(t) - 1$, $f_1 = t\log(t) - t + 1$, and $f_{\frac{1}{2}} = -t^{1/2} + t/2 + 1/2$, corresponding to the mean-squared distance, Itakura-Saito distortion, and Kullback-Leibler divergence, and the intermediate distance $B_{f_{\frac{1}{2}}}$, respectively.

3 The Basic Optimization Problem

Let $\Delta \subset \mathbf{R}^r$ and let $F : \Delta \to \mathbf{R}$ be a real-valued function. We assume that Δ is a closed convex set, and that F is strictly convex and C^1 on the interior of Δ. In this paper we will not be concerned with the technical details of allowing F to be infinite on Δ, but remark that these details can be carried through.

Definition 1 *For $v \in \mathbf{R}^r$ and $q \in \Delta$, define the Legendre transform $v \circ_F q$ by*

$$v \circ_F q = \operatorname*{arg\,min}_{p \in \Delta} B_F(p, q) + v \cdot p\,.$$

Lemma 2 *The map $(v, q) \mapsto v \circ_F q$ defines a smooth action of $\mathbf{R}^r$ on Δ by*

$$v \circ_F (w \circ_F q) = (v + w) \circ_F q\,.$$

Proof: Note that $v \circ_F q$ is uniquely determined by

$$\nabla F(v \circ_F q) = \nabla F(q) + v\,.$$

Hence

$$\begin{aligned} \nabla F(v \circ_F (w \circ_F q)) &= \nabla F(w \circ_F q) + v \\ &= \nabla F(q) + w + v \\ &= \nabla F((v + w) \circ_F q) \end{aligned}$$

so that this map is in fact a smooth action. $\square$

The optimization problem we consider is the following. Let A be an $n \times r$ matrix. This matrix will impose n linear constraints on $p \in \Delta$. Let $q_0 \in \Delta$ be a *default distribution*, chosen so that $\nabla F(q_0) = 0$. Finally, let $\tilde{p} \in \Delta$ be given; this will be thought of as the *empirical* *distribution*, since it typically arises from a set of training samples that determine the linear constraints.

We now define $\mathcal{P}(A, \tilde{p})$ and $\mathcal{Q}(A, q_0)$ by

$$\begin{aligned}\mathcal{P}(A,\tilde{p}) &= \{p \in \Delta \mid Ap = A\tilde{p}\} \\ \mathcal{Q}(A,q_0) &= \{q \in \Delta \mid q = (\lambda^\top A) \circ_F q_0,\ \lambda \in \mathbf{R}^n\}\end{aligned}$$

The following theorem is well-known, and establishes the duality between the two natural projections of $B_F(p,q)$ with respect to the families $\mathcal{P}(A,\tilde{p})$ and $\mathcal{Q}(A,q_0)$.

Theorem 3 *Suppose $B_F(\tilde{p}, q_0) < \infty$. Then there exists a unique $q_\star \in \Delta$ satisfying*

1. $q_\star \in \mathcal{P}(A,\tilde{p}) \cap \bar{\mathcal{Q}}(A,q_0)$
2. $B_F(p,q) = B_F(p,q_\star) + B_F(q_\star, q)$ *for any* $p \in \mathcal{P}(A,\tilde{p})$ *and* $q \in \bar{\mathcal{Q}}(A,q_0)$
3. $q_\star = \arg\min_{q \in \bar{\mathcal{Q}}} B_F(\tilde{p}, q)$
4. $q_\star = \arg\min_{p \in \mathcal{P}} B_F(p, q_0)$

Moreover, any of these four properties determines $q_\star$ uniquely.

Note that since $\nabla F(q_0) = 0$, $\arg\min_{p\in\mathcal{P}} B_F(p,q_0) = \arg\min_{p\in\mathcal{P}} F(p)$. Property 2 is called the *Pythagorean property* since it resembles the Pythagorean theorem if we imagine that $B_F(p,q)$ is the square of Euclidean distance and $(p, q_\star, q)$ are the vertices of a right triangle.

4 Feature Selection

The optimization problem stated in the previous section is to minimize the convex function $F(p)$ subject to linear constraints $Ap = b$. We present a new iterative algorithm for finding this minimum in the following section. But how do these constraints arise? The "learning" algorithm that we propose is a simple greedy algorithm which reduces the Bregman distance $B_F(\tilde{p}, q)$ as much as possible at each step.

