Maximum-entropy principle and neural networks that learn to construct approximate wave functions

L. Diambra* and A. Plastino†
Departamento de Física, Universidad Nacional de La Plata, Casilla de Correo 727, 1900 La Plata, Argentina
(Received 17 April 1995)

By recourse to a maximum-entropy-based method for the training of perceptrons, we show that an appropriately prepared network is able to construct approximate ground state wave functions of good quality with the sole knowledge of a few expectation values, even if nothing is known concerning the interaction potential.

PACS number(s): 87.10.+e, 03.65.Ge, 05.20.−y

I. INTRODUCTION

The last years have witnessed a surge of applications of neural networks to scientific problems. Neural networks have exhibited remarkable properties for data processing, having found use in a wide variety of environments such as identification and classification of physical objects, time series, and image reconstruction.

Most network designs involve neural network multilayer, exclusively feedforward networks of analog units. In this architecture, values of an appropiate set of variables are encoded in the input pattern as the state of the input-layer units and these patterns are analyzed by one or more hidden layers [1–5]. The ensuing pattern of output activities gives, in appropriately encoded form, the results of the network's classification process, its version of the completed image, or its computed values or assignments for contingent physical quantities. Given a representative set of examples, together with an effective learning rule, such systems can indeed capture the essential physical relationships and correlations that govern the pertinent class of input-output associations, as evidenced both by accurate performance on the training examples and by reliable generalizations or predictions for novel input patterns. The trained networks are able to *predict*, i.e., to produce outputs corresponding to *new* inputs (that are not included in the training set) on the basis of an adequately selected working *hypothesis*. This hypothesis is, of course, represented by a set of synaptic weights W_i that, when appropriately implemented, yields good results for the examples of the training set. Much effort has consequently been devoted to the task of developing suitable training algorithms that are able to adjust the synaptic weights so as to enable the network to *infer* the correct answer when presented with a new input. Of course, one wishes for algorithms that accomplish such a goal within a reasonable (CPU) time and with a not too large number of examples. The most popular learning methods involve minimization of an energy (or cost) function that depends upon the set of training patterns. Diverse approaches to this end include simmulated annealing [6], genetic algorithms [7], and gradient methods [1,8,9]. A cost function is minimized by recourse to an algorithm that incorporates a degree of randomness, as represented by a "temperature" or by "mutations."

In the present effort we shall focus our attention upon the selection of the working hypothesis (WH). We wish here to show that by recourse to an adequate WH, a neural network can learn to construct rather good approximate ground-state wave functions (GSWF) on the basis of the knowledge of just a few expectation values, *even if one knows nothing concerning the interaction potential.* To this end, a recently devised information theory (IT) approach is utilized [10] that provides us with the functional form, which is of a typical aspect characteristic of Jaynes' IT approach [10]. A number of parameters are to be determined on the basis of the available informational output, by solving a complicated set of coupled partial differential equations [10].

It would certainly be highly desirable to be in a position of entirely bypassing the orthodox, cumbersome, and somewhat tedious process of determining these parameters. This motivates us to try to ascertain whether a better way can be found by recourse to the networks learning abilities, which should enable them to gather ("capture"), from a set of suitable examples, the essentials of the system's characteristic correlations. The concomitant procedure would be in better agreement with the spirit of any IT approach than the conventional route, which passes through a system of partial differential equations, as *only the input informational supply would be utilized.*

To this end, we shall employ, in *two widely different senses*, Jaynes' maximum-entropy principle (ME) [11–15]: (1) so as to optimize the WH in the learning process, on the one hand, and (2) in order to build up approximate GSWF, on the basis of a limited amount of information on the other one. In what follows we briefly describe, separately, these two ingredients. None of them is new. What we claim is that the juxtaposition of them yields an *original* recipe for devising networks that are able to build up approximate GSWF.

*Electronic address: diambra@venus.fisica.unlp.edu.ar
†Electronic address: plastino@venus.fisica.unlp.edu.ar

The paper is organized as follows: the optimal WH is discussed in Sec. II and applied in Sec. IV in order to build up, on the basis of *scarce* information, approximate GSWF. The underlying ME quantal approach employed to this end is described in Sec. III. Section V discusses and summarizes the present results.

