2022 30th Mediterranean Conference on Control and Automation (MED)
June 28 - July 1, 2022. Athens, Greece

2022 30th Mediterranean Conference on Control and Automation (MED) | 978-1-6654-0673-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/MED54222.2022.9837242

# Risk Sensitivity and Entropy Regularization in Prototype-based Learning

Christos Mavridis, Erfaun Noorani, and John S. Baras

***Abstract*— Prototype-based learning methods have been extensively studied as fast, recursive, data-driven, interpretable, and robust learning algorithms. We study the effect of entropy regularization in prototype-based learning regarding (i) robustness with respect to the dataset and the initial conditions, and (ii) the generalization properties of the learned representation. A duality relationship, with respect to a Legendre-type transform, between free energy and Kulback-Leibler divergence measures, is used to show that entropy-regularized prototype-based learning is connected to exponential objectives associated with risk-sensitive learning. We use these results to incentivize the development of entropy-regularized prototype-based learning algorithms by highlighting its properties, including (i) memory and computational efficiency, (ii) gradient-free training rules, and (iii) the ability to simulate an annealing optimization process that results in progressively growing competitive-learning neural network architectures.**

## I. Introduction

Learning from data samples has become an important part of machine intelligence and a component of virtually all autonomous cyber-physical systems. In particular, prototype-based learning algorithms [1]–[5] can provide valuable insights into the nature of the space of the observations, and is being used in numerous applications, including supervised [1], [6] unsupervised [7], and reinforcement learning [8], [9].

The main idea behind prototype-based learning is the representation of the data space –which is assumed to be a finite-dimensional vector space– by a set of representatives $M := \{\mu_i\}$, typically called prototypes, or codevectors [3], in an optimal way according to an average distortion measure:

$$\min_{M} \; J(M) := \mathbb{E}\left[\min_{i} d(X, \mu_i)\right],$$

where the proximity measure $d$ defines the similarity between the random input $X$ and a codevector $\mu_i$. Representing the input in terms of memorized exemplars is an intuitive approach which parallels similar concepts from cognitive psychology and neuroscience. In this regard, prototype-based algorithms can be viewed as interpretable, robust [10], data-driven and topology-preserving competitive-learning neural network architectures, which can be formulated as online stochastic approximation algorithms [1], [5], that are fast to train and sparse in the sense of memory complexity.

Research partially supported by the Office of Naval Research (ONR) grant N00014-17-1-2622, a grant from Northrop Grumman Corporation, a grant from the Army Research Lab (ARL), and by the Clark Foundation. The authors are with the Department of Electrical and Computer Engineering and the Institute for Systems Research, University of Maryland, College Park, USA. emails:{mavridis, enoorani, baras}@umd.edu.

Entropy regularization requires relaxing the original problem to a soft-clustering problem, introducing the association probabilities $p(\mu_i|X)$, and replacing the cost function $J$ by $D(M) := \mathbb{E}\left[\sum_i p(\mu_i|X) d(X, \mu_i)\right]$. This probabilistic framework allows the use of the Shannon entropy $H(M)$ as a measure of uncertainty induced by the prototype-based representation. One can replace the original problem by

$$\min_{M} \; F_\lambda(M) := D(M) - \lambda H(M),$$

parameterized by a coefficient $\lambda$, which acts as a Lagrange multiplier controlling the trade-off between minimizing the distortion $D$ and maximizing the entropy $H$.

In this work, we use known results on the duality, with respect to a Legendre-type transform, between the free energy $\log \mathbb{E}\left[e^Z\right]$ of a random variable and Kulback-Leibler divergence measures, to show that entropy-regularized prototype-based learning is connected to exponential criteria associated with risk-sensitive learning [11]–[13]. Through this connection, the results of this work formally support the experimental observations that entropy regularization provides robustness with respect to input perturbations and initial conditions [1], [14]. Finally, we use these results to incentivize the development of entropy-regularized prototype-based learning algorithms, which have recently shown several appealing properties [1], [7], [9], [15], including the ability to (i) simulate an annealing optimization process that results in progressively growing competitive-learning neural network architectures, and (ii) formulate the training rule as a gradient-free stochastic approximation algorithm with convergence guarantees.

