Nonlinear feedforward networks with stochastic ouputs: infomax implies redundancy reduction

Jean-Pierre Nadal†§, Nicolas Brunel†and Nestor Parga‡

† Laboratoire de Physique Statistique de l'E.N.S.¶
Ecole Normale Supérieure
24, rue Lhomond, F-75231 Paris Cedex 05, France
‡ Departamento de Física Teórica
Universidad Autónoma de Madrid
Cantoblanco, 28049 Madrid, Spain

Abstract.

We prove that maximization of mutual information between the output and the input of a feedforward neural network leads to full redundancy reduction under the following sufficient conditions: (1) the input signal is a (possibly nonlinear) invertible mixture of independent components; (2) there is no input noise; (3) the activity of each output neuron is a (possibly) stochastic variable with a probability distribution depending on the stimulus through a deterministic function of the inputs; both the probability distributions and the functions can be different from neuron to neuron; (4) optimization of the mutual information is performed over all these deterministic functions. This result extends the one obtained in [1] where the case of deterministic outputs was considered.

PACS numbers: 87.30

§ To whom correspondence should be addressed.

¶ Laboratoire associé au C.N.R.S. (U.R.A. 1306), à l'ENS, et aux Universités Paris VI et Paris VII.

1. Introduction

Independent Component Analysis (ICA), and in particular Blind Source Separation (BSS), can be obtained from the maximization of mutual information, as first shown in [1]. This result was obtained for a deterministic processing system, with an arbitrary input-output relationship. Technically, a small additive noise was considered in order to define the mutual information between the inputs and the outputs variable. Then the zero noise limit was taken in order to extract the relevant "contrast" cost function for the deterministic case, which is nothing but the output entropy. The relevance for BSS was pointed out: in the particular case in which the inputs are linear combinations of independent random variables ("sources"), one can use a feedforward network (with no hidden layer), and nonlinear transfer functions; then the outputs of the system will give the independent components if both the weights and the transfer functions are adapted in such a way that mutual information is maximized.

The practical interest of this information theoretic based cost function was then demonstrated by [2, 3] in several BSS applications. Since then, it has also been realized [4, 5] that the cost function in the form written in [2] is in fact identical to the one derived several years before from a maximum likelihood approach [6].

In [1], the implication of the link between infomax and redundancy reduction in the modeling of sensory systems was emphasized. The main result was that, under some conditions, optimization of a nonlinear processing system implies the optimization of the linear part of the processing as if no nonlinearity was present. This may help to explain why *linear* models of visual systems lead to predictions of receptive fields and constrast sensivity curves that compare favorably with experimental data [7, 8, 9, 10]. The present study, in which *stochastic* nonlinear outputs are considered, provides additional support to this analysis.

More precisely, in the present work we extend the main result of [1] to the case where it is the *probability distribution* of each output which depends on the input variables through some deterministic function. In section 2.1 we define explicitely the class of models to be studied. As illustrative examples, we consider the case of the simplest feedforward architecture (the one used in BSS applications, but here with noisy outputs), and the case of a feedforward network of spiking neurons. In section 3, we first write the mutual information between the input and the output in a way that will be useful for our purpose; then we detail the proof of the fact that infomax leads to redundancy reduction - whenever redundancy reduction can indeed be obtained. In section 4 we illustrate our result on specific cases. In section 5 we shortly discuss what will or may happen in some cases where the derivation of section 3 does not apply (e.g. when input noise is present). Perspectives are given in the Conclusion.

2. The model class

2.1. Stochastic processing

We consider a system with an output of m units, devoted to the processing of some input signal $\mathcal{S}$. Our interest being in the modeling of neural processing, we will think of the system as a feedforward neural network with a layer of m output neurons responding to the stimulus $\mathcal{S}$. The model is more precisely defined as follows. We assume a well defined probability measure $d\rho[\mathcal{S}]$ on the space of stimuli (in the simplest case, $\mathcal{S}$ might be some N dimensional field, $\mathcal{S} = \{S_j, j = 1, ..., N\}$ and $d\rho[\mathcal{S}] = d^N S\ \rho[\mathcal{S}]$). For a given input $\mathcal{S}$, each neuron i ($i = 1, ..., m$) has its activity V_i computed according to some stochastic rule that depends on $\mathcal{S}$ but not on the activity of the other neurons: the probability of the output $\mathbf{V}$ given $\mathcal{S}$ is factorized:

$$Q(\mathbf{V} \mid \mathcal{S}) \ = \ \prod_i Q_i(V_i \mid \mathcal{S}) \tag{1}$$