II. OPTIMIZING THE WORKING HYPOTHESIS

A. The general idea

This is to be accomplished according to Ockham's razor, i.e., with the minimum number of assumptions compatible with the available input. Our essential tools are those of the IT approach to statistical mechanics, as embedded in the ME Principle [11–15]. A learning protocol can be developed in this fashion that will be applied to the simplest layered network: the *perceptron.*

Consider a network with N input units ξ_i connected to an output unit ζ whose state is determined according to $\zeta = g(h)$, where $g(x)$ is the transfer function of the output neuron, which is assumed invertible, and $h = \xi \cdot \mathbf{W}$ is the membrane potential. For each set of weights $\mathbf{W}$ the student perceptron (SP) maps ξ on ζ. We train the SP with a set of P inputs ξ^μ, with $\mu = 1, ..., P$ and the corresponding appropriate outputs $\zeta_0^\mu(\xi)$, as provided by a teacher perceptron (TP) with weights $\mathbf{W}_0$. Of course, the SP and the TP share an identical arquitecture. It is obvious that

$$g^{-1}(\zeta_0^\mu) = \xi^\mu \cdot \mathbf{W}, \tag{1}$$

where ξ^μ is an "input-patterns" matrix and $g^{-1}(\zeta_0^\mu)$ is a vector of components $\left[g^{-1}(\zeta_0^1), g^{-1}(\zeta_0^2), ..., g^{-1}(\zeta_0^P)\right]$, given by the output patterns, which constitute our available information.

Our *leit-motiv* is that of introducing an IT algorithm in order to determine the weights $\mathbf{W}$ on the basis of an *incomplete* information supply [in the present situation, $\mathrm{ran}(\xi^\mu) < N$, in general]. To this end we take advantage of the ME pseudoinverse technique recently reported by Baker-Jarvis [16]. In order to infer weights consistent with Eq. (1) we shall assume that *each set of weights* $\mathbf{W}$ *is realized with probability* $P(\mathbf{W})$ (our essential IT ingredient). In other words, we introduce a (normalized) probability distribution over the possible sets $\mathbf{W}$. Of course,

$$\int P(\mathbf{W}) d\mathbf{W} = 1, \tag{2}$$

where $d\mathbf{W} = dW_1 dW_2 \cdots dW_N$. Expectation values $\langle W_i \rangle$ are defined in the fashion

$$\langle W_i \rangle = \int P(\mathbf{W}) W_i d\mathbf{W}, \tag{3}$$

and a *relative* entropy is, in the usual way [11–13], associated to the probability distribution, namely,

$$S = -\int P(\mathbf{W}) \ln\left(\frac{P(\mathbf{W})}{P_0(\mathbf{W})}\right) d\mathbf{W}, \tag{4}$$

where $P_0(\mathbf{W})$ is an appropriately chosen *a priori* distribution [11–13]. This entropy is to be maximized, subject to the constraints (1). Our *central* idea is that of reinterpreting these equations in a rather particular fashion, i.e., we recast them as follows:

$$g^{-1}(\zeta_0^\mu) = \xi^\mu \cdot \langle \mathbf{W} \rangle, \tag{5}$$

where explicit account is taken of the fact that we are assumed to be dealing with *many* sets of weights, each one being realized with a given probability.

As customary [12], one is then led to freely maximizing the quantity

$$S' = -\int \left\{ P(\mathbf{W}) \ln\left(\frac{P(\mathbf{W})}{P_0(\mathbf{W})}\right) + \alpha P(\mathbf{W}) + (\xi^\mu)^t \lambda \mathbf{W} P(\mathbf{W}) \right\} d\mathbf{W}, \tag{6}$$

where α and λ are Lagrange multipliers associated, respectively, to the normalization condition (2) and to the constraints (1). Variation of S' with respect to $P(\mathbf{W})$ immediately gives

$$P(\mathbf{W}) = \exp[-(1+\alpha)] \exp(-\mathbf{\Gamma} \cdot \mathbf{W}) P_0(\mathbf{W}), \tag{7}$$

where $\mathbf{\Gamma} = (\xi^\mu)^t \lambda$. As in statistical mechanics, one conveniently defines the partition function Z

$$Z = \int d\mathbf{W} \exp(-\mathbf{\Gamma} \cdot \mathbf{W}) P_0. \tag{8}$$

A choice is now to be made concerning the *a priori* probability distribution P_0 [11–13]. Here we select a Gaussian P_0, i.e., choose it to be proportional to $\exp\left(-\frac{\mathbf{W} \cdot \mathbf{W}}{2a^2}\right)$, with a (formally) free parameter a. The results, however, do not depend upon the value of a.