## II. Prototype-based Learning

Prototype-based learning algorithms construct a set of prototype vectors as an optimal representation of the data space [2], [3], [5]. In particular, given a random variable $X : \Omega \to S \subseteq \mathbb{R}^d$ defined in a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, and a divergence measure $d : S \times ri(S) \to [0, \infty)$, where $ri(S)$ represents the relative interior of $S$, prototype-based unsupervised learning seeks to identify a set of codevectors $M := \{\mu_h\}_{h=1}^K$, where $\mu_h \in ri(S_h)$, for all $h = 1, \ldots, K$, such that the function $Q : S \to M$, defined as the random variable $Q(X) = \sum_{h=1}^K \mu_h \mathbb{1}_{[X \in S_h]}$ minimizes the objective function:

$$\min_{M,V} \; J(Q) := \mathbb{E}\left[d\left(X, Q\right)\right]. \tag{1}$$

This is essentially a hard-clustering problem, i.e., the quantizer $Q$ assigns an input vector $X$ to a unique codevector

$\mu_h \in M$ with probability one, and is equivalent to

$$\min_{\{\mu_h\}_{h=1}^K} \sum_{h=1}^{K} \mathbb{E}\left[d\left(X,\mu_h\right)\mathbb{1}_{[X\in S_h]}\right] \tag{2}$$

where $V := \{S_h\}_{h=1}^K$ is a Voronoi partition, i.e.,

$$S_h = \left\{ x \in S : h = \arg\min_{\tau=1,\ldots,K} \; d(x,\mu_\tau) \right\}, \; h = 1,\ldots,K.$$

The actual distribution of $X \in S$ is usually unknown, and a set of independent realizations $\{X_i\}_{i=1}^n := \{X(\omega_i)\}_{i=1}^n$, for $\omega_i \in \Omega$, are available instead. In case the observations $\{X_i\}_{i=1}^n$ are available a priori, the solution to Problem (1) can be approximated by variants of the LBG algorithm [16], which includes the widely used $k$-means algorithm [17], but recursive algorithms have been also developed [5] in case of online observations.

Problem (1) can be extended for supervised learning. In this case, Learning Vector Quantization (LVQ) algorithms [18], can be formulated in a similar framework as vector quantization algorithms that solve Problem (1), by making use of a modified distortion measure [18]–[20]. Because of their similar structure, the analysis presented in this work will be focused on the unsupervised problem. The results can be directly extended to the supervised problems, as well.

### A. The role of Bregman Divergences

Prototype-based algorithms rely on a divergence measure $d$ that quantifies the proximity between different vector representations. Bregman divergences offer a generalization of convex metrics, and are defined as follows: Let $\phi : H \to \mathbb{R}$, be a strictly convex function defined on a vector space $H$ such that $\phi$ is twice F-differentiable on $H$. Then, a Bregman divergence $d_\phi : H \times H \to [0,\infty)$ is defined as:

$$d_\phi\left(x,\mu\right) = \phi\left(x\right) - \phi\left(\mu\right) - \frac{\partial\phi}{\partial\mu}\left(\mu\right)\left(x-\mu\right), \tag{3}$$

where $x, \mu \in H$, and the continuous linear map $\frac{\partial\phi}{\partial\mu}\left(\mu\right) : H \to \mathbb{R}$ is the Fréchet derivative of $\phi$ at $\mu$.

Bregman divergences have been shown to enhance the performance of learning algorithms [21]. Two notable examples of Bregman divergences are the squared Euclidean distance $d_\phi(x,\mu) = \|x-\mu\|^2$ (generated by the function $\phi(x) = \langle x, x\rangle$, $x \in \mathbb{R}^d$), and the generalized I-divergence $d_\phi(x,y) = \langle x, \log x - \log\mu\rangle - \langle \mathbb{1}, x-\mu\rangle$ which reduces to the widely used Kullback-Leibler divergence if $\langle \mathbb{1}, x\rangle = 1$.

More importantly, they have been shown (see [1] and the references therein) to be both necessary and sufficient for the optimizer $\mu_h$ of (2) to be analytically computed as the expected value $\mu_h := \mathbb{E}\left[X|S_h\right]$ of the data inside $S_h$, which is implicitly used by many "centroid" algorithms, such as $k$-means [17]. In Section III, we will show a similar result for the proposed algorithm that uses a soft-partition approach.

## III. Soft-Clustering and Entropy Regularization

In the clustering problem (Problem 1), the distortion function $J$ is typically non convex and riddled with poor local minima.

Soft-clustering approaches have been proposed as a probabilistic generalization of the clustering problem (1), in an attempt to deal with poor local minima. In this case, the input vector $X$ is assigned, through the quantizer $Q$, to all codevectors $\mu_h \in M$ with probabilities $p(\mu_h|X)$ , where $\sum_{h=1}^K p(\mu_h|X) = 1$. The quantizer $Q : S \to M$ becomes a discrete random variable, with the set $M$ being its image, and can be fully described by the values of $M = \{\mu_h\}_{h=1}^K$ and the probability functions $\{p(\mu_h|x)\}_{h=1}^K$. We can rewrite the expected distortion as

$$\begin{aligned} D(M) &= \mathbb{E}\left[d_\phi(X,Q)\right] \\ &= \mathbb{E}\left[\mathbb{E}\left[d_\phi(X,Q)|X\right]\right] \\ &= \int p(x) \sum_\mu p(\mu|x) d_\phi(x,\mu)\; dx \end{aligned}$$

where $p(\mu|x)$ is the association probability relating the input vector $x$ with the codevector $\mu$.