The model is completely defined by the choice of the m conditional probabilities $Q_i(V_i \mid \mathcal{S})$ of observing the state V_i when the stimulus $\mathcal{S}$ is presented to the network. The dependency of this probability Q_i on the stimulus is assumed to be entirely through a scalar u_i, which is itself a *deterministic* function of the stimulus:

$$Q_i(V_i \mid \mathcal{S}) \ = \ Q_i(V_i \mid u_i[\mathcal{S}]),\ i = 1, ..., m. \tag{2}$$

Hence the class of models we will consider is finally defined by:

$$\begin{array}{rcl} \mathcal{S} & \rightarrow & \mathbf{u} = \{u_i[\mathcal{S}], i = 1, ..., m\}, \\ \mathbf{u} = \{u_i, i = 1, ..., m\} & \rightarrow & \mathbf{V} = \{V_i, i = 1, ..., m\} \\ & & \text{with probability } \prod_i Q_i(V_i \mid u_i) \end{array} \tag{3}$$

Let us first give specific examples of this general model.

2.2. Illustrative examples

A first particular example of network belonging to the above family, is the one of the simple feedforward network, with no hidden unit, responding to a multidimensional input $\mathcal{S} = \{S_1, ..., S_N\}$, with noisy additive outputs. In this case the activities of the output neurons are given by

$$\begin{array}{rcl} V_i & = & u_i[\mathcal{S}] \ + \ z_i, \\ u_i[\mathcal{S}] & = & f_i(h_i),\ \ i = 1, ..., m \end{array} \tag{4}$$

where the f_i are the (possibly non linear) transfer functions, and the h_i, modeling the postsynaptic potential (PSP), give the linear part of the processing. More precisely:

$$h_i[\mathcal{S}] = \sum_{j=1}^{N} J_{ij}\ S_j \tag{5}$$

the J_{ij} being the synaptic efficacies. Finally the z_i are independent noises (e.g., Gaussian noises). In this particular model, the conditional probability distributions

Q_i are obtained from the noise distributions (which may be different from neuron to neuron):

$$Q_i(V_i \mid u_i) = \Pr(z_i = V_i - u_i). \tag{6}$$

A second example is given by the same simple architecture but with a multiplicative noise, e.g.,

$$V_i \;=\; u_i[\mathcal{S}] \;+\; z_i\,\sqrt{u_i[\mathcal{S}]}. \tag{7}$$

The class defined by (3) includes also multilayer feedforward networks, where the activities of the output neurons in the last layer can be described by, say, an equation similar to (4). But this class concerns also cases of neurons with discrete outputs. Indeed, an interesting case is the one of spiking neurons. The simplest model in the class is the one of output neurons emitting spikes according to a Poisson process, with mean firing rates ν_i that are deterministic functions of the input stimulus [11, 12, 13]:

$$\nu_i \;=\; u_i[\mathcal{S}],\, i = 1, ..., N. \tag{8}$$

The information on the input signal is encoded in the numbers of spikes $k_i, i = 1, ..., m$ observed during some given time window t:

$$Q_i(k_i \;\text{ spikes emitted in } [0,t] \mid \nu_i) = \frac{(\nu_i t)^{k_i} \exp(-\nu_i t)}{k_i!} \tag{9}$$

Information processing by such neurons is studied in detail in [11, 12] for both short and large time windows. Such neuronal models give not only an example of discrete output distributions but also an example where the noise strength is stimulus dependent. This is more clearly seen in the large time limit where, for each output neuron i, k_i gives a good estimate of the mean firing rate ν_i: the empirical firing rate $V_i = k_i/t$ tends to be a Gaussian centered around ν_i, but with a variance that depends also on ν_i, in such a way that the activity V_i can be described by the equation (7).