It is now an easy matter to explicitly evaluate the partition function. We find

$$Z = \prod_{i=1}^{N} \left(2a^2\pi\right)^{1/2} \exp\left(\frac{a^2\Gamma_i^2}{2}\right), \tag{9}$$

so that with (3) and the distribution (7) one has, for the $\langle W_i \rangle$, the convenient expression

$$\langle W_i \rangle = -2a^2 \Gamma_i. \tag{10}$$

Notice that the definition of $\mathbf{\Gamma}$ and the constraints (1) allow one to express the $\langle W_i \rangle$ in the fashion

$$\langle \mathbf{W} \rangle = \mathcal{P}_I[\xi^\mu] g^{-1}(\zeta_0^\mu), \tag{11}$$

where $\mathcal{P}_I[\xi^\mu] = (\xi^\mu)^t [\xi^\mu (\xi^\mu)^t]^{-1}$ is the Moore-Penrose pseudoinverse [17]. The most probable configuration of weights [compatible with the constraints (1)] is thus given in terms of a pseudoinverse matrix (that of ξ^μ). This resembles (but is in fact distinct from) the Personnaz-Guyon-Dreyfus [18,19] projection rule for memorizing (without errors) correlated patterns in the Hopfield model. Notice that with the choice (11) the training er-

ror vanishes. Additionally, the set of "inverse" examples $\{-\xi^\mu, -\zeta_0(\xi^\mu)\}$ possesses an associated distribution identical to that given by (7). Consequently, $-\zeta_0(\xi^\mu)$ is that output produced by the network for the input $-\xi^\mu$.

B. The specific implementation

For our present purposes we shall construct the membrane potential following the work of Matus and Perez [20]. This involves recourse to membrane potentials with a high-order dependence on the firing rates. The membrane potential depends upon the state of the input layer neurons $h = h(\xi_1, \xi_2, ..., \xi_N)$, and the functional dependence being arbitrary, we expand h as a power series (any fixed value ξ_i^0 can be chosen for the zeroth-order term)

$$h = h^0 + \sum_i^N \frac{\partial\, h^0}{\partial \xi_i}\left(\xi_i - \xi_i^0\right) + \frac{1}{2}\sum_{ij}^N \frac{\partial^2\, h^0}{\partial\xi_i \partial\xi_j}\left(\xi_i - \xi_i^0\right) \times \left(\xi_j - \xi_j^0\right) + \frac{1}{6}\sum_{ijk}^N \frac{\partial^3\, h^0}{\partial\xi_i\partial\xi_j\partial\xi_k}\left(\xi_i - \xi_i^0\right) \times \left(\xi_j - \xi_j^0\right)\left(\xi_k - \xi_k^0\right) + \cdots, \tag{12}$$

so that, after conveniently rearranging things, we get an ordinary looking expansion of the kind

$$h = \theta^0 + \sum_i w_i \xi_i + \sum_{ij} w_{ij}\xi_i\xi_j + \sum_{ijk} w_{ijk}\xi_i\xi_j\xi_k + \cdots, \tag{13}$$

where θ^0 is the threshold defined by $\theta^0 = h^0 + \sum_i^N \frac{\partial\, h^0}{\partial \xi_i}\xi_i^0 + \frac{1}{2}\sum_{ij}^N \frac{\partial^2\, h^0}{\partial\xi_i\partial\xi_j}\xi_i^0\xi_j^0 + \cdots$, and the linear weights $w_i = \sum_i^N \frac{\partial\, h^0}{\partial\xi_i} + \sum_j^N \frac{\partial^2\, h^0}{\partial\xi_i\partial\xi_j}\xi_j^0 + \cdots$, etc. For the sake of simplicity, we restrict our analysis to the first terms in the expansion (13), this approximation being, in general, good enough in many cases [20].

In order to determine the weights by recourse to the method described above, we define a vector δ whose components are (1) the $\{\xi_i\}$, (2) the second-order terms $\{\xi_i^2, \xi_i\xi_j\}$, and (3) the third-order terms $\{\xi_i^3, \xi_i^2\xi_j, \xi_i\xi_j\xi_k\}$. The matrix of the P inputs ξ^μ is associated, via the matrix δ^μ, with the corresponding outputs ζ_0^μ. This allows the input patterns to "capture" the essential correlations of the system in a rather natural fashion. With the weights w_i, w_{ij}, and w_{ijk} we build up that particular vector $\mathbf{W}$ that verifies the relation $g^{-1}(\zeta_0^\mu) = \delta^\mu \cdot \mathbf{W}$. The ME algorithm prescribes that the most probable configuration of weights, compatible with the relevant constraints, is given by $\mathbf{W} = \mathcal{P}[\delta^\mu]\, g^{-1}(\zeta_0^\mu)$. The most probable configuration of weights [compatible with the constraints (1)] is thus given in terms of a pseudoinverse matrix (that of δ^μ).

The above results are immediately generalized to networks with several output neurons. The appropriate map is given by $\zeta_j = g\left(\xi \cdot \mathbf{W}_j\right)$ and the weights become

$$\mathbf{W}_j = \mathcal{P}[\delta^\mu]\, g^{-1}\left(\zeta_j^\mu\right), \tag{14}$$

which is the ME recipe for the quantal learning process to be here described.

