Neural Networks 60 (2014) 203–221

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Online computing of non-stationary distributions velocity fields by an accuracy controlled growing neural gas

Hervé Frezza-Buet *

Supélec, 2, rue Édouard Belin, 57070 Metz, France
Georgia Tech Lorraine, UMI 2958, 2-3, rue Marconi, 57070 Metz, France

ARTICLE INFO

Article history:
Received 9 March 2014
Received in revised form 29 August 2014
Accepted 31 August 2014
Available online 8 September 2014

Keywords:
Vector quantization
Growing neural gas
Velocity field

ABSTRACT

This paper presents a vector quantization process that can be applied online to a stream of inputs. It enables to set up and maintain a dynamical representation of the current information in the stream as a topology preserving graph of prototypical values, as well as a velocity field. The algorithm relies on the formulation of the accuracy of the quantization process, that allows for both the updating of the number of prototypes according to the stream evolution and the stabilization of the representation from which velocities can be extracted. A video processing application is presented.

1. Introduction

Our every day life interaction with the world we are immersed in relies on our ability to recognize people, objects, places, from the flow of analogical signals provided by our sensors. In other words, our brain is able to process dynamical, multi-modal and continuous perceptive signals in a way that enables us to be aware of our reality through the mental handling of symbols, which are the constitutive discrete elements involved in our cognition. Bridging the gap between the discrete, serial and predicative nature of our mind (including speech) and the analogous, high-dimensional, complex and of course unlabeled information provided by the world, is done effortlessly by each of us. Nevertheless, endowing a machine with the least of such skills proved to be a great challenge for computer scientists since the earliest days of artificial intelligence and machine learning. Indeed, automated manipulation of logical predicates hardly meets signal processing techniques, even if both fields provide advanced computing paradigms. This difficulty is referred to as the anchoring problem (Coradeschi & Saffiotti, 2003), that every robotic engineer has experienced when s/he spends hours in adjusting, in vain, the thresholds of crucial decision-making parts of his/her control system.

In the field of machine learning and statistics, vector quantization techniques offer a battery of tools for representing the distribution of vectors sampled according to some unknown and often continuous, process. This representation relies on a finite number of prototypical vectors, determined according to the statistics of the population of the sampled vectors. A straightforward procedure introduced in Martinez and Schulten (1994) enables to connect the obtained prototypes in order to form a *graph* that reflects the topology of the input vector distribution given by the process. Considering this graph as a full symbolic representation of this input is certainly improper, but the fact remains that a graph is a discrete representation of the input process that delivers the vectors, for which graph-related algorithms can be used. This is why we consider the topology preserving vector quantization techniques as relevant approaches to the anchoring problem. The work presented in this paper is a step in this direction. An overview of clustering approaches for artificial intelligence can be found in Qin and Suganthan (2004) and a focus on topology preserving techniques in García-Rodríguez et al. (2012).

Most approaches of vector quantization consider stationary input distributions, but modifications of the algorithms have been proposed in order to cope with non-stationarity (Frezza-Buet, 2008; Fritzke, 1997). This consists in designing algorithms which are robust to changes in the input statistics, so that they keep on representing the instantaneous properties of the input while it changes. In this paper, we aim at going one step further and represent (by a topology preserving vector quantization) the current input statistics staying sensitive to their variations. Indeed, such temporal variations of the input may contain the relevant cues for understanding the semantics of the input. This is for example the motivation for the approaches based on optical flows in computer vision (Chao, Gu, & Napolitano, 2014). In the approach presented in this paper, as opposed to other vector quantization methods for

* Correspondence to: Supélec, 2, rue Édouard Belin, 57070 Metz, France. Tel.: +33 387764735.
E-mail address: Herve.Frezza-Buet@supelec.fr.

http://dx.doi.org/10.1016/j.neunet.2014.08.014
0893-6080/

non stationary data, temporal variations are represented by a velocity field built and updated over the graph representing the data distribution.

The paper is organized as follows. In Section 2 are introduced general notations, as well as a way to model the input data. This model allows for a *quantitative interpretation* of the parameters of our algorithm. Fitting a real data flow to this model provides the semantics for the most central parameters of our approach and thus rationalizes the settings. This section also introduces the *Voronoï contribution* that measures the quality of some vector quantization process. Relying on this measure to control the quantization process is a contribution of this paper and this is what Section 3 addresses. Section 4 presents a new version of our previous GNG-T algorithm (Frezza-Buet, 2008). The explicit use of Voronoï contribution and a real semantics of the parameters enables us to reformulate this previous work into a much more stable algorithm. Because of this significant improvement, GNG-T can be decorated with few additional instructions in order to capture the velocity field of the input distribution, as Section 5 shows. Section 6 presents experiments on 2D non-stationary distributions of pixels, taken from a video as well as a comparison with optical flows. Section 7 concludes.

Last, let us underline that throughout the paper, all algorithms are provided, as well as links to videos and code available from our web site. This allows for reproducing our experiments as well as using GNG-T for addressing broader application domains.