2.3. Decomposition in PSP and transfer functions

Coming back to the general case defined by (3), it will appear that it is convenient to decompose each transformation $u_i[\mathcal{S}], i = 1, ..., m$ in two steps:

$$\mathcal{S} \;\rightarrow\; h_i \;=\; h_i[\mathcal{S}] \tag{10}$$

$$h_i \;\rightarrow\; u_i \;=\; f_i(h_i) \tag{11}$$

where $h_i[.]$ is some deterministic function of the stimulus, and f_i a real valued function of a single variable. Such a decomposition appears naturally in most neural models, in particular in the example presented in the previous section. By analogy with the simplest case (4), we will call the h_i's "PSP functions" (which in the general case need not to be linear functions of the stimulus), and the f_i's the "transfer functions".

It is clear that such a decomposition (10,11) of u_i in a (non necessarily linear) PSP function h_i and a transfer function f_i, always possible, is somewhat arbitrary:

indeed, if one chooses the function $u_i[.]$ and any invertible function $f_i[.]$, h_i is then defined as $h_i[\mathcal{S}] = f_i^{-1}[u_i[\mathcal{S}]]$. In fact, it is precisely this arbitrariness which we will use below, when we will ask for the maximization of information over all possible choices of functions $u_i, i = 1, ..., m$.

3. Maximization of the mutual information

3.1. The mutual information

The amount of information that the output layer, characterized by its activity $\{V_i, i = 1, ..., m\} = \mathbf{V}$, conveys about the input $\mathcal{S}$ is given by the *mutual information* I between the input and output distributions[14]:

$$I \ = \int d\rho[\mathcal{S}] \int d^m V Q(\mathbf{V} \mid \mathcal{S}) \ \ln \frac{Q(\mathbf{V} \mid \mathcal{S})}{p(\mathbf{V})} \tag{12}$$

where $p(\mathbf{V})$ is the output probability distribution. Since $Q(\mathbf{V} \mid \mathcal{S}) = Q(\mathbf{V} \mid \mathbf{u}[\mathcal{S}])$, with deterministic (but not necessarily invertible) transformation $\mathcal{S} \to \mathbf{u}[\mathcal{S}]$, the mutual information I between the V's and $\mathcal{S}$ is equal to the mutual information between the V's and the u's. To see this, one can rewrite I as

$$\begin{aligned} I \ = \int d^m u \left[\int d\rho[\mathcal{S}] \prod_i \delta(\ u_i \ - \ u_i[\mathcal{S}]\ ) \right] & \\ \times \int d^m V \ Q(\mathbf{V} \mid \mathbf{u}) \ \ln \frac{Q(\mathbf{V} \mid \mathbf{u})}{p(\mathbf{V})} & \end{aligned} \tag{13}$$

that is

$$I \ = \int d^m u \ \mathcal{P}(\mathbf{u}) \int d^m V Q(\mathbf{V} \mid \mathbf{u}) \ \ln \frac{Q(\mathbf{V} \mid \mathbf{u})}{p(\mathbf{V})} \tag{14}$$

where $\mathcal{P}(\mathbf{u})$ is the probability distribution of $\mathbf{u}$ induced by the stimulus distribution,

$$\mathcal{P}(\mathbf{u}) \ = \int d\rho[\mathcal{S}] \prod_i \delta(\ u_i \ - \ u_i[\mathcal{S}]\ ) \tag{15}$$

and $p(\mathbf{V})$ can also be written as

$$p(\mathbf{V}) \ = \int d^m u \ \mathcal{P}(\mathbf{u}) \ Q(\mathbf{V} \mid \mathbf{u}). \tag{16}$$

The r.h.s. of eq. (14) is precisely the mutual information between the V's and the u's.

The factorization of the output probability distribution given the stimulus, equation (1), that is

$$Q(\mathbf{V} \mid \mathbf{u}) = \prod_i Q_i(V_i|u_i) \tag{17}$$

allows to rewrite the mutual information in the following way:

$$I \ = \sum_i I_i \ - \ \mathcal{R} \tag{18}$$

where each I_i is the information conveyed by neuron i alone, and $\mathcal{R}$ is the redundancy contained in the set of the m outputs. More precisely the I_i are the individual mutual informations conveyed by a single neuron:

$$I_i \;=\; \int du_i\, \mathcal{P}_i(u_i) \int dV_i\; Q_i(V_i|u_i) \ln\left(\frac{Q_i(V_i|u_i)}{p_i(V_i)}\right) \tag{19}$$