III. MAXIMUM-ENTROPY APPROXIMATE WAVE FUNCTIONS

Our aim now is to apply the above described metodology in order to *infer*, on the basis of some appropriate information, probability distributions associated with quantum GSWF [10], *when nothing is known concerning the interaction potential.* In the present, introductory instance, we shall concentrate our efforts upon the simplest situation: the one-dimensional case.

For the convenience of the reader we proceed now to give a brief recapitulation of the quantal ME approach of Canosa, Plastino, and Rossignoli [21] (the second ingredient anticipated in the Introduction). The possibility of employing just a reduced set of (relevant) expectation values in order to describe the most salient features of a physicaw eystem is, of course, the *raison d'être* of statistical mechanics. In more recent developments based upon IT, the pertinent statistical operator is built up by recourse to Jaynes' ME [12–15]. However, if one wishes to apply a similar treatment in order to describe *pure* states one is immediately confronted with a quite serious difficulty: for these states the von Neumann-Shannon entropy identically vanishes.

The way to go if one wishes to perform "statistical inferences" on a WF was reported in [21]. In later works, several applications have been successfully implemented in several fields [10,21–29].

A theoretical construct, the so-called "quantal" entropy S_Q is introduced [10,21], which is defined by

$$S_Q = -\int |\psi|^2 \ln |\psi|^2\, d\tau. \tag{15}$$

The probabilistic distribution is, however, that associated with the squared modulus of the pertinent WF [10]. The *a priori* distribution is here the uniform one [10,21] and becomes just a normalization constant. S_Q is, of course, basis dependent and not invariant with respect to an unitary transformation that changes the basis. The basis to be utilized is determined by the nature of the expectation values (EV) to be given as the "information supply"

$$\langle f \rangle = -\int |\psi|^2\, f\, d\tau. \tag{16}$$

It is assumed that these EV refer to *commuting* operators, so that one employs just that basis that diagonalizes them. Maximization of S_Q, subject to the constraints posed by the input information confronts one with an extremalization problem identical to that faced originally by Jaynes [see Eq. (6)], and one ends up with a set of equations formally identical to the ME ones. The modulus squared of the WF is of the familiar exponential aspect

$$\psi(x) = Q(x) \exp\left[-\frac{1}{2}\left(\lambda^0 + \sum_{l=1}^{L} \lambda^l x^l\right)\right], \qquad (17)$$

where the Lagrange multipliers are to be obtained by solving the set of coupled equations

$$\frac{\partial \lambda^0}{\partial \lambda^l} = -x^l, \qquad (18)$$

and $Q(x)$ is a suitable polynomial [10]. For ground-state wave functions (no nodes) one immediately gets from (17) the WF itself [21]. For excited states, somewhat more elaborate considerations apply [28], which will not interest us here.

We wish here to avoid facing the system (18) by recourse to suitable trained networks. In building up approximate WF with them, of course, the Lagrange multipliers turn out to be essential ingredients in the training process.

The approach we have just outlined has proved to be quite useful in describing both the ground states and excited states a variety of many-body systems, for an unrestricted range of coupling constants [10,21–29]. In particular, it has been shown to provide one with many-body wave functions that are of a better quality, in several different environments, than the Hartree-Fock [23], the BCS [24], or the random-phase-approximation ones [26]. This "quantal" ME approach is thus a very reliable one, appropriate for training neural networks.

IV. SIMPLE APPLICATIONS

As a first application we consider two different one-dimensional problems, namely, the anharmonic oscillator $[V(x) = \alpha x^2 + \beta x^3 + \gamma x^4]$ and the Morse potential $\{V(x) = A[1 - \exp(-x)]^2\}$. The corresponding Hamiltonian can be generally written as

$$\widehat{H} = \frac{\widehat{P}^2}{2} + V\left(\widehat{X}\right). \qquad (19)$$

Our aim is to approximately reconstruct the ground-state GSWF of (19) with the sole knowledge of a few expectation values $\{\langle x^l \rangle, l = 1, ..., L\}$. The ME prescription is [10]

$$\psi(x) = Q_0(x) \exp\left[-\frac{1}{2}\left(\lambda^0 + \sum_{l=1}^{L} \lambda^l x^l\right)\right], \qquad (20)$$

where $Q_0(x)$ is an appropriate polynomial and L is the number the input expectation values. The parameters λ^l are to be evaluated ("learned") by our network and λ^0 is a normalization constant.

