

Complexity bounds of emerging structures in self-organizing networks

Achim G. Hoffmann

University of New South Wales

School of Computer Science & Engineering

Kensington, 2033 NSW, Australia

E-mail: achim@cs.unsw.oz.au

Abstract

The idea of self-organizing systems is to acquire a meaningful internal structure just by being exposed to some ‘natural’ environment. Due to the complex network dynamics it appears very hard to analyze the structures that may emerge in such a system. Results on the complexity of classification functions and the preconditions necessary in order to allow the computation of such functions are presented.

1 Introduction

Processes of self-organization are believed to be capable to capture complex environmental conditions by emerging system structures, e.g. Grossberg [3], or Ritter et.al. [6] etc. The basic idea is, to have a system which learns to behave useful in a certain sense by just getting some *unclassified* objects of the respective domain. Here, the system gets *no* feedback whether its learning approach is correct or not. I.e. self-organization is a kind of unsupervised learning. This contrasts supervised approaches, where the system is confronted with pre-classified objects or feedback whether its predictions have been correct.

Recent work on probabilistic analysis on clustering and self-organization can be found e.g. in [1]. As a consequence, the only source of information for the system is the fact, that not all possible objects will be presented, but only a certain ‘meaningful’ subset. This subset is assumed to show some ‘natural’ clusters which should be recognized by the self-organizing system. Figure 1 shows the ‘data perspective’. In figure 2 the ‘system perspective’ of the self-organization process is shown. A possible application may be to extract compound features for (high level) symbolic learning approaches. In the following, the general computational abilities of self-organizing systems are investigated. By the use of Kolmogorov complexity the possibly complex system dynamics became analyzable. The paper is organized as follows. The formal

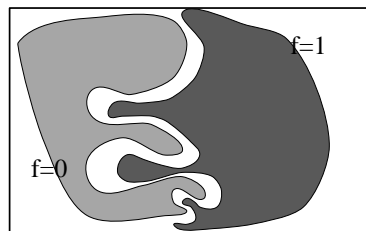


Figure 1: The subset Y_n (shaded) of the basic set X_n and a classification function f .

framework for the considerations in this paper is presented in section 2. In section 3 the complexity of classification functions that can be acquired from unclassified samples is investigated. Section 4 contains concluding remarks.

2 Preliminaries

Consider X_n being the set of all possible vectors of n binary input signals to a self-organizing system. The subset of X_n which is supposed to be classified by a self-organizing system is denoted by Y_n .

Let us distinguish the two following steps in the activities of a self-organizing system S :

1. The learning phase: The maximal number k of classes to be determined is supplied to S . After that, a sequence of unclassified objects is presented to the system. Based on that input S modifies itself such that it will classify objects according to a certain classification function $f : X_n \rightarrow \{0, \dots, k-1\}$.
2. The classification phase: The system gets as input some object $x \in X_n$. It outputs the value of the learned function $f(x)$.

We denote by $S[x_1, \dots, x_i](x)$ the value of the function f that emerged in S after providing S exactly with the sequence x_1, \dots, x_i of unclassified examples through S 's learning phase.

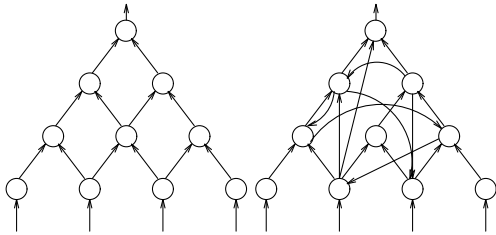


Figure 2: The left figure shows the initial net structure of a self-organizing neural network. After the learning phase the network has reorganized its internal structure as shown in the right figure. The new network structure should compute a desired classification function f .

Definition 1 A self-organizing system S is a network describable by the following items:

- a) the functionality of a single neuron. Often a certain threshold function of the sum of the weighted inputs to the neuron is proposed.
- b) the topological organization of a complete net consisting of a large number of neurons.
- c) The algorithm which controls the self-organization process through learning.