where $\mathcal{P}_i(u_i)$ and $p_i(V_i)$ are the marginal probability distributions for neuron i,

$$\mathcal{P}_i(u_i) \;=\; \int d\rho[\mathcal{S}]\; \delta(\; u_i \;-\; u_i[\mathcal{S}]\;), \tag{20}$$

$$p_i(V_i) \;=\; \int du_i\; \mathcal{P}_i(u_i)\; Q_i(V_i|u_i). \tag{21}$$

The redundancy $\mathcal{R}$ is the Kullback divergence between the joint probability distribution $p(\mathbf{V})$ and the factorized distribution $\prod_i p_i(V_i)$:

$$\mathcal{R} \;=\; \int d^mV\; p(\mathbf{V})\; \ln\left(\frac{p(\mathbf{V})}{\prod_i p_i(V_i)}\right) \tag{22}$$

Since the mutual informations I_i are positive, and the redundancy $\mathcal{R}$ is also positive, the mutual information I will be maximized when the redundancy is as small as possible, and at the same time each individual mutual information is as large as possible. This shows already that the maximization of information transfer will lead to some redundancy reduction. However, we want to know if, whenever complete redundancy reduction is possible, this will indeed give the maximum of the mutual information. Indeed, one may wonder whether it is possible to maximize the I_i and minimize $\mathcal{R}$ at the same time, or whether it could be possible to accept a non zero redundancy in order to increase considerably the individual mutual informations I_i. The purpose of this section is to give a precise answer to these questions. Explicitely, we address the question of maximizing the mutual information (14) over all possible choices of functions $u_i, i = 1, ..., m$.

Before starting, several remarks are in order.

- Depending on the type of conditional probabilities Q_i, the optimization problem may not be well defined unless maximization is performed under some constraints (e.g., the mean output ativity is given). In addition, for a specific application other constraints may have to be taken into account. One should note also that possible constraints on u_i are implicitely contained in Q_i. For instance, stating that u_i belongs to $[0, 1]$ is the same as stating that $Q_i(.|u_i)$ is not defined (or equivalently is zero) outside the interval $[0, 1]$.

  In order to keep the discussion general, we will not distinguish between the cases with and without constraints: in all what follows *maximization* will be meant for *constrained maximization* for all cases for which a set of constraints has to be prescribed.

- What we have just said apply to constraints on the Q_i alone. Other type of constraints may prevent from reaching a factorized solution. In particular, in practical cases a specific architecture for the system is chosen (e.g. a multilayer feedforward neural network), which means that the functions $u_i[\mathcal{S}]$ belongs to a parametrized family of functions. In such cases the optimization has to be performed over the functions belonging to that family, and the derivation which follows may not apply.
  Indeed in what follows we will not put any restriction on the set of admissible functions (apart from those resulting from the definition of the Q_i's). Only at the end, in section 5.1, we will come back to the role of constraints.

3.2. Step 1: maximization of the individual informations

Let us consider an individual term I_i, as given by (19). This quantity depends on the conditional distribution $Q_i(.|.)$, and on the probability distribution $\mathcal{P}_i(.)$ of u_i:

$$I_i \; = \; I[Q_i, \mathcal{P}_i] \tag{23}$$

The probability Q_i is given: it defines completely the neuron model. We can consider the maximisation of (23) under all possible choices of $\mathcal{P}_i(.)$. Let us call $\mathcal{P}_i^{opt}$ a probability distribution which realizes the maximum of I_i. One should note that the maximum I_i^{opt} of the mutual information (as well as $\mathcal{P}_i^{opt}$) is a function of $Q_i(.|.)$ only:

$$I_i^{opt} \; = \; I[Q_i, \mathcal{P}_i^{opt}] \equiv C[Q_i] \tag{24}$$

In the context of Communication Theory, $Q_i(.|.)$ defines a channel, and $C[Q_i]$ is its Shannon capacity [14].