The performance of our algorithm has been studied in the case of networks with (a) linear transfer functions $g(x) = x$ and (b) membrane potentials h characterized by a high-order (second and third) dependence upon the firing rates. The training set is given by pairs {input, output}. The output content is that of the vectors $\lambda_\mu = \{\lambda_\mu^l,\ l = 1, ..., 4\}$, where $\mu = 1, ..., P$ is the example label (we deal with P examples). Within a predetermined interval (see Table I), we randomly choose (with uniform probability) the λ^l and to each vector λ we associate a GSWF of the form (20). The input is prepared in the following fashion. To first order in the couplings, it is given simply by the moments of the GSWF ψ_μ associated with the appropriate set λ_μ

$$\langle x_\mu^l \rangle = \int \psi_\mu^*(x)\ x^l\ \psi_\mu(x)\ dx, \qquad l = 1, \ldots, 4. \quad (21)$$

To second order we *add* products of these moments as well, of the form: $\left\{\left\{\langle x^l \rangle, \langle x^l \rangle \langle x^m \rangle\right\}, \left\{\lambda^l\right\}\right\}$, and, to third order, we also incorporate *"threefold"* products of the type: $\left\{\left\{\langle x^l \rangle, \langle x^l \rangle \langle x^m \rangle, \langle x^l \rangle \langle x^m \rangle \langle x^n \rangle\right\}, \left\{\lambda^l\right\}\right\}$, with $\mu = 1, \ldots, P$.

In order to evaluate the performance of our algorithm, the generalization error E_g is defined in terms of some deviation measure ϵ between the desired outputs λ_{exact}^l and the actual outputs λ^l corresponding to the given inputs $\langle x^l \rangle$. Of course, changes in the λ^l affect the GSWF, for different l , in distinct ways (with diverse "intensities"). In studying the concomitant errors we must therefore appropriately weigh the distinct contributions. We do this weighing with reference to the expectation values associated to the λ^l and define the "distance" with respect to the desired output (the deviation measure referred to above) as

$$\epsilon = \frac{1}{2} \sum_{l=1}^{4} \left[\left|\langle x^l \rangle\right| \left(\lambda^l - \lambda_{\text{exact}}^l\right)^2\right]. \qquad (22)$$

The generalization error is the average of the distance (22) over new examples (not belonging to the training set) $E_g = \langle \epsilon \rangle$. In this paper the average is evaluated over 30 new examples. It is seen in Fig. 1 that, as higher-order couplings are added, E_g significantly diminishes (up to a factor 10). E_g becomes smaller and smaller as the number of examples augments, until a saturation plateau is reached (which is different for each type of network). The associated, "critical" number of examples P_{crit} is larger the higher the coupling order, although

TABLE I. Range of possible λ^l values, employed in order to train the neural network.

Multipliers	λ^1	λ^2	λ^3	λ^4
$\lambda_{\min}$	-1.500	-0.700	-0.500	0.300
$\lambda_{\max}$	1.500	0.700	0.350	1.100