The complete system S then can be described by concatenating the descriptions of all its parts. Let the description of S be denoted by $descr(S)$.

2.1 Measuring the complexity of emerging structures

For measuring the complexity of emerging system structures, Kolmogorov's notion of the complexity of finite objects as introduced in Kolmogorov [5] is used. According to this idea, the complexity of a (classification) function f is the length of the shortest (binary encoded) program for a fixed universal Turing machine U which computes the function f . The length of that program is also called the **Kolmogorov complexity** of f and is denoted by $K(f)$. This notion has been introduced for investigations of learning complex functions within connectionist models of computation in Hoffmann [4]. Freivalds & Hoffmann [2] used the notion to investigate the relation between clustering and inductive inference.

The complexity $comp(S)$ of S is therefore defined as the Kolmogorov complexity of its description. I.e. the minimal encoding of the systems functionality. I.e.

$$comp(S) = K(descr(S)).$$

Kolmogorov complexity is used for the following considerations, since it is often claimed that neural networks or *self-organizing systems* are capable of acquiring complex functions.

For such subsymbolic approaches to learning and classification various more or less complicated and, as an unfortunate consequence, more or less incomprehensible models of computation have been proposed. Hence, Kolmogorov complexity is used in order to distinguish between seemingly complicated functions - distributed over the entire network - and really complex functions that have been acquired. I.e. functions that cannot be described essentially shorter in any other way.

In the following, possible classifications of the subset Y_n are investigated. For the purpose of measuring the complexity of a classification on Y_n , for each possible classification function f the class F_{f,Y_n} of classification functions which are equivalent in their values of Y_n will be considered. I.e., $F_{f,Y_n} = \{h | (\forall i \in Y_n) f(i) = h(i)\}$. Let $K_{Y_n}(f)$ denote the minimal Kolmogorov complexity among all classification functions in F_{f,Y_n} , i.e.

$$K_{Y_n}(f) = \min_{f' \in F_{f,Y_n}} K(f').$$

Hence, $K_{Y_n}(f)$ denotes the Kolmogorov complexity of the least complex description of the classification of the subset Y_n according to f .

2.2 Probabilistic setting

For the following probabilistic setting some probability distribution P_n on X_n for all n is assumed. Since objects are only allowed to be chosen from Y_n , $P_n(X_n \setminus Y_n) = 0$. Moreover,

$$(\forall x \in Y_n) P_n(x) > 0 \wedge \sum_{x \in Y_n} P_n(x) = 1.$$

In other words, Y_n represents the set of 'naturally' appearing objects. Thus, Y_n represents the kind of information that is potentially provided to the system.

Moreover, the values of an emerging classification function on the elements of $(X_n \setminus Y_n)$ are assumed to be of no relevance.

3 The complexity of emerging classification functions

This section investigates the *complexity* of classification functions emerging in self-organizing networks.

3.1 Monotonically growing classification competence

In the following, we require from a system S that it does not change its ‘mind’ about classification decisions for single objects once made.

Definition 2 A monotonically classifying self-organizing system S classifies incrementally the presented objects and never changes the classification of previously classified objects. I.e. S is monotonically classifying, if

$$(\forall k)(\forall j < k) S[y_1, \dots, y_k](y_j) = S[y_1, \dots, y_{k+1}](y_j).$$

Theorem 1 Assume an arbitrary but fixed $Y_n \subseteq X_n$ and a fixed probability distribution P_n on Y_n for arbitrary n . Then, for any monotonically classifying self-organizing system S holds the following: If for any sequence of objects randomly drawn according to P_n , S finally acquires the same target classification function f , then the Kolmogorov complexity of f is upper bounded as follows:

$$K_{Y_n}(f) \leq \text{comp}(S) + n + \text{const}.$$

Proof: The theorem is proved by contradiction. Assume S determines a classification function f with $K_{Y_n}(f) > \text{comp}(S) + n + \text{const}$. (n is the amount of information necessary for describing one object of X_n .) Then, there must be an object $y_1 \in Y_n$ as follows. Provide y_1 as the first element of a ‘randomly’ drawn sequence to S . Then, there must be another object $y_2 \in Y_n$, which cannot be classified according to f . This is since the amount of information provided so far ($\text{comp}(S) + n$) does not suffice for determining the function f . Since $K_{Y_n}(f)$ is by definition the smallest amount of information in order to compute all values of f on Y_n , there must be an $y_2 \in Y_n$ for which $f(y_2)$ differs from $S[y_1](y_2)$ for at least one appropriate y_1 . \square