We assume that we have a set of *candidate features*, or constraints:

$$\mathcal{C} = \{g : \Delta \subset \mathbf{R}^r \to \mathbf{R}\}$$

After the n-th step of the induction algorithm, we will have n linear constraints, represented by the $n \times r$ matrix A. In the $(n+1)$-st step, we consider adding all possible candidate constraints. That is, if $g \in \mathcal{C}$, let A_g be the $(n+1) \times r$ matrix

$$(A_g)_{ij} = \begin{cases} A_{ij} & \text{if } 1 \le i \le n \\ g_j & \text{if } i = n+1. \end{cases}$$

Definition 4 *If $q \in \mathcal{Q}(A, q_0)$ and $g \in \mathcal{C}$, let $\mathcal{Q}(g,q) \subset \Delta$ be given by*

$$Q(g,q) = \{q' \in \Delta \mid q' = ((0_{1\times n}, \lambda)A_g) \circ_F q,\ \lambda \in \mathbf{R}\}$$

The gain $G(q,g)$ of the candidate feature g is defined as

$$G(g,q) = \sup_{q_\lambda \in \mathcal{Q}(g,q)} \left(B_F(\tilde{p}, q) - B_F(\tilde{p}, q_\lambda)\right)$$

Algorithm 5 (Feature Selection)

Initial Data: Reference distribution $\tilde{p}$ and $q_0 \in \Delta$.
Output: A model $q_\star \in \Delta$ and constraints A such that $q_\star = \arg\min_{q \in \bar{\mathcal{Q}}(A,q_0)} B_F(\tilde{p}, q)$.
Iterate:

1. *For each candidate $g \in \mathcal{C}(q^{(n)})$ compute the gain $G(g, q^{(n)})$.*
2. *Let $g_n = \arg\max_{g \in \mathcal{C}(q^{(n)})} \mathcal{G}(g, q^{(n)})$ be the feature having the largest gain.*
3. *Compute $q_\star = \arg\min_{q \in \bar{\mathcal{Q}}(A^{(n)}_{g_n}, q_0)} B_F(\tilde{p}, q)$.*
4. *Set $q^{(n+1)} = q_\star$, $A^{(n+1)} = A^{(n)}_{g_n}$, $n \leftarrow n+1$, and go to step 1.*

Steps 2 and 3 can be carried out using the iterative scaling algorithm presented in the following section. There are many possible variations on this basic algorithm. For example, tree-like models can be constructed by taking conjunctions of features.

5 An Iterative Scaling Algorithm

We will now assume that F is of the form $F(p) = \sum_{i=1}^r f(p_i)$, where the strictly convex function f is finite and C^1 on $\mathbf{R}_+$, with $\lim_{t\to 0} f'(t) = -\infty$. We will also assume, without loss of generality, that $A_{ij} \ge 0$ for all i and j.

Consistent with our earlier notation, we will denote

$$\alpha \circ_f y = \arg\min_{x \in \mathbf{R}_+} B_f(x,y) + \alpha x$$

for $\alpha \in \mathbf{R}$ and $y \in \mathbf{R}_+$. Finally, we will use the notation

$$A^\sharp_j = \sum_{i=1}^n A_{ij}, \quad j = 1, \ldots r$$

Algorithm 6 (Iterative Scaling)

1. *Choose $q^{(0)}$ so that $f'(q^{(0)}_i) = 0$*
2. *For each $j = 1, \ldots, n$, let $\gamma^{(k)}_j \in [-\infty, \infty)$ be the unique solution of*

$$\sum_{i=1}^r A_{ji}(\gamma^{(k)}_j A^\sharp_i) \circ_f q^{(k)}_i = b_j$$

3. *Set $q^{(k+1)} = \gamma^{(k)} \circ_f q^{(k)}$ and $k \leftarrow k+1$ and go to 1.*

Theorem 7 *Let $q^{(k)}$ be the sequence determined by the Iterative Scaling algorithm. Then $B_F(p, q^{(k)})$ decreases monotonically to $B_F(\tilde{p}, q_\star)$ and $q^{(k)}$ converges to $q_\star = \arg\min_{p\in\mathcal{P}} F(p)$.*

The basic idea behind our proof of this theorem is to make use of an auxiliary function which bounds the change in divergence from below after each iteration.

Definition 8 *Fixing a linear constraint matrix A, if $\gamma \in \mathbf{R}^n$ we will now use the notation $(\gamma^\top A) \circ_F q \equiv \gamma \circ_A q$. A function $\mathcal{A} : \mathbf{R}^n \times \Delta \to \mathbf{R}$ is an auxiliary function for $L(q) \equiv -B_F(\tilde{p}, q)$ with respect to A if*

1. *For all $q \in \Delta$ and $\gamma \in \mathbf{R}^n$*
$$L(\gamma \circ_A q) \geq L(q) + \mathcal{A}(\gamma, q)$$
2. *$\mathcal{A}(\gamma, q)$ is continuous in $q \in \Delta$ and C^1 in $\gamma \in \mathbf{R}^n$ with $\mathcal{A}(0, q) = 0$ and*
$$\frac{d}{dt}\Big|_{t=0} \mathcal{A}(t\gamma, q) = \frac{d}{dt}\Big|_{t=0} L((t\gamma) \circ_A q)$$

We can use an auxiliary function $\mathcal{A}$ to construct an iterative algorithm for maximizing L. We start with $q^{(k)} = q_0$ and recursively define $q^{(k+1)}$ by

$$q^{(k+1)} = \gamma^{(k)} \circ_A q^{(k)} \quad \text{with} \quad \gamma^{(k)} = \arg\max_\gamma \mathcal{A}(\gamma, q^{(k)}) \,.$$

It is clear from property 1 of the definition that each step of this procedure increases L. The following proposition implies that in fact the sequence $q^{(k)}$ will reach the maximum of L.