Now for a given choice of the function $u_i[\mathcal{S}]$ one has $I_i \leq C[Q_i]$. Since the redundancy $\mathcal{R}$ is positive, it is then clear that, for *any* choice of the functions $u_i, i = 1, ..., m$, we have the upper bound

$$I \leq \sum_{i=1}^{m} C[Q_i] \tag{25}$$

3.3. Step 2: minimization of the redundancy

Let us now assume that there exists at least one set of m functions $\mathcal{S} \to \{h_i^{fac}[\mathcal{S}], i = 1, ..., m\} = \mathbf{h}^{fac}$ such that the probability distribution induced on $\mathbf{h}^{fac}$ by $\mathcal{S}$,

$$\Psi(\mathbf{h}) \; = \; \int d\rho[\mathcal{S}] \prod_i \delta(\, h_i \; - \; h_i^{fac}[\mathcal{S}]\,), \tag{26}$$

factorizes:

$$\Psi(\mathbf{h}) \; = \; \prod_i \Psi_i(h_i) \tag{27}$$

Note that such a factorization is not trivial since, for a given set of PSP functions $h_i[.]$, the marginal distributions Ψ_i are given by:

$$\Psi_i(h_i) \;=\; \left[\prod_{k\neq i}\int dh_k\right]\;\Psi(\mathbf{h}) \;=\; \int d\rho[\mathcal{S}]\;\delta(\,h_i\;-\;h_i[\mathcal{S}]\,). \tag{28}$$

In fact the above hypothesis (27) is equivalent to state that the input signal is a (possibly nonlinear) invertible mixture of independent components.

We now choose the functions u_i as in (10,11), with these particular PSP functions h_i^{fac} which realize (27), and with transfer functions f_i which are yet arbitrary. Then clearly with such a choice the output distribution factorizes as well:

$$p(\mathbf{V}) \;=\; \prod_i\;\Big[\int dh_i\;\Psi_i(h_i)\;Q_i(V_i|f_i(h_i))\Big] \;=\; \prod_i\;p_i(V_i). \tag{29}$$

As a result, the redundancy $\mathcal{R}$ is zero.

3.4. Step 3: the optimal transfer functions

We can now construct the optimal functions u_i^{opt}. We have chosen the PSP functions in Step 2. We know that the upper bound (25) will be reached if each marginal distribution $\mathcal{P}_i$ of u_i can be set equal to the optimal distribution $\mathcal{P}_i^{opt}$ computed in Step 1. This is easily realized by choosing the transfer functions f_i.

Explicitely, this is obtained for each i by writing that the probability distribution for u_i, as induced by $\Psi_i(h_i)$, should be equal to $\mathcal{P}_i^{opt}(u_i)$:

$$\mathcal{P}_i^{opt}(u_i) \;=\; \int dh_i\Psi_i(h_i)\;\delta(u_i\;-\;f_i(h_i)). \tag{30}$$

This gives the "equalization" rule

$$\mathcal{P}_i^{opt}(f_i(h_i))\;|\frac{df_i}{dh_i}| \;=\; \Psi_i(h_i) \tag{31}$$

This result can be also written as an equation for the optimal function $\mathcal{S}\to u_i^{opt}[\mathcal{S}]$ (taking here f_i as a monotonic function):

$$\int^{u_i^{opt}} du\,\mathcal{P}_i^{opt}(u) \;=\; \int^{h_i[\mathcal{S}]} dh\;\Psi_i(h) \tag{32}$$

As a result, the mutual information is exactly equal to the upper bound (25),

$$I^{opt} \;=\; \sum_{i=1}^{m} C[Q_i] \equiv \mathcal{C}[\mathbf{Q} = \{Q_i, i = 1, ..., m\}] \tag{33}$$

This maximum, $\mathcal{C}[\mathbf{Q}]$, is thus the information capacity of the system. This capacity, equal to the sum of the m individual capacities, can be reached only if complete redundancy reduction can be performed.

4. Specific cases

We have shown that whenever redundancy reduction can be performed exactly, then the capacity (33) can be reached. Let us illustrate on this result on specific cases.

4.1. Deterministic limit

If one takes the limit of a deterministic output,

$$Q_i(V_i \mid u_i) \ \to \ \delta(V_i \ - \ u_i) \tag{34}$$

one recovers the result of [1]. In particular, in the case of a bounded output, say $0 \leq V_i \leq 1$, the optimal distribution $\mathcal{P}_i^{opt}(u_i)$ is the uniform distribution on $[0,1]$; then one recovers the standard equalization rule[15],[1],

$$\frac{d\, f_i}{dh_i} \ = \ \Psi_i(h_i) \tag{35}$$

which relates the optimal transfer functions to the marginal probability distributions of the independent fields h_is. We come back now to the general, stochastic, case.