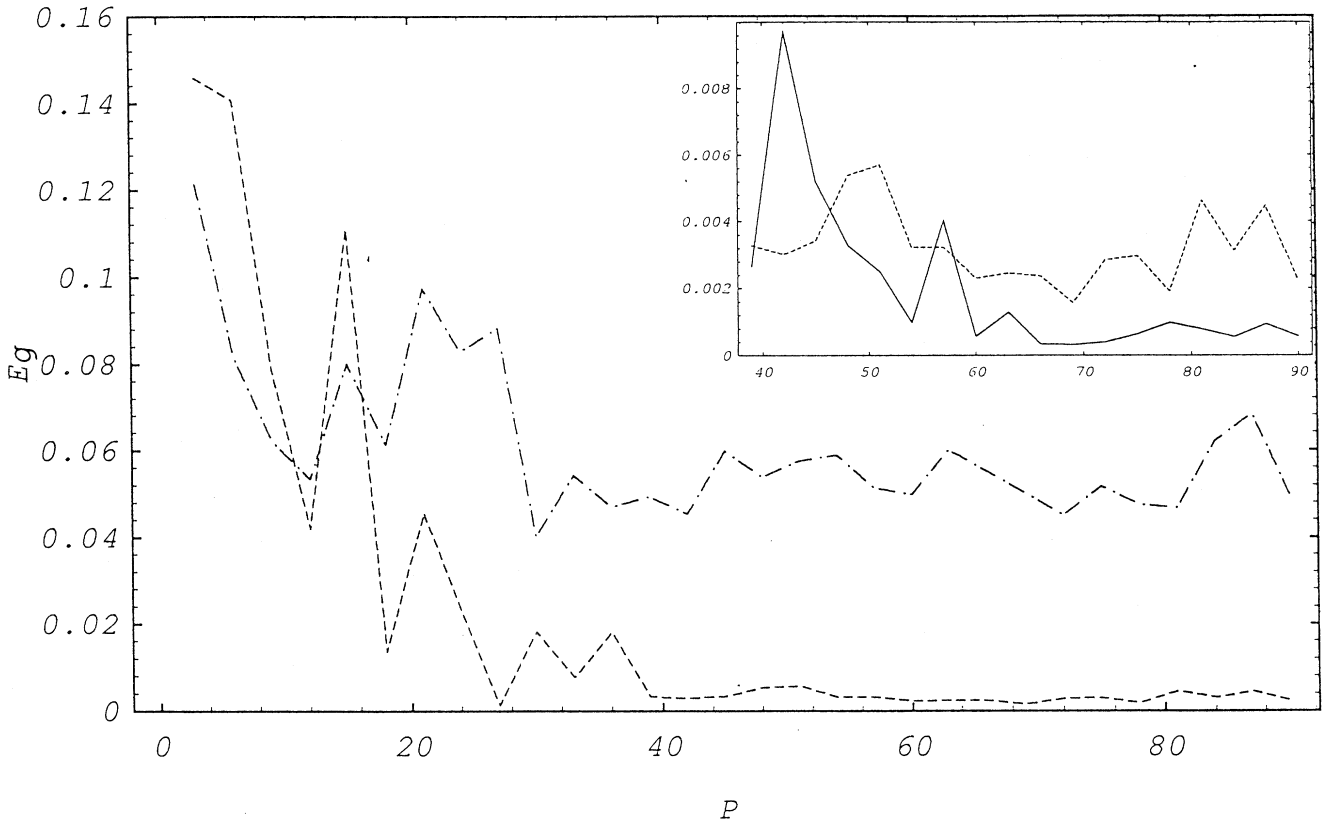

FIG. 1. Generalization error computed with 30 new inputs vs number of the examples P for three types of networks (they differ in the coupling *orders* included). The dot-dashed line corresponds to the inclusion of just first-order coupling. The results represented by the dashed one include second-order coupling. Inset: Generalization error for a net that includes up to third-order coupling (solid line). Comparison is made with second-order results.

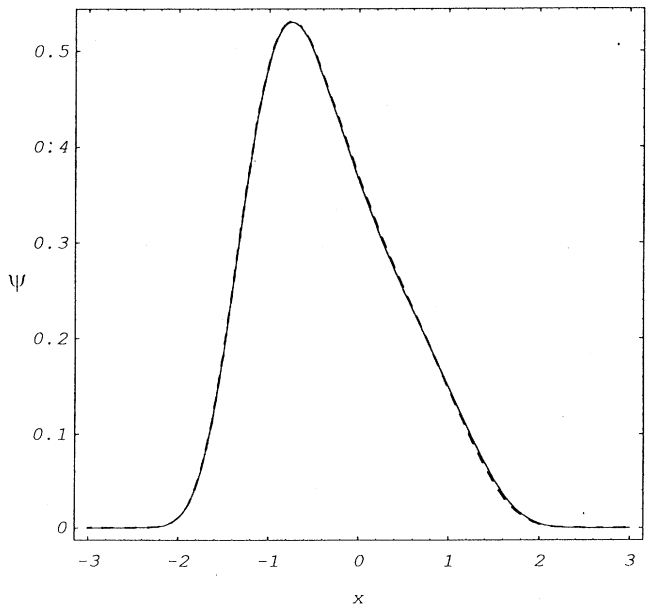

FIG. 2. The approximate GSWF for an anharmonic oscillator (dashed line) inferred by a third-order-coupling neural network trained with 63 examples is compared to the exact wave function (solid line).

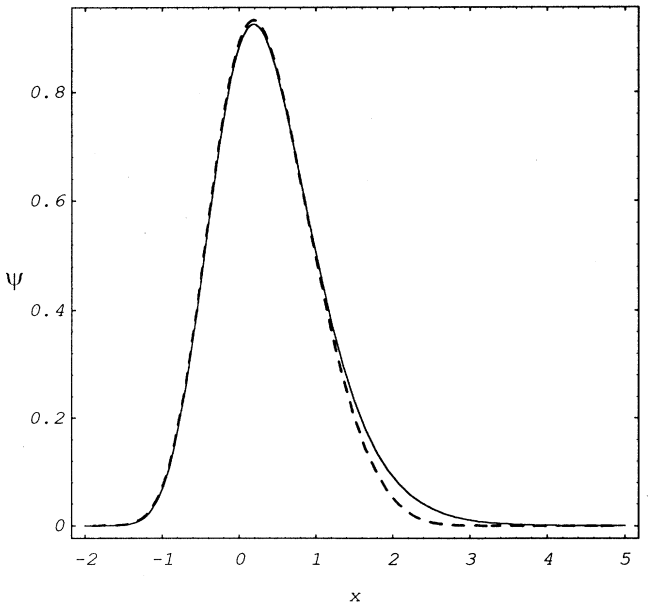

FIG. 3. The approximate GSWF for the Morse potential (dashed line) inferred by a third- order-coupling neural network trained with 63 examples is compared to the exact wave function (solid line).

this quantity is always of the order of the number of neurons N. In Figs. 2 and 3 we display inferred GSWF for each type of potential, comparing them to the exact Schrödinger ones.