Theorem 9 *Suppose $q^{(k)}$ is any sequence in Δ with*

$$q^{(0)} = q_0 \quad \text{and} \quad q^{(k+1)} = \gamma^{(k)} \circ_A q^{(k)}$$

where $\gamma^{(k)} \in \mathbf{R}^n$ satisfies

$$\mathcal{A}(\gamma^{(k)}, q^{(k)}) = \sup_\gamma \mathcal{A}(\gamma, q^{(k)}) \,.$$

Then $L(q^{(k)})$ increases monotonically to $\max_{q\in\bar{\mathcal{Q}}} L(q)$ and $q^{(k)}$ converges to the distribution $q_\star = \arg\max_{q\in\bar{\mathcal{Q}}} L(q)$.

To use the theorem to construct a practical algorithm we must determine an auxiliary function $\mathcal{A}(\gamma, q)$ for which $\gamma^{(n)}$ satisfying the required condition can be determined efficiently. This can be achieved as follows.

Lemma 10 *Let $\mathcal{A}(\gamma, q)$ be defined as*

$$\mathcal{A}(\gamma, q) = -\sum_{i=1}^{r}\sum_{j=1}^{n} A_{j|i} B_f(\tilde{p}_i, (\gamma_j A_i^\sharp) \circ_f q_i) + B_F(\tilde{p}, q)$$

where $A_{j|i} = A_{ji}/A_i^\sharp$. Then $\mathcal{A}$ is an auxiliary function for $L(q)$.

Proof: Property 2 of Definition 8 is easy to verify. To prove property 1, note that

$$\begin{aligned}
&L(\gamma \circ_A q) - L(q) \\
&\quad = B_F(\tilde{p}, q) - B_F(\tilde{p}, (\gamma^\top A) \circ_F q) \\
&\quad = B_F(\tilde{p}, q) - \sum_{i=1}^{r} B_f(\tilde{p}_i, (\gamma^\top A)_i \circ_f q_i) \\
&\quad \geq B_F(\tilde{p}, q) - \sum_{i=1}^{r}\sum_{j=1}^{n} A_{j|i} B_f(\tilde{p}_i, (\gamma_j A_i^\sharp) \circ_f q_i) \\
&\quad = \mathcal{A}(\gamma, q) \,.
\end{aligned}$$

The first equality above is a simple calculation. The second equality follows from the facts that $B_F(p,q) = \sum_i B_f(p_i, q_i)$ and $(v \circ_F q)_i = v_i \circ_f q_i$. The inequality is a consequence of the definition of $A^\sharp$ and the convexity of the Bregman distance B_f. □

Theorem 7 follows immediately from the above lemma and Theorem 9.

References

[1] L.M. Bregman, "The relaxation method to find the common point of convex sets and its applications to the solution of problems in convex programming," *USSR Computational Mathematics and Mathematical Physics*, **7**, 200–217, 1967.

[2] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*, Wadsworth, Belmont, 1984.

[3] Y. Censor, "Optimization of 'log-x'-entropy over linear equality constraints," *SIAM J. Control Optim.* **25**, 921–933, 1987.

[4] Y. Censor and A. Lent, "An iterative row-action methd for interval convex programming," *J. Optim. Theory Appl.* **34**, 321–353, 1981.

[5] I. Csiszár, "Maxent, mathematics, and information theory," In *Maximum Entropy and Bayesian Methods*, K. Hanson and R. Silver, eds., Kluwer Academic Publishers, 1996.

[6] I. Csiszár, "Why least squares and maximum entropy? An axiomatic approach to inference for linear inverse problems," *Ann. Statist.*, **19**(4), 2032–2066, 1991.

[7] I. Csiszár, "Generalized projections for non-negative functions," *Acta Math. Hungar.*, **68**(1-2), 161–185, 1995.

[8] J. Darroch and D. Ratcliff, "Generalized iterative scaling for log-linear models," *Ann. Math. Statist.* **43**, 1470–1480, 1972.

[9] S. Della Pietra, V. Della Pietra, and J. Lafferty, "Inducing features of random fields," *IEEE Trans. Pattern Analysis and Machine Intell.*, **19**(4), April 1997 (in press).

[10] A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," Journal of the Royal Statistical Society **39**, No. B, 1–38, 1977.

[11] P.S. Neelakanta, S. Abusalah, D. De Groff, R. Sudhakar, and J.C. Park, "Csiszár's generalized error measures for gradient-descent-based optimizations in neural networks using the backpropagation algorithm," *Connection Science*, **8**, No. 1, 79–114, 1996.