4.2. Blind source separation

A particular case where a factorization as in (27) is possible is the one of a stimulus which is a linear mixture of N independent sources: this corresponds to the source separation problem[16, 1], for which one can use the linear model (5) with $m = N$. Then, there exists synaptic efficacies J_{ij}'s (linear filters) such that (27) is obtained, each h_i being proportional to one of the sources. Our result implies that, whatever the processing (output) noise level (e.g. the numerical resolution), such couplings will be found if the mutual information is maximized over both the J_{ij}'s and the transfer functions. This is the extension of the result in [1] to the case of stochastic outputs. One should note that the optimal transfer functions, for a non deterministic output, are still related to the marginal distributions of the independent components, although in a less straightforward way than in the deterministic case.

4.3. Spiking neurons

The very same result applies to spiking neurons. To illustrate this, let us consider the case of the model defined by (8) and (9), together with

$$\begin{aligned} \nu_i[\mathcal{S}] \ &= \ u_i[\mathcal{S}] \ = \ f_i(h_i[\mathcal{S}]),\ i = 1, ..., N, \\ h_i[\mathcal{S}] &= \sum_{j=1}^{N} \ J_{ij}\ S_j. \end{aligned} \tag{36}$$

With the hypothesis that the input is a linear mixture of independent sources, maximization of the mutual information will then give J_{ij}'s such that (27) is true, together with the optimal transfer functions f_i's, which can be computed for each i separetly.

The optimal distribution $\mathcal{P}^{opt}(u)$ is known for various cases [11, 12]. In particular for a neuron emitting spikes according to a Poisson process, with given smallest and largest frequencies ν_{min}, ν_{max}, in the large time limit $\mathcal{P}^{opt}(u)$ gives a uniform distribution for $\sqrt{u}$:

$$\mathcal{P}_{opt}(u) \;=\; \frac{1}{2} \frac{1}{\sqrt{\nu_{max}} - \sqrt{\nu_{min}}} \frac{1}{\sqrt{u}} \tag{37}$$

Each optimal transfer function f_i^{opt} can then be derived from the equalization rule (31) (with every $\mathcal{P}_i^{opt}(u)$ given by (37) if ν_{min} and ν_{max} do not depend on i). We will not discuss how biological systems manage to adapt both receptive fields and transfer functions - if they do. We just remind that adaption of receptive fields is a well established fact in particular in early visual systems, and that there are experimental evidences for the adaptation of transfer functions [15].

5. Discussion

In this section we comment on cases where the hypotheses needed for the derivation of section 3 are not fulfiled.

5.1. The effect of constraints

We come back to the maximization of the mutual information under constraints which do not reduce to constraints on the Q_i alone.

5.1.1. Constraints that allow factorization. We consider the case of constraints on the u_i's (that is on the architecture) which are such that there still exist PSP functions (satisfying the constraints) wich realize a factorial code. We argue that, in such case, a factorial code still gives a maximum of the mutual information, although it could be a local maximum. Suppose we have at least one family of m PSP functions h_i such that (27) is true. Then the redundancy is set to zero, its absolute minimal value. Now we are left with the optimization of the individual mutual informations over all possible transfer functions which are allowed by the constraints. One then obtains the maximum amount of information that can be conveyed by the network when the transfer functions are restricted to this particular set. This, of course, does not imply that one has obtained an absolute maximum of the mutual information. In fact, if there exists several families of PSP functions that lead to factorization, they may not lead to the same value of the mutual information.

Another aspect is the choice of the number m of outputs. If the input lies in a space of dimension N, one can extract at most N independent component. In the unconstrained case, the optimal number of output is then $m = N$. On the contrary, when resources are limited it may be that information cannot be conveyed on every component, so that there is an optimal number of outputs $m < N$ (that is more units would not convey more information).

5.1.2. Comparison with the linear case. The above discussion is well illustrated by the results obtained for a linear network with a Gaussian input[8, 9, 17]. In that case maximization of the mutual information has to be performed under some constraints. In fact, one can see that the constraints play the same role as non linear transfer functions [1], and this allows us to make a comparison with non linear networks.