The main idea behind entropy regularization, is to seek the distribution that minimizes $D$ subject to a specified level of randomness, measured by the Shannon entropy

$$\begin{aligned} H(M) &= \mathbb{E}\left[-\log p(X,Q)\right] \\ &= H(X) + H(Q|X) \\ &= H(X) - \int p(x) \sum_\mu p(\mu|x) \log p(\mu|x)\; dx \end{aligned}$$

by appealing to Jaynes' maximum entropy principle [22]. This multi-objective optimization is conveniently formulated as the minimization of the Lagrangian

$$\min_M F_\lambda(M) := D(M) - \lambda H(M) \tag{4}$$

where $\lambda$ is a parameter that acts as a Lagrange multiplier. As $\lambda$ varies, the sequence of the solutions will correspond to a Pareto curve of the multi-objective optimization (4). In this regard, the entropy $H$, acts as a regularization term.

However, one central question to be answered is what exactly $F_\lambda$ represents, and why would someone seek to minimize this measure instead of just the average distortion $D$. In Section IV we show that $\min F_\lambda$ represents the free energy $\lambda \log \mathbb{E}\left[e^{\frac{1}{\lambda} d_\phi(X,Q)}\right]$ which is directly connected to risk-sensitive learning objectives. This result formally supports why solving (4) as the sequence of problems for decreasing coefficients $\lambda$, resembles an annealing optimization process, as will be discussed in Section V.

To minimize $F\lambda$ in (4) we form a coordinate block optimization algorithm, by successively minimizing it with respect to the association probabilities $p(\mu|x)$ and the codevector locations $\mu$. Minimizing $F$ with respect to the association probabilities $p(\mu|x)$ is straightforward and yields the

Gibbs distribution

$$p(\mu|x) = \frac{e^{-\frac{d(x,\mu)}{\lambda}}}{\sum_{\mu} e^{-\frac{d(x,\mu)}{\lambda}}},\ \forall x \in S \tag{5}$$

while, in order to minimize $F$ with respect to the codevector locations $\mu$ we set the gradients to zero

$$\begin{aligned} \frac{d}{d\mu} D = 0 \implies & \frac{d}{d\mu} \mathbb{E}\left[\mathbb{E}\left[d(X,\mu)|X\right]\right] = 0 \\ \implies & \int p(x)p(\mu|x) \frac{d}{d\mu} d(x,\mu)\ dx = 0 \end{aligned} \tag{6}$$

The following theorem, proved in [1], shows that there is an analytical solution to the last optimization step (6) in a convenient centroid form, if $d$ is a Bregman divergence.

*Theorem 1:* Assuming the conditional probabilities $p(\mu|x)$ are fixed, the Langragian $F_\lambda$ in (4) is minimized with respect to the codevector locations $\mu$ by

$$\mu^* = \mathbb{E}\left[X|\mu\right] = \frac{\int xp(x)p(\mu|x)\ dx}{p(\mu)} \tag{7}$$

if $d := d_\phi$ is a Bregman divergence for some function $\phi$ that satisfies the definition.

## IV. Risk Sensitivity and Entropy Regularization

In this Section, we use known results on the duality, with respect to a Legendre-type transform, between the free energy $\log \mathbb{E}\left[e^Z\right]$ of a random variable $Z$ and Kulback-Leibler divergence measures, to show that entropy-regularized prototype-based learning is connected to exponential objectives associated with risk-sensitive learning.

### A. Risk Sensitive Learning

Risk-sensitive learning algorithms are based on objective functions that incorporate some notion of risk, e.g., higher moments of the cost function, and have shown promising results in addressing some of the issues associated with risk-neutral learning [11]–[13], [23]. A widely used risk-sensitive objective is the exponential criterion [24]:

$$\min_{\mu} J_\beta(\mu) := \frac{1}{\beta} \log \mathbb{E}\left[e^{\beta C(\mu)}\right] \tag{8}$$

where $C(\mu)$ represents a cost function that depends on a parameter vector $\mu \in \mathbb{R}^d$, and the risk parameter $\beta \in \mathbb{R}$ is a design parameter, which controls the level of risk-seeking or risk-averse behaviour of the agent. Optimization of risk-sensitive performance measures have a long history, starting with Markowitz mean-variance Portfolio Theory [25]. The exponential criterion is a mathematically convenient and intuitively appealing risk measure with a firm theoretical foundations rooted in Large Deviation Theory. Its Taylor's series expansion reads as

$$\frac{1}{\beta} \log \mathbb{E}\left[e^{\beta C(\mu)}\right] = \mathbb{E}\left[C(\mu)\right] + \frac{\beta}{2} \mathrm{Var}\left[C(\mu)\right] + O(\beta^2) \tag{9}$$

revealing that the risk parameter $\beta$ controls the trade-off between the maximization of the expectation and maximization/minimization of risk, quantified mainly by the variance $\mathrm{Var}\left[C(\mu)\right]$ of the cost function $C(\mu)$ and to a lesser degree by higher order terms. When $J_\beta$ is an objective function to be minimized (as in (8)), it is called risk-averse (or "pesimistic") for $\beta > 0$ and risk-seeking (or "optimistic") for $\beta < 0$.