2. Notations and properties

One crucial problem in vector quantization is choosing the appropriate number of prototypes. In case of the clustering into a previously known number of clusters, as for example for clustering digits into ten groups, the appropriate number of prototypes is induced by the problem itself. However, when quantization aims at summarizing a continuous distribution by a discrete set of prototypes, one has to determine the number of prototypes to be used. This choice is even more delicate when non stationary distributions are concerned, since this number of prototypes needs to be adjusted while the distribution changes, in order to be kept permanently appropriate. After all, the meaning of *appropriate* is the core question when the number of prototypes has to be determined.

Besides particularities in their learning rules, the approaches in the Growing Neural Gas (GNG) literature implement a strategy for controlling the number of prototypes. However, this process mainly relies on empirically tuned threshold values, except for RGNG (Qin & Suganthan, 2004) that is based on a minimum description length criterion and for approaches like HOGNG (Cao & Suganthan, 2003) for which GNG participates in a supervised learning where the empirical risk is available for being used as a stopping criterion. For the approaches involving a classical GNG, as it was done recently for character recognition in Fujita (2013), the number of prototypes grows until it reaches a preset maximal value or until the error accumulated for each prototype, when the whole data set is presented, is below a predefined threshold. As the point is to find the suitable number of prototypes for a *fixed* data set, increasing the number of prototypes until some error-based stopping criterion is met remains feasible and then the process stops. Some modified GNG, like GANG (Cselényi, 2005) or SGNG (Tence, Gaubert, Soler, De Loor, & Buche, 2013) applies a similar strategy. SGONG (Stergiopoulou & Papamarkos, 2009) slightly differs since the number of samples for which a given prototype wins is used. The network grows until this number is lower than some threshold. Nevertheless, for the application to hand gesture recognition, the authors also adopt a strategy consisting of using 33 prototypes, from empirical considerations. Last, let us mention GWR (Marsland, Shapiro, & Nehmzow, 2002) that adds a habituation criterion to the error-based criterion. Both are driven by thresholds. Moreover, the habituation calculation makes sense for stationary distributions only.

For our GNG-based approach as well, a strategy for controlling the number of prototypes is proposed. It has the advantage of enabling the user to define a priori, by setting a scalar parameter, what an *appropriate* number of prototypes actually means, according to how s/he intends to use the data. Once defined, this parameter is kept constant even when the data distribution changes, as opposed to the previous methods considering stationary inputs only. As forthcoming Section 3 shows, the meaning of *appropriate* relies on two concepts, introduced beforehand in this section. The first one (Section 2.1) is a view of the data as a rejection sampling process, applied at each time since the data is non stationary. The second one (Section 2.2) is the Voronoï contribution, that is a measure used in the control strategy introduced in Section 3. The relevance of this latter concept is shown by empirical measures, given at the end of Section 2.2.

2.1. Modeling a non stationary input

Let us denote by X a bounded input set from which input samples $\xi \in X$ are drawn. Let us model the variation of input samples concentration over X as a density function $p \in [0, 1]^X$, where B^A is the set of functions from A to B. Note that density p is not a probability density since it is not required to be normalized. Let us denote by $a \sim \mathcal{U}_A$ a value a sampled according to a uniform distribution over A. Let us define a sample set $S_p^N \subset X, N \in \mathbb{N}$ a finite set of samples obtained according to p by Algorithm 1.

Algorithm 1 Computation of S_p^N.

```
1: S_p^N ← ∅ // Start with an empty set.
2: for i ← 1 to N do
3:   // Let us consider N attempts to add a sample in S_p^N.
4:   ξ ∼ U_X,  u ∼ U_[0,1[ // Choose a random position ξ.
5:   if u < p(ξ) then
6:     // The test will pass with a probability p(ξ).
7:     S_p^N ← S_p^N ∪ {ξ} // ξ is kept (i.e. not rejected).
8:   end if
9: end for
10: return S_p^N
```

Such a procedure is close to a rejection sampling of the probability density related to p (Andrieu, de Freitas, Doucet, & Jordan, 2003). The actual number of samples in S_p^N depends on both p and N, which will be discussed further.

A non stationary input is modeled here as a sequence of sample sets. Let $p^t \in [0, 1]^X$ be a non stationary density and N^t some arbitrary sampling number at time t. A non stationary input stream is defined throughout the paper as the sequence $\left(S_{p^t}^{N^t}\right)_{t\in T}$. Let us stress that the input stream defined by this way is discrete, since time instants are organized as a sequence and each $S_{p^t}^{N^t}$ is a finite set of samples.

2.2. Voronoï contribution

Vector quantization basically consists in summarizing a distribution by a finite set of representative values, usually called the prototypes. The prototypes are representative since they are chosen in order to minimize a distortion function. Classical definitions of vector quantization concepts can be found in Patra (2011), where densities of probability are concerned. Let us recall here these definitions in the restricted case of a finite input sample set