Let us consider two types of constraints: one on the outputs (e.g. the mean output variance is given), and one on the synaptic efficacies (e.g. the sum of the norms of the coupling vectors is given). In both cases, qualitatively the maximum of the mutual information is obtained by (see [8, 17] for details):

(i) first, performing a principal component analysis: one finds the PSP functions h_i's giving a factorial code, each h_i giving the projection of the input onto the ith principal component;

(ii) second, one multiplies each component by a weight, say x_i, in order to satisfy the constraint; this is the gain control analogous to the choice of the transfer function.

Hence for both type of constraints, the network is able to find the independent components (that is here the principal components), and the global maximum is obtained by a factorized solution. There is however a qualitative difference between these two type of constraints. If one fixes the mean output variance, the same amount of information is extracted from each one of the m component. This is in agreement with the result of section 3.1 obtained for nonlinear units: a constraint on the output variances is a constraint on the Q_i only; if all the Q_i are identical (and a same constraint is applied to all of them), then all the capacities $C[Q_i]$ are equal. This is not the case for a constraint on the couplings (which is a constraint on the architecture, on the u_is). There, the information conveyed by each component will depend on the value of the associated eigenvalue. In addition, some weight x_i may be zero, which means that there is in fact an optimal number of output units $m \leq N$. The reason is that at a large enough noise level the constraint cannot be fulfilled for principal components with small eigenvalues, so that no information can be extracted from these components. This is again an effect of putting a constraint on the architecture.

5.2. Maximization when zero redundancy cannot be achieved

The exact link between infomax and redundancy reduction remains unclear when one cannot find a set of functions that factorizes the input distribution as in (27). However, we point out that it is always possible to minimize the redundancy $\mathcal{R}$ over all possible functions $\mathbf{u}$, *under the constraint that, for each i, the marginal probability distribution of u_i is equal to $\mathcal{P}_i^{opt}(u)$*. To do so, one decomposes each functions u_i as a transfer function f_i applied onto a PSP function h_i. For a given choice of the PSP functions, we have the induced probability distribution $\Psi(\mathbf{h})$, and the resulting marginal distributions $\Psi_i(h_i), i = 1, ..., m.$ (see (28)). For each neuron i, one can achieve $\mathcal{P}_i^{opt}(u)$ by choosing the appropriate transfer function according to the equalization rule (31). In this way, one obtains $I_i = C[Q_i]$, and one has

$$I \;=\; \mathcal{C}[\mathbf{Q}] \;-\; \mathcal{R} \tag{38}$$

The redundancy $\mathcal{R}$ is an implicit function of the probability distribution of $\mathbf{u} = \{u_i = f_i(h_i), i = 1, ..., m\}$. Since we choose the transfer functions according to (31), the f_i are functions of the h_i, and thus $\mathcal{R}$ is an implicit function of the functions $h_i[\mathcal{S}], i = 1, ..., m$.

The representation $u_i = f_i(h_i)$ might lead to a practical way - an algorithm - for performing the optimization. More importantly, this representation allowed us to show that it is *always* possible to satisfy a constraint -in fact essentially any constraint - on the marginal probability distributions.

5.3. Plausible effect of a nonzero input noise

The results obtained so far concern the zero input noise case. In [1], an expansion at first order in the input noise shows that the main result is still valid at this order, the noise providing a scale with which to choose among different solutions that all lead to a factorial code. The situation will be different with a finite input noise, in which case the factorization (1) of the output distribution given the input is no more true in general.

Let us however speculate on the possible effect of a finite input noise. In the case of a linear system with a Gaussian input, infomax and redundancy reduction have been studied in detail for arbitrary levels of input and output noises [8, 9, 17]. As we already said, the qualitative result is that optimal processing is obtained by performing principal component analysis, assigning to each component i a (possibly zero) weight x_i that depends on the corresponding eigenvalue and the noise level [8, 17]. For the linear network, the weights can be given two interpretations: x_i can be seen a gain control, as mentionned above, or as a redundancy introduced in that particular channel i - in fact an equivalent solution is obtained by having a certain number (of order x_i) of output neurons for each component i. It would be interesting to know whether a similar result holds for nonlinear systems and non Gaussian inputs: one may expect mutual information to be maximized if independent components are separated, and then some redundancy is added - that is, say, several output neurons are extracting a same independent component. Further work is clearly necessary in order to see whether any general statement can be derived on the effect of input noise.