### B. Duality between Free Energy and KL divergence

Consider a measurable space $(\Omega, \mathcal{F},$ where $\mathcal{F})$ is a $\sigma$-algebra on $\Omega$. Let $\mathcal{P}(\Omega)$ be a set of probability measures $P : \Omega \to [0,1]$, and $P_\mu, P_\nu \in \mathcal{P}(\Omega)$. In addition, consider a bounded measurable function $Z : \Omega \to \mathbb{R}$. The exponential criterion

$$J_\beta(Z) = \frac{1}{\beta} \log \mathbb{E}_{P_\mu}\left[e^{\beta Z}\right] \tag{10}$$

is called free energy [26]. It is known (see e.g. [27] and the references therein) that the free energy $J_\beta(Z)$ and the KL divergence:

$$D_{KL}(P_\nu, P_\mu) = \begin{cases} \int \log(\frac{dP_\nu}{dP_\mu}) dP_\nu & \text{if } C_{KL}(P_\nu, P_\mu) \\ \infty & \text{otherwise} \end{cases} \tag{11}$$

are in duality with respect to a Legendre-type transform [28], in the following sense:

$$J_\beta(Z) = \begin{cases} \sup_{P_\nu \in \mathcal{P}(\Omega)} \left\{ \mathbb{E}_{P_\nu}\left[Z\right] - \frac{1}{\beta} D_{KL}(P_\nu, P_\mu) \right\},\ \beta > 0 \\ \inf_{P_\nu \in \mathcal{P}(\Omega)} \left\{ \mathbb{E}_{P_\nu}\left[Z\right] - \frac{1}{\beta} D_{KL}(P_\nu, P_\mu) \right\},\ \beta < 0 \end{cases} \tag{12}$$

Here the conditions $C_{KL}(P_\nu, P_\mu)$ include $P_\nu \ll P_\mu$ and $\int \log(\frac{dP_\nu}{dP_\mu}) dP_\nu \in L^1(P_\nu)$.

### C. Connection to Entropy-regularized Prototype-based Learning

We can formulate the risk-sensitive objective for prototype-based learning problem according to the definitions of Section III and IV-A. As before, let $X : \Omega \to S \subseteq \mathbb{R}^d$ be a random vector defined in a measurable space $(\Omega, \mathcal{F})$, equipped with a set $\mathcal{P}(\Omega)$ of probability measures $P : \Omega \to [0,1]$, and $\mu := \{\mu_h\}_{h=1}^K$ a set of codevectors, such that $\mu_h \in S$, for all $h = 1, \ldots, K$. Let also a quantizer $Q : S \to \mu$ be a discrete random variable, with the set $\mu$ being its image.

We define the random vector $Z = (X, Q) \in \mathbb{R}^{2d}$ and a Bregman divergence measure (see Section II-A) $d : \mathbb{R}^d \times \mathbb{R}^d \to [0, \infty)$ such that $d : \Omega \to \mathbb{R}$ is a bounded measurable function. We note that a measurable function $P_\mu : (X, Q) \mapsto [0,1]$ depends on the parameters $\mu := \{\mu_h\}_{h=1}^K$ and is a probability measure in $(\Omega, \mathcal{F})$, i.e., $P_\mu \in \mathcal{P}(\Omega)$. The risk-sensitive prototype-based learning objective now takes the form:

$$J_\beta(\mu) := \frac{1}{\beta} \log \mathbb{E}\left[e^{\beta d(Z_\mu)}\right] \tag{13}$$

where we have used the implicit notation $Z_\mu \sim P_\mu$, i.e., $\mathbb{E}\left[e^{\beta d(Z_\mu)}\right] := \mathbb{E}_{Z \sim P_\mu}\left[e^{d(Z)}\right]$. Using the duality relation (12), the risk-sensitive objective function (13) becomes:

$$J_\beta(\mu) = \begin{cases} \sup_{\nu} \left\{ \mathbb{E}\left[d(Z_\nu)\right] - \frac{1}{\beta} D_{KL}(P_\nu, P_\mu) \right\},\ \beta > 0 \\ \inf_{\nu} \left\{ \mathbb{E}\left[d(Z_\nu)\right] - \frac{1}{\beta} D_{KL}(P_\nu, P_\mu) \right\},\ \beta < 0 \end{cases} \tag{14}$$

where it is understood that $Z_\mu \sim P_\mu$ and $Z_\nu \sim P_\nu$ with $\mu, \nu$ being different parameter vectors (sets of codevectors), and $P_\mu, P_\nu \in \mathcal{P}(\Omega)$.