The restriction of being a monotonically classifying system in theorem 1 may appear quite strong, although the change of classifications certainly cannot be accepted to an unlimited extent.

3.2 Approximating a classification function

However in the following, limits for approximations of complex target classification functions are considered. An approximation of a target function is defined as follows:

Definition 3 Assume for some n a fixed $Y_n \subseteq X_n$ and a fixed probability distribution P_n on Y_n .

Then, it is said a function f ε -approximates a target function f_t , if

$$\sum_{x \in \{x | x \in Y_n \wedge (f(x) \neq f_t(x))\}} P_n(x) \leq \varepsilon$$

Theorem 2 strongly upper-bounds the complexity of classification functions, which can be ε -approximated with a reasonable probability of success.

Theorem 2 For all $0 < \varepsilon < \frac{1}{2}$ and $0 < \delta < 1$, for an arbitrary self-organizing system S , an arbitrary but fixed sample size s and a target classification function f_t with a complexity

$$K_{Y_n}(f_t) > \text{comp}(S) + 2n + \text{const},$$

the following holds: If providing S with a sample of size s randomly drawn according to an arbitrary probability distribution P_n on X_n , the emerging classification function f in S will with probability of at least δ be **no** ε -approximation of f_t .

Proof: Construct a suitable probability distribution P_n on Y_n for which the theorem holds as follows. First of all, note that $|Y_n| > 2$. Moreover, the following proposition holds:

Proposition 1 For all possible self-organizing systems S , there are two objects $y_1, y_2 \in Y_n$ and a subset $Z = \{z_1, \dots, z_n\} \subseteq (Y_n \setminus \{y_1, y_2\})$ with the following property:

$$S[y_1, y_2](y_1) \neq S[y_1, y_2, z_1, \dots, z_m](y_1)$$

Proof: Assume, the proposition is false. Then, there exists a function f'_t which is equivalent to f_t on the set Y_n with $K_{Y_n}(f'_t) \leq \text{comp}(S) + 2n + \text{const}$. Hence, a contradiction. \square

I.e. there is an element $y_1 \in Y_n$, whose classification changes, when S gets the elements of the set Z as input. This fact is due to the complexity of f_t . I.e. $2n$ is the amount of information for describing y_1 and y_2 . Let P_n be given by $P_n(y_1) = P_n(y_2) = \frac{1-\gamma}{2}$ and $\forall i \in \{1, \dots, m\} P_n(z_i) = \frac{\gamma}{m}$ for some $0 < \gamma < 1$. γ can be chosen arbitrarily close to 0. Hence, for all δ there is an appropriate γ as follows: The probability that the complete set Z occurs in a sample of size s randomly drawn according to P_n is less than $1 - \delta$. I.e., y_1 will be misclassified with probability greater than $1 - \delta$, since it will be classified according to $S[y_1, y_2]$. \square

Theorem 2 indicates that the complexity of ‘meaningful’ classification functions, which can emerge based on unclassified examples only is rather limited.

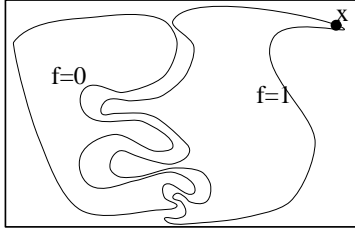


Figure 3: The shape of an appropriate function f separating the right from the left area is actually encoded by the particular location of the informative object x .

3.3 On the minimal sample size

The following theorem gives a minimal number of randomly drawn examples necessary in order to allow the emergence of a complex and desired structure in a self-organizing system. This is a best case consideration in both, the possible probability distribution and the assumed information provided by the encoding of the examples.