For somewhat more involved examples we shall now focus our attention upon the radial Schrödinger equation (three dimensions) and tackle first the Coulomb potential and then a more complex radial one. For a radial three-dimensional potential $U(r)$, we face the radial equation

$$\left[\frac{d^2}{dr^2} + U(r) + E\right] R(r) = 0, \tag{23}$$

with $U(r) = V(r) + \frac{l(l+1)}{2r^2}$, i.e.,

$$\left[\frac{d^2}{dr^2} + V(r) + \frac{l(l+1)}{2r^2} + E\right] R(r) = 0. \tag{24}$$

Here the radial part of the wave function is of the ME form $[\psi_r(r) = R(r)/r]$

$$\psi_r(r) = r^l \exp\left[-\frac{1}{2}\left(\lambda^0 + \sum_{k=1}^{L} \lambda^k r^k\right)\right]. \tag{25}$$

As promised, we deal here with the Coulomb case and the more involved one posed by the potential

$$V(r) = a_2 r^2 + a_3 r^3 + a_5 r^5 + a_8 r^8, \tag{26}$$

with arbitrary a_2, a_3, a_5, a_8 $(a_8 > 0)$.

In Figs. 4 and 5 we display inferred GSWF for each of these two types of (radial) potentials, comparing them to the exact Schrödinger ones. An excellent agreement is obtained.

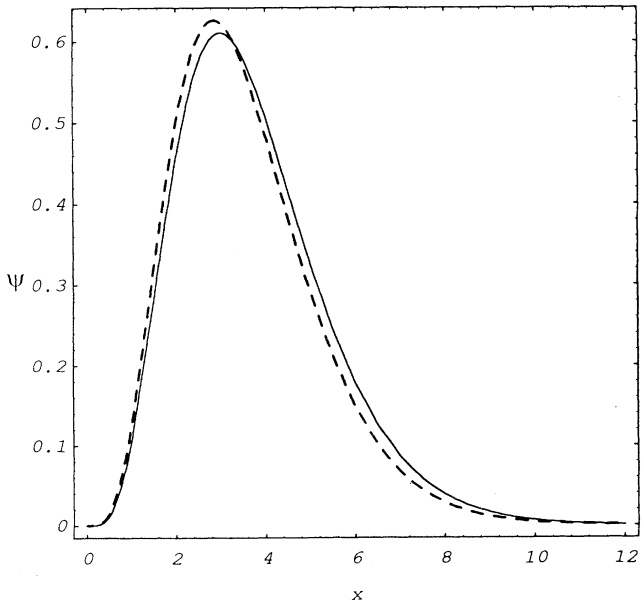

FIG. 4. The approximate GSWF for the Coulomb potential (dashed line) inferred by a third-order-coupling neural network trained with 63 examples is compared to the exact wave function (solid line).

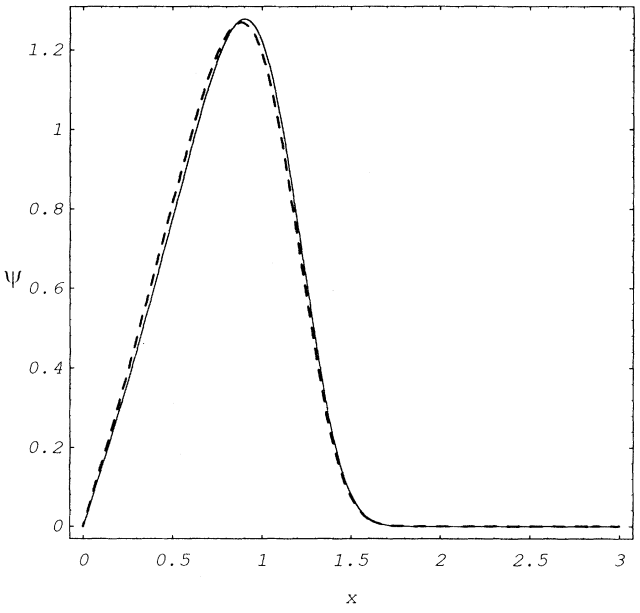

FIG. 5. The approximate GSWF for the potential (26) (dashed line) inferred by a third-order-coupling neural network trained with 63 examples is compared to the exact wave function (solid line).