We focus on the "optimistic" problem, i.e., when $\beta < 0$, and, without loss of generality, assume that the infimum can be attained, i.e.,

$$J_\beta(\mu) = \min_\nu \left\{ \mathbb{E}\left[d(Z_\nu)\right] - \frac{1}{\beta} D_{KL}(P_\nu, P_\mu) \right\} \tag{15}$$

Equation (15) is similar to the entropy-regularization problem in (4) and can give insights on what the Lagrangian $F_\lambda$ in (4) represents, justify the properties of the algorithms that are based on it, and reveal ways that these algorithms can be generalized.

In particular, under the assumption that $P_\mu$ is a uniform probability measure, the KL divergence can be written as $-D_{KL}(P_\mu, P_\nu) = H(P_\mu) - I$, where $I$ is a constant term [29]. Therefore, in this case (15) is equivalent with

$$J_\beta(\mu) = \min_\nu \left\{ \mathbb{E}\left[d(Z_\nu)\right] - \frac{1}{|\beta|} H(Z_\nu) \right\}, \ \mu \sim U \tag{16}$$

where $H(Z_\nu) = \mathbb{E}_{Z\sim P_\nu}\left[-\log p(Z)\right]$ represents the joint entropy of $Z = (X, Q)$ under the probability measure $P_\nu$. It follows directly that $\min F_\lambda = J_{\frac{1}{|\beta|}}(\mu)$, if $\mu \sim U$ follows the uniform distribution. In other words, under the assumption that $P_\mu$ is a uniform probability measure, the Lagrangian $F_\lambda$ represents the Helmholtz free energy in statistical mechanics [26], a result that formally explains the properties of the annealing process described in Section V-B.

Notice that the Helmholtz free energy is only one instance of the objective function $J_\beta$, and different heuristic assumptions on the probability measure $P_\mu$ result in different measures $J_\beta$. For example, if we assume that $P_\mu = P_\nu$, then $D_{KL}(P_\nu, P_\mu) = 0$, and (15) retrieves the risk-neutral learning problem:

$$J_\beta(\mu) = \min_\nu \mathbb{E}\left[d(Z_\nu)\right], \ \mu = \nu \tag{17}$$

where $Z_\nu = (X, Q) \sim P_\nu \sim P_\mu$. As a final note, solving the risk-sensitive learning problem (13) makes no assumption on the measure $P_\mu$, and results in a joint optimization problem

$$\begin{aligned} \min_\mu J_\beta(\mu) &= \min_\mu \frac{1}{\beta} \log \mathbb{E}\left[e^{\beta d(Z_\mu)}\right] \\ &= \begin{cases} \min_\mu \max_\nu \left\{ \mathbb{E}\left[d(Z_\nu)\right] - \frac{1}{\beta} D_{KL}(P_\nu, P_\mu) \right\}, & \beta > 0 \\ \min_\mu \min_\nu \left\{ \mathbb{E}\left[d(Z_\nu)\right] - \frac{1}{\beta} D_{KL}(P_\nu, P_\mu) \right\}, & \beta < 0 \end{cases} \end{aligned} \tag{18}$$

Equation (18) reveals that the risk-sensitive problem implicitly computes the optimal representation $P_\mu$ instead of relying on different heuristics, but constitutes a significantly harder problem to solve numerically. The advantages of solving the entropy-regularization problem in (4) instead are highlighted in Section V.

## V. Properties of Entropy-regularized Prototype-based Learning

In this section we demonstrate the potential advantages of solving the entropy-regularized problem defined in (4) compared to solving the original vector quantization Problem (1) and its risk-sensitive counterpart in (18).

### A. Robustness

As explained in Section IV-C, the entropy-regularized objective $F_{\frac{1}{|\beta|}}$ in (4) is connected to the risk-sensitive exponential objective $\frac{1}{\beta}\log\mathbb{E}\left[e^{\beta d_\phi(X,Q)}\right]$, the Taylor's series expansion of which reads as

$$D(X, Q) + \frac{\beta}{2}\text{Var}\left[d_\phi(X, Q)\right] + O(\beta^2) \tag{19}$$

where $\beta < 0$. In other words, the average distortion objective function has been augmented by the variance term $\text{Var}\left[d_\phi(X, Q)\right]$, and, to a lesser degree, by higher order terms. The variance term is connected to the distribution of the random vector $(X, Q)$. Minimizing the distortion measure $D(X, Q)$ subject to a level of variance $\text{Var}\left[d_\phi(X, Q)\right]$ is a heuristic approach that alleviates the effect of the initial conditions (random variable $Q$), which aligns with the observations found in [1]. The same reasoning extends to the robustness of the approach with respect to perturbations in the dataset, which are associated with the random variable $X$. The variance term $\text{Var}\left[d_\phi(X, Q)\right]$ is directly connected to the dissimilarity measure $\mathbb{E}\left[d_\phi(X, \mu_h)\mathbb{1}_{[X\in S_h]}\right]$ inside each Voronoi region $S_h$ (see Section II), as opposed to the total average distortion $\int p(x)\sum_h p(\mu_h|x) d_\phi(x, \mu_h)\, dx$ which is minimized by the first term $D(M)$. This is directly connected to the "purity" of each cluster, in terms of how cohesive each Voronoi region is forced to become, i.e., the parameter $\beta$ implicitly controls the average distortion of the data points of each Voronoi region with respect to their representative.