Theorem 3 Let be $k = \frac{K_{Y_n}(f_t) - \text{comp}(S)}{n}$. For all $0 < \delta < 1$, all self-organizing systems S , $k > 16, k > 16 \ln \frac{1}{\delta}$ if the sample size $|s|$ is less than

$$k^{\frac{3}{2}} \ln \frac{1}{\delta}$$

the following holds: In the self-organizing system S , with probability greater than δ a classification function f will emerge which classifies the set Y_n not according to f_t .

Proof: In order to provide sufficient algorithmic information to S , at least $k = \frac{K_{Y_n}(f_t) - \text{comp}(S)}{n}$ different examples have to be presented. Thus, the sample has to contain at least these k examples. The probability for the occurrence of all k examples in the sample has its maximum in the case where all k objects have the same probability to occur, i.e. if $P_n(x_i) = \frac{1}{k}$ for all $i \in \{1, \dots, k\}$. Thus, it remains to show that the following inequality holds for $|s| = k^{\frac{3}{2}} \ln \frac{1}{\delta}$: $P(\{x_1, \dots, x_k\} \subseteq s) \leq 1 - \delta$. In a random sequence of length r chosen from k equally likely objects the probability of having the entire set $\{x_1, \dots, x_k\}$ in s is less than

$\frac{\binom{k}{k} k^{r-k}}{k^r} = \frac{(r-k)^k}{k! k^k}$ let $r = \ln \frac{1}{\delta} k^{\frac{3}{2}}$ and plug it into the expression above: $\frac{\ln \frac{1}{\delta} k^{\frac{3}{2}} - k}{k! k^k} < \frac{k^{\frac{3}{2}} (\ln \frac{1}{\delta})^{\frac{k}{2}}}{k!}$. The latter one is less than $1 - \delta$ for $k \geq 16, k > 16 \ln \frac{1}{\delta}$. \square

Theorem 3 assumes indirectly that the necessary information for encoding the target function f_t is literally contained in the representation of some particular examples. This is certainly a rather optimistic assumption. See figure 3 for illustration.

4 Conclusion and further work

It has been shown, that the complexity of classification functions acquired under reasonable preconditions is seriously limited. Moreover, in a probabilistic setting - related to Valiant's PAC-learning model [7] - the complexity of a function that can be acquired does not exceed significantly the complexity of the used inference strategy. Finally, a bound on the minimal number of randomly drawn examples necessary in order that a complex structure may emerge has been given for the best possible case.

By investigating further the conditions under which a large degree of (meaningful) complexity can be acquired this line of research has the potential to answer questions concerning the necessary and/or sufficient conditions of the environment of biological systems which has supposedly been developed by evolution.

References

- [1] J. Buhmann and H. Kühnel. Complexity optimized data clustering by competitive neural networks. *Neural Computation*, 5:75–88, 1993.
- [2] R. Freivalds and A. G. Hoffmann. An inductive inference approach to classification. In *Proceedings of the 3rd Workshop on Analogical and Inductive Inference*, pages 187–196, Schloß Dagstuhl, Germany, October 1992. Springer-Verlag.
- [3] S. Grossberg. Adaptive pattern classification and universal recoding I & II. *Biological Cybernetics*, 23:121–134 & 187–202, 1976.
- [4] A. G. Hoffmann. On computational limitations of neural network architectures. In *Proceedings of the 2nd IEEE Symposium on Parallel and Distributed Processing*, pages 818–825, 1990.
- [5] A. N. Kolmogorov. Three approaches to the quantitative definition of information. *International Journal for Computer Mathematics*, 2:157–168, 1968. (originally published in Russian: Problemi peredachi informacii, vol. 1, No. 1, 1965, p. 3–11.).
- [6] H. Ritter, T. Martinez, and K. Schulten. *Neural Computation and Self-organizing Maps*. Addison & Wesley, 1992.
- [7] L. Valiant. A theory of the learnable. *Communications of the ACM*, 27:1134–1142, 1984.