V. DISCUSSION AND CONCLUSIONS

We have shown in this paper that by recourse to an appropriate, IT-based pseudoinverse technique, a neural network can be trained so as to infer GSWF when provided with a reduced set of relevant expectation values. We specially emphasize the fact that *one needs no knowledge at all concerning the associated interaction potential.*

A remarkable fact is also to be noted: the *small* quantity of input moments needed for the obtainment of approximate GSWF of excellent quality. This is certainly a notable facet of our approach, which shows that one is here dealing with a rather potent *learning* algorithm.

Our results can still be improved by recourse to couplings of an order higher than the third, specially in connection with the most "influential" λ^l, i.e., (λ^3, λ^4).

Our formalism can be easily applied to a variety of problems. We stress the fact that our learning rule, obtained by recourse to a ME technique, guarantees perfect learning of the given examples (zero training error) *and* yields an excellent performance (the generalization error diminishes in a linear fashion). This ME approach is much better than the one obtained by simply minimizing the training error. The latter also guarantees perfect learning but is rather poor at "generalizing."

Summing up, a new, ME approach to do elementary quantum mechanics in a neural network has been introduced that seems to offer promissory perspectives.

[1] F. Rosemblatt, in *Principles of Neurodynamics* (Spartan, New York, 1962).
[2] D. E. Rumelhart and J. L. McClelland, in *Parallel Distributed Processing* (MIT, Cambridge, MA, 1988).
[3] H. S. Seung, H. Sompolinsky and N. Tishby, Phys. Rev. A **45**, 6056 (1992).
[4] M. Opper and D. Haussler, Phys. Rev. Lett. **66**, 2677 (1991).
[5] T. Watkin, A. Rau, and M. Biehl, Rev. Mod. Phys. **65**, 499 (1993).
[6] S. Kirkpatrick, C. Gellat, and M. Vecci, Science **220**, 671 (1983).
[7] J. Holland, in *Evolution, Learning and Cognition*, edited by Y. S. Lee (World Scientific, Singapore, 1988).
[8] D. B. Parker, MIT Technical Report No. TR-47 1985 (unpublished).
[9] Y. Le Cun, in *Disordered Systems and Biological Organization*, edited by E. Bienenstock, F. Fogelman, and G. Weisbuch (Springer, Berlin, 1986).
[10] A. R. Plastino and A. Plastino, Phys. Lett. A **181**, 446 (1993).
[11] C. E. Shannon and W. Weaver, in *The Mathematical Theory of Communication* (University of Illinois Press, Chicago, IL, 1949).
[12] E. T. Jaynes, Phys. Rev. **108**, 171 (1957).
[13] R. D. Levine and M. Tribus, *The Maximum Entropy Principle* (MIT Press, Boston, MA, 1978).
[14] A. Katz, in *Principles of Statistical Mechanics* (Freeman, San Fransisco, 1967).
[15] N. Agmon, Y. Alhassid, and R.D. Levine, in *The Maximum Entropy Formalism*, edited by R. D. Levine and M. Tribus (MIT Press, Cambridge, 1979).
[16] J. Baker-Jarvis, J. Math. Phys. **30**, 302 (1989).
[17] A. Albert, in *Regression and Moore-Penrose Pseudoinverse* (Academic, New York, 1972).
[18] L. Personnaz, I. Guyon, and G. Dreyfus, Phys. Rev. A **34**, 4217 (1986).
[19] T. Kohonen, in *Self-organization and Associative Memory* (Springer-Verlag, Berlin, 1984).
[20] I. J. Matus and P. Perez, Phys. Rev. A **43**, 5683 (1991).
[21] N. Canosa, A. Plastino, and R. Rossignoli, Phys. Rev. A **40**, 519 (1989).
[22] N. Canosa, R. Rossignoli, and L. Diambra, Phys. Lett. A **185**, 133 (1994).
[23] N. Canosa, A. Plastino, and R. Rossignoli, Nucl. Phys. A **512**, 550 (1990).
[24] N. Canosa, A. Plastino, R. Rossignoli, and H. G. Miller, Phys. Rev. C **45**, 1162 (1992).
[25] L. Arrachea, N. Canosa, A. Plastino, M. Portesi, and R. Rossignoli, Phys. Rev. A **45**, 7104 (1992).
[26] N. Canosa, A. Plastino, and R. Rossignoli, Nucl. Phys. **A550**, 453 (1992).
[27] L. Arrachea, N. Canosa, A. Plastino, and R. Rossignoli, Phys. Lett. A **176**, 353 (1993).
[28] M. Casas, A. Plastino, and A. Puente, Phys. Rev. A **49**, 2312 (1994).
[29] M. Casas, A. Plastino, and A. Puente, Phys. Lett. A **184**, 385 (1994).