### B. Progressively Growing Set of Prototypes

Solving the entropy-regularized problem (4) can add several interesting properties to the implementation of the learning algorithm. One such property is the ability to progressively "grow" the set of codevectors $M = \{\mu_h\}_{h=1}^K$, by adaptively adding more codevectors, thus increasing the number $K$. We note that if the prototype-based learning architecture is viewed as a competitive-learning neural network, this property provides the ability to start with a few neurons and progressively grow the neural network as needed.

The main observation towards this goal is to view the minimization of the Lagrangian $F_\lambda$ in (4)

$$\min_M F_\lambda(M) := D(M) - \lambda H(M)$$

as a sequence of deterministic optimization problems, parameterized by the Lagrange coefficient $\lambda$, that are progressively solved at successively reducing parameter levels. This has been shown to correspond to an annealing process [1], [14] where $\lambda$ represents a temperature level $T$. This annealing process is experimentally shown to contribute to avoiding poor local minima, and provide robustness with respect to the initial conditions. We stress that both these properties are formally justified by the analysis given in this work.

Adding to the robustness properties of this annealing process, it is significant that, as the temperature $T$ is lowered, the system undergoes a sequence of "phase transitions", which consists of natural cluster splits where the cardinality of the codebook (number of clusters) increases. This is

a bifurcation phenomenon and provides a useful tool for controlling the size of the clustering model relating it to the scale of the solution. At very high values ($\lambda \to \infty$) the optimization yields uniform association probabilities

$$p(\mu|x) = \lim_{T\to\infty} \frac{e^{-\frac{d(x,\mu)}{T}}}{\sum_{\mu} e^{-\frac{d(x,\mu)}{T}}} = \frac{1}{K}$$

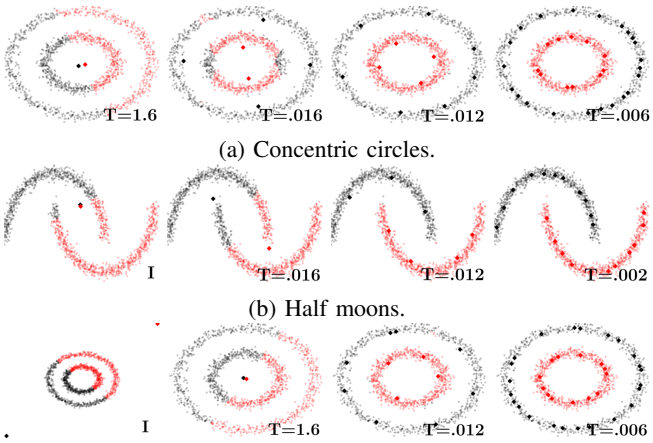
and, provided $d := d_\phi$ is a Bregman divergence, all the codevectors are located at the same point $\mu = \mathbb{E}[X]$, which is the expected value of $X$ (Theorem **??**). This is true regardless of the number of codevectors available. As $T$ decreases and reaches a critical temperature level, the number of unique solutions to the optimization problem increases. The number of different codevectors resulting from the optimization process is referred to as *effective codevectors* [1]. These define the cardinality of the codebook, which changes as we lower the temperature. In other words, an algorithmic implementation needs only as many codevectors as the number of effective codevectors, which depends only on the parameter $T$ (or $\lambda$), i.e. the Lagrange multiplier of the multi-objective minimization problem in (4).

We showcase this bifurcation phenomenon in two simple, but illustrative, binary classification problems in two dimensions (Fig. 1). The underlying class distributions are shaped as concentric circles (Fig. 1a), and half moons (Fig. 1b), respectively. All datasets consist of 1500 samples. Since the objective is to give a geometric illustration of how the algorithm works, the squared Euclidean distance is used as a proximity measure. The algorithm starts at high values of $T$ (or $\lambda$) with a single codevector for each class. As the temperature coefficient $T$ gradually decreases (Fig. 1, from left to right), the number of codevectors, i.e., the complexity of the model, progressively increases. The accuracy of the algorithm typically increases as well. Finally, Fig. 1c showcases the robustness of the proposed algorithm with respect to the initial configuration. Here the codevectors are poorly initialized outside the support of the data, which is not assumed known a priori (e.g. online observations of unknown domain). In this example the LVQ algorithm has been shown to fail [30]. In contrast, the entropy term $H$ in the optimization objective of (4), allows for the online adaptation to the domain of the dataset and helps to prevent poor local minima.