6. Conclusion

We have obtained the strong result that, for a processing system defined as in (3), maximization of the mutual information leads to full redundancy reduction whenever the input signal is a (possibly nonlinear) invertible mixture of independent components. More precisely, this result is the extension of the one obtained for the deterministic case [1] to a broader class of models, where the outputs activities can be stochastic variables. In particular it applies to spiking neurons. It also implies that for performing blind source separation one can use the mutual information as a cost function even if the numerical resolution on the outputs is very poor.

Several lines of research are suggested by considering cases where the conditions, under which the result is obtained, are not fulfilled. In particular, we commented on the case of constraints onto the architecture, which may even prevent from finding independent components. This will happen for an invertible nonlinear mixture of independent components if the inverse cannot be computed with the chosen neural architecture. Finally we discussed shortly the plausible role of input noise, in which case some redundancy is clearly required in order to increase the signal to noise ratio for each independent component. It is then of interest to study a network with a number of output cells larger than the dimension of the stimulus (input) space. A typical case on which we are presently working[13] is the coding of a scalar by a population of neurons.

Acknowledgements

This work has been partly supported by the French-Spanish program ”Picasso”.

[1] J.-P. Nadal and N. Parga, Nonlinear neurons in the low-noise limit: a factorial code maximizes information transfer. *NETWORK*, 5:565–581, 1994.

[2] A. Bell and T. Sejnowski, An information-maximisation approach to blind separation and blind deconvolution. *Neural Computation*, 7:1129–1159, 1995.

[3] A. Bell and T. Sejnowski, The 'independent components' of natural scenes are edge filters. *Vision Research* 37(23): 3327-3338, 1997.

[4] J.-P. Nadal and N. Parga, Redundancy reduction and independent component analysis: Algebraic and adaptive approaches. *Neural Computation*, 9(7):1421-1456, 1997.

[5] J-F Cardoso, Infomax and maximum likelihood for source separation. *IEEE Signal Processing Letters*, 4(4):112-114, 1997.

[6] D.-T. Pham, Ph. Garrat, and Ch. Jutten, Separation of a mixture of independent sources through a maximum likelihood approach. In *Proc. EUSIPCO*, pages 771–774, 1992.

[7] J.H. van Hateren, Theoretical predictions of spatiotemporal receptive fields of fly lmcs, and experimental validation. *J. Comp. Physiology A*, 171:157–170, 1992.

[8] R. Linsker, Self-organization in a perceptual network. *Computer*, 21:105–17, 1988.

[9] J. J. Atick, Could information theory provide an ecological theory of sensory processing. *NETWORK*, 3:213–251, 1992.

[10] Y. Dan, J. J. Atick, and R. C. Reid, Efficient coding of natural scenes in the lgn: experimental test of a computational theory. *The Journal of Neuroscience* 16: 3351–3362, 1996.

[11] R. B. Stein, The information capacity of nerve cells using a frequency code. *Biophys. Journal* 7:797–826, 1967.

[12] N. Brunel and J.-P. Nadal, Optimal tuning curves in a simple model of spiking neurons. proceedings of ESANN'97 (16-17 April, 1997), M. Verleysen Ed., D-facto publications (Brussels).

[13] N. Brunel and J.-P. Nadal, Mutual information, Fisher information and population coding. LPSENS preprint 1997, submitted to Neural Computation.

[14] R. E. Blahut, *Principles and Practice of Information Theory.* Addison-Wesley, Cambridge MA, 1988.

[15] S. B. Laughlin, A simple coding procedure enhances a neuron's information capacity. *Z. Naturf.*, C 36:910–2, 1981.

[16] P. Comon, Independent component analysis, a new concept ? *Signal Processing*, 36:287–314, 1994.

[17] P. Del Giudice, A. Campa, N. Parga, and J.-P. Nadal, Maximization of mutual information in a linear noisy network: a detailed study. *NETWORK*, 6:449–468, 1995.