$\mathcal{T}$-Linear Discriminant Functions

Brijnesh J. Jain and Klaus Obermayer

Berlin University of Technology
Dept. of Electrical Engineering and Computer Science
Sekr. 2-1, Franklinstr. 28/29, 10587 Berlin, Germany

Linear discriminant functions constitute an elementary building block for learning classifiers, and therefore have been investigated thoroughly. This aim of this contribution is to generalize linear discriminant functions for finite structures such as point patterns, trees, lattices, or graphs. The proposed $\mathcal{T}$-linear discriminant functions for structures constitute an elementary building block for constructing and analyzing large margin classifiers in the domain $\mathcal{X}_{\mathcal{T}}$ and more complex supervised and unsupervised structural neural learning machines. The results builds upon and extends the theory of $\mathcal{T}$-spaces to be presented in this meeting and we therefore presume the results and notations from the corresponding abstract [2].

The most important findings of linear discriminant functions are based on the algebraic and geometric properties of the inner product. Thus, to generalize linear functions for finite structures, we first have to generalize the inner product. A key problem is the absence of an addition for structures. Hence, it is impossible to construct a similarity measure for structures that is bilinear. But we can define a similarity measure for structures that has the same geometric properties as an inner product. To this end, let $\mathcal{X}_{\mathcal{T}}$ be a $\mathcal{T}$-space over the Euclidean space $\mathcal{X}$. Then the inner product $\langle \cdot, \cdot \rangle$ on $\mathcal{X}$ induces a function

$$\langle \cdot, \cdot \rangle_{\mathcal{T}} : \mathcal{X}_{\mathcal{T}} \times \mathcal{X}_{\mathcal{T}} \to \mathbb{R}, \quad \big([\boldsymbol{x}], [\boldsymbol{y}]\big) \mapsto \max \Big\{ \langle \boldsymbol{x'}, \boldsymbol{y'} \rangle \,:\, \boldsymbol{x'} \in [\boldsymbol{x}], \boldsymbol{y'} \in [\boldsymbol{y}] \Big\},$$

where $[\boldsymbol{x}]$ denotes the equivalence class of all possible vector representations of a structure (see [2]). The inner $\mathcal{T}$-product $\langle \cdot, \cdot \rangle_{\mathcal{T}}$ gives rise to a norm-like function

$$\|\cdot\|_{\mathcal{T}} : \mathcal{X}_{\mathcal{T}} \to \mathbb{R}, \quad [\boldsymbol{x}] \mapsto \sqrt{\langle [\boldsymbol{x}], [\boldsymbol{x}] \rangle_{\mathcal{T}}}\,,$$

which together with the $\mathcal{T}$-norm satisfies the Cauchy-Schwarz inequality. Hence, the inner $\mathcal{T}$-product has the same geometrical properties as the standard inner product although it is not bilinear.

Using the inner $\mathcal{T}$-product, we define a $\mathcal{T}$-linear discriminant function by

$$y([\boldsymbol{x}]) : \mathcal{X}_{\mathcal{T}} \to \mathbb{R}, \quad [\boldsymbol{x}] \mapsto \langle [\boldsymbol{w}], [\boldsymbol{x}] \rangle_{\mathcal{T}} + b,$$

where $[\boldsymbol{x}]$ is an *input structure*, $[\boldsymbol{w}]$ is a *weight structure* and $b \in \mathbb{R}$ the *bias*. The discriminant $y([\boldsymbol{x}])$ implements a two-category classifier in the obvious way: An input structure $[\boldsymbol{x}]$ is assigned to class $\mathcal{C}_1$ if $y(\boldsymbol{x}) \geq 0$ and to class $\mathcal{C}_2$ otherwise.

Geometrical interpretation: In the Euclidean space $\mathcal{X}$, the decision boundary $\mathcal{H}([\boldsymbol{w}], b) \subseteq \mathcal{X}$ consists of all vectors $\boldsymbol{x}$ for which $y([\boldsymbol{x}]) = 0$. Geometrically,

$\mathcal{H}([\boldsymbol{w}], b)$ is a convex polytope composed of hyperplane segments, each of which is tangent to the sphere with center $\mathbf{0}$ and radius $b/\left\|[\boldsymbol{w}]\right\|_{\mathcal{T}}$. The directions of the hyperplanes segments $\mathcal{S}(\boldsymbol{w'}, b)$ are determined by the corresponding normal vectors $\boldsymbol{w'} \in [\boldsymbol{w}]$. The segments are defined by the cone of all vectors $\boldsymbol{x}$ having closest angle to $\boldsymbol{w'}$ over all elements from from $[\boldsymbol{w}]$.

Learning: Suppose that $\mathcal{Z} = \{([\boldsymbol{x}_1], y_1), \ldots, ([\boldsymbol{x}_k], y_k)\} \subseteq \mathcal{X}_{\mathcal{T}} \times \{\pm 1\}$ is a training set consisting of k input-output pairs. According to the principle of empirical risk minimization, the goal is to find a weight structure $[\boldsymbol{w^*}]$ and bias b^* that minimizes some cost function, for example the generalized perceptron criterion function [1]

$$F([\boldsymbol{w}], b) = \sum_{i=1}^{k} \max\left\{0, -y_i \cdot \Big(\langle [\boldsymbol{x}_i], [\boldsymbol{w}] \rangle_{\mathcal{T}} + b \Big) \right\}.$$

The cost function is locally Lipschitz, and therefore smooth almost everywhere. Hence, we can use subgradient methods from nonsmooth optimization to minimize F.

Suppose that the i-th example is misclassified. We adjust $[\boldsymbol{w}]$ and b according to the following update rule

$$[\boldsymbol{w}] \leftarrow [\boldsymbol{w} + \eta y_i \boldsymbol{x}_i^*],$$

where η is the learning parameter and $\boldsymbol{x}_i^* \in [\boldsymbol{x}_i]$ a representation satisfying the equation

$$\langle \boldsymbol{w}, \boldsymbol{x}_i^* \rangle = \langle [\boldsymbol{w}], [\boldsymbol{x}_i] \rangle_{\mathcal{T}} \quad \text{and} \quad b \leftarrow b + \eta y_i.$$

Provided that the learning rate is decreased at each update step t such that $\sum_t \eta_t \to \infty$ and $\sum_t \eta_t^2 < \infty$ and provided there exists a solution $[\boldsymbol{w}^*]$ and b^* with $F([\boldsymbol{w}^*], b^*) = 0$, then the algorithm generates sequences $([\boldsymbol{w}_t])_{t \in \mathbb{N}}$ and $(b_t)_{t \in \mathbb{N}}$ that converge to a solution of the separable training sample. As opposed to the standard perceptron convergence theorem, this convergence result only guarantees to converge to a solution of a separable problem in the limit after an infinite rather than finite number of update steps. Thus, from a practical point of view, there is (so far) no guarantee that we actually obtain a solution in the separable case.

It is unclear, how to derive a stronger convergence result. The key problem is that the direction of descent is not unique at nonsmooth points, which can introduce oscillations. We assume that infinite oscillations only occur in pathological cases, because the set of nonsmooth points is a set of Lebesgue measure zero. Our assumption is also supported by first experiments using synthetic data, where separable problems are solved after a finite number of update steps. Hence, further research aims at deriving conditions that ensure finite convergence.

References

1. B.J. Jain. *Structural Neural Learning Machines.* PhD thesis, Berlin University of Technology, 2005.

2. B.J. Jain and K. Obermayer. Learning in structured input and output spaces. In *The Learning Workshop, Snowbird*, 2007.