### C. Recursive Gradient-Free Training Rule

Another advantage of the entropy-regularized problem (4) is the existence of an analytic solution to the optimization problem, given by (5), and (7). While the Gibbs distribution in (5) is easy to compute, the conditional expectation $\mathbb{E}[X|\mu]$ in eq. (7) needs to be approximated by the use of a large dataset. This is not ideal in most practical applications and results to computationally costly iterations that are slow to converge. We note that this is a problem that occurs when trying to solve the risk-sensitive problem (13), as well.

To deal with this problem, an Online Deterministic Annealing (ODA) algorithm was proposed in [1], as a recursive training rule that dynamically updates an estimate of




(a) Concentric circles.

(b) Half moons.

(c) Poor initial conditions.

Fig. 1: $(a)$-$(b)$Illustration of the evolution of the annealing process for decreasing temperature $T$ in binary classification in 2D. $(c)$ Showcasing robustness with respect to bad initial conditions.

the effective codevectors using one data observation at a time. The online training rule is formulated as a stochastic approximation algorithm [1]:

$$\begin{cases} \rho_i(n+1) & = \rho_i(n) + \alpha(n)\left[\hat{p}(\mu_i|x_n) - \rho_i(n)\right] \\ \sigma_i(n+1) & = \sigma_i(n) + \alpha(n)\left[x_n \hat{p}(\mu_i|x_n) - \sigma_i(n)\right] \end{cases} \tag{20}$$

where the quantities $\hat{p}(\mu_i|x_n)$ and $\mu_i(n)$ are recursively updated as follows:

$$\begin{aligned} \hat{p}(\mu_i|x_n) &= \frac{\rho_i(n) e^{-\frac{d(x_n,\mu_i(n))}{T}}}{\sum_i \rho_i(n) e^{-\frac{d(x_n,\mu_i(n))}{T}}} \\ \mu_i(n) &= \frac{\sigma_i(n)}{\rho_i(n)}, \end{aligned} \tag{21}$$

The recursive algorithm (20), (21) is gradient-free and converges almost surely to a possibly sample path dependent solution of the block optimization (5), (7) [1].

### D. Reduced Complexity

The recursive nature of the algorithm (20), (21) results in a significant reduction in complexity, that comes in two levels. The first refers to the recursive nature of the optimization iterations and the fact that no gradients need to be estimated. The second refers to huge reduction in memory complexity, since we bypass the need to store the entire dataset, as well as the association probabilities $\{p(\mu|x),\ \forall x\}$ that map each data point in the dataset to each cluster.

The complexity of the recursive approach (20), (21) for a fixed temperature coefficient $T_i$ (or $\lambda_i$) is $O(N_{c_i}(2K_i)^2 d)$, where $N_{c_i}$ is the number of stochastic approximation iterations needed for the convergence of (20) and corresponds to the number of data samples observed, $K_i$ is the number of codevectors of the model at temperature $T_i$, and $d$ is the dimension of the input vectors, i.e., $X \in S \subseteq \mathbb{R}^d$. Therefore, assuming a training dataset of $N$ samples and a temperature

schedule $\{T_1 = T_{max}, T_2, \ldots, T_{N_T} = T_{min}\}$, the worst case complexity of the annealing approach becomes [6]:

$$O(N_c(2\bar{K})^2 d)$$

where $N_c = \max_i \{N_{c_i}\}$ is an upper bound on the number of data samples observed until convergence at each temperature level, and

$$N_T \leq \bar{K} \leq \min \left\{ \sum_{n=0}^{N_T-1} 2^n, \sum_{n=0}^{\log_2 K_{max}} 2^n \right\} < N_T K_{max}$$

where the actual value of $\bar{K}$ depends on the bifurcations occurred as a result of reaching critical temperatures and the effect of the regularization mechanisms described above. Note that typically $N_c \ll N$ as a result of the stochastic approximation algorithm, and $\bar{K} \ll N_T K_{max}$ as a result of the progressive nature of the ODA algorithm. For more details the readers are referred to [6].

## VI. Conclusion

We have studied the effect of entropy regularization in prototype-based learning regarding the learned representation (set of prototypes) and the robustness of the algorithm implementations. We use known results on the duality, with respect to a Legendre-type transform, between the free energy and Kulback-Leibler divergence measures, to show that entropy-regularized prototype-based learning is connected to exponential objectives associated with risk-sensitive learning. We use these results to incentivize the development of entropy-regularized prototype-based learning algorithms as recursive, data-driven, interpretable, robust, and fast to train and evaluate algorithms for both unsupervised and supervised problems. In particular we highlight their (i) memory and computational efficiency, (ii) ability to be trained with recursive gradient-free optimization methods, and (iii) ability to simulate an annealing optimization process that results in the development of progressively growing competitive-learning neural network architectures.

## References


[1] C. N. Mavridis and J. S. Baras, "Online deterministic annealing for classification and clustering," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[2] A. Banerjee, S. Merugu, I. S. Dhillon, and J. Ghosh, "Clustering with bregman divergences," *Journal of Machine Learning Research*, vol. 6, no. Oct, pp. 1705–1749, 2005.

[3] M. Biehl, B. Hammer, and T. Villmann, "Prototype-based models in machine learning," *Wiley Interdisciplinary Reviews: Cognitive Science*, vol. 7, no. 2, pp. 92–111, 2016.

[4] T. Villmann, S. Haase, F.-M. Schleif, B. Hammer, and M. Biehl, "The mathematics of divergence based online learning in vector quantization," in *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*. Springer, 2010, pp. 108–119.

[5] C. N. Mavridis and J. S. Baras, "Convergence of stochastic vector quantization and learning vector quantization with bregman divergences," *IFAC-PapersOnLine*, vol. 53, no. 2, 2020.

[6] C. Mavridis and J. Baras, "Towards the one learning algorithm hypothesis: A system-theoretic approach," *arXiv preprint arXiv:2112.02256*, 2021.

[7] C. N. Mavridis and J. S. Baras, "Progressive graph partitioning based on information diffusion," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 37–42.

[8] ——, "Vector quantization for adaptive state aggregation in reinforcement learning," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 2187–2192.

[9] C. N. Mavridis, N. Suriyarachchi, and J. S. Baras, "Maximum-entropy progressive state aggregation for reinforcement learning," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 5144–5149.

[10] S. Saralajew, L. Holdijk, M. Rees, and T. Villmann, "Robustness of generalized learning vector quantization models against adversarial attacks," in *International Workshop on Self-Organizing Maps*. Springer, 2019, pp. 189–199.

[11] E. Noorani and J. S. Baras, "Risk-sensitive reinforcement learning and robust learning for control," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 2976–2981.

[12] ——, "Risk-sensitive reinforce: A monte carlo policy gradient algorithm for exponential performance criteria," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 1522–1527.

[13] ——, "Embracing risk in reinforcement learning: The connection between risk-sensitive exponential and distributionally robust criteria," in *2022 American Control Conference (ACC)*. IEEE, 2022.

[14] K. Rose, "Deterministic annealing for clustering, compression, classification, regression, and related optimization problems," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2210–2239, 1998.

[15] C. N. Mavridis, N. Suriyarachchi, and J. S. Baras, "Detection of dynamically changing leaders in complex swarms from observed dynamic data," in *International Conference on Decision and Game Theory for Security*. Springer, 2020, pp. 223–240.

[16] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications*, vol. 28, 1980.

[17] L. Bottou and Y. Bengio, "Convergence properties of the k-means algorithms," in *Advances in Neural Information Processing Systems*, 1995, pp. 585–592.

[18] T. Kohonen, *Learning Vector Quantization*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1995, pp. 175–189.

[19] A. Sato and K. Yamada, "Generalized learning vector quantization," in *Advances in Neural Information Processing Systems*, 1996, pp. 423–429.

[20] B. Hammer and T. Villmann, "Generalized relevance learning vector quantization," *Neural Networks*, vol. 15, no. 8-9, pp. 1059–1068, 2002.

[21] H. K. B. Babiker and R. Goebel, "Using kl-divergence to focus deep visual explanation," *arXiv preprint arXiv:1711.06431*, 2017.

[22] E. T. Jaynes, "Information theory and statistical mechanics," *Physical Review*, vol. 106, no. 4, p. 620, 1957.

[23] Y. Chow and M. Ghavamzadeh, "Algorithms for CVaR Optimization in MDPs," *Advances in Neural Information Processing Systems*, vol. 27, pp. 3509–3517, 2014.

[24] D. Jacobson, "Optimal Stochastic Linear Systems With Exponential Performance Criteria and Their Relation to Deterministic Differential Games," *IEEE Transactions on Automatic Control*, vol. 18, no. 2, pp. 124–131, 1973.

[25] H. Markowitz, "Portfolio Selection," *The Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.

[26] J.-D. Deuschel and D. W. Stroock, *Large deviations*. American Mathematical Soc., 2001, vol. 342.

[27] P. Dai Pra, L. Meneghini, and W. J. Runggaldier, "Connections between stochastic control and dynamic games," *Mathematics of Control, Signals and Systems*, vol. 9, no. 4, pp. 303–326, 1996.

[28] M. Donsker and S. Varadhan, "Large deviations for markov processes and the asymptotic evaluation of certain markov process expectations for large times," in *Probabilistic Methods in Differential Equations*. Springer, 1975, pp. 82–88.

[29] A. Galashov, S. M. Jayakumar, L. Hasenclever, D. Tirumala, J. Schwarz, G. Desjardins, W. M. Czarnecki, Y. W. Teh, R. Pascanu, and N. Heess, "Information Asymmetry in KL-Regularized RL," pp. 1–25, 2019.

[30] J. S. Baras and A. LaVigna, "Convergence of a neural network classifier," in *Advances in Neural Information Processing Systems*, 1991.