Interacting neural networks

R. Metzler and W. Kinzel
Institut für Theoretische Physik, Universität Würzburg, Am Hubland, D-97074 Würzburg, Germany

I. Kanter
Minerva Center and Department of Physics, Bar Ilan University, 52900 Ramat Gan, Israel

(Received 6 March 2000)

Several scenarios of interacting neural networks which are trained either in an identical or in a competitive way are solved analytically. In the case of identical training each perceptron receives the output of its neighbor. The symmetry of the stationary state as well as the sensitivity to the used training algorithm are investigated. Two competitive perceptrons trained on mutually exclusive learning aims and a perceptron which is trained on the opposite of its own output are examined analytically. An ensemble of competitive perceptrons is used as decision-making algorithms in a model of a closed market (El Farol Bar problem or the Minority Game. In this game, a set of agents who have to make a binary decision is considered.); each network is trained on the history of minority decisions. This ensemble of perceptrons relaxes to a stationary state whose performance can be better than random.

PACS number(s): 84.35.+i, 07.05.Mh, 02.50.Le

Simple models of neural networks describe a wide variety of phenomena in neurobiology and information theory. Neural networks are systems of elements interacting by adaptive couplings which are trained by a set of examples. After training they function as content addressable associative memory, as classifiers, or as prediction algorithms. Using methods of statistical physics many of these phenomena have been elucidated analytically for infinitely large neural networks [1,2].

Most studies of feed-forward neural networks have concentrated on a single network learning a fixed rule, which is usually a second network, the so-called teacher. The teacher network is presenting examples, sets of input/output data, and the student network is adapting its weights to this set of examples. In an on-line training scenario each example is presented only once, hence training is a dynamical process [3,4]. The teacher network may also generate a time series of output numbers [5,6], and the student learns by following the time series. The weights of the teacher network are fixed in this scenario.

Many phenomena in biology, social science, and computer science may be modeled by a system of interacting adaptive algorithms (see, e.g., [7]). However, little is known about general properties of such systems. In this paper we derive an analytic solution of a system of interacting neural networks. Each network is a simple perceptron with an N-dimensional weight vector. These networks receive an identical input vector, produce output bits, and learn from each other. In Sec. I each network is trained by the output of its neighbor, with a cyclic flow of information. By iterating the training step for randomly chosen input vectors, the dynamical process relaxes to a stationary state. In the limit of $N\rightarrow\infty$ we describe the process by ordinary differential equations for a few order parameters, similiar to the usual student/teacher scenario [3,8]. We identify the symmetries of the stationary state and find phase transitions when increasing the learning rate of the training steps.

In Secs. II and III we study different training scenarios with two interacting perceptrons and various learning algorithms. In Sec. IV we apply the system of interacting networks to a problem of game theory called the Minority Game, which is derived from the El-Farol Bar problem [9,10]. In this game, we consider a set of agents who have to make a binary decision. Each agent wins only if he/she belongs to the minority of all decisions. This process is iterated. Each agent has to develop an algorithm which makes a decision according to the history of the global minority decisions. The problem recently received a lot of attention in the context of statistical physics [11]. Here we follow a unique approach: Each agent uses a perceptron for making his/her decision, and each perceptron is trained on the minority of all output bits.

I. MUTUAL LEARNING, SYMMETRIC CASE

In this section we investigate a system of interacting neural networks as follows: several identical networks are arranged on an oriented ring. All networks receive an identical input and produce different output according to their weight vectors. Each network is trained by the output of its neighbor on the ring. This process is iterated until a stationary state is reached in which the norms and angles between the weight vectors no longer change. We are interested in the properties of this stationary state.

We consider the simplest feed-forward networks, an ensemble of K simple perceptrons, which are represented by N-dimensional weight vectors $\mathbf{w}_i$ $(i=1,\ldots,K)$ and which map a common input vector $\mathbf{x}$ onto binary outputs $\sigma_i=\mathrm{sgn}(\mathbf{x}\cdot\mathbf{w}_i)$. As order parameters we use the norms $w_i=|\mathbf{w}_i|$ and the respective overlaps $R_{ij}=\mathbf{w}_i\cdot\mathbf{w}_j$ or $\cos(\theta_{ij})=\mathbf{w}_i\cdot\mathbf{w}_j/w_iw_j$. When only two perceptrons are considered, the subscript is dropped: $\cos(\theta)=\mathbf{w}_1\cdot\mathbf{w}_2/w_1w_2$. The components of the input vector (or pattern) are Gaussian with mean 0 and variance 1, yielding $\mathbf{x}\cdot\mathbf{x}=O(N)$.

The updates are of the form

$$\mathbf{w}_i^{+}=\mathbf{w}_i+(\eta_i/N)f(\sigma_i,s)s\,\mathbf{x} \tag{1}$$

for unnormalized weights or

$$\mathbf{w}_i^+ = \frac{\mathbf{w}_i + (\eta_i/N) f(\sigma_i, s) s\, \mathbf{x}}{|\mathbf{w}_i + (\eta_i/N) f(\sigma_i, s) s\, \mathbf{x}|} \tag{2}$$

for normalized $\mathbf{w}_i$. The $+$ denotes a quantity after one learning step, η_i is the learning rate, s is the desired output, and $f(\sigma_i, s)$, the so-called weight function, defines the learning algorithm. We mostly use $f=1$ (the Hebbian rule, called H from now on) and $f=\Theta(-\sigma_i s)$ (the perceptron learning rule, abbreviated P [1]), and the respective variations where the $\mathbf{w}_i$ are kept normalized, denoted as H_N and P_N, respectively.

We derive differential equations for the order parameters in the thermodynamic limit $N\to\infty$ by taking the scalar product of the update rules and introducing a time variable $\alpha = p/N$, where p is the number of patterns shown so far. We use the analytic tools which were previously developed for the teacher/student scenario [4,3]. If the order parameters are self-averaging (see [12] for criteria of self-averaging in this context), integrating over the distribution of patterns gives deterministic differential equations for the order parameters as $N\to\infty$. The required averages are listed in the Appendix.

A. Perceptron learning rule

We first restrict ourselves to two perceptrons that try to come to an agreement by learning the output of the respective other perceptron. For rule P with identical learning rates $\eta_1 = \eta_2 = \eta$, the update rule is

$$\begin{aligned} \mathbf{w}_1^+ &= \mathbf{w}_1 + \frac{\eta}{N} \mathbf{x}\, \sigma_2 \Theta(-\sigma_1 \sigma_2), \\ \mathbf{w}_2^+ &= \mathbf{w}_2 + \frac{\eta}{N} \mathbf{x}\, \sigma_1 \Theta(-\sigma_1 \sigma_2). \end{aligned} \tag{3}$$

The sum of both vectors is conserved under this rule: if a learning step takes place, it has the same direction and absolute value, but different signs for the two vectors. This conservation can be used to link w_1 and w_2 to $\cos(\theta)$: assuming that $w_1 = w_2 = w$ and starting from $\theta_0 = \pi/2$, simple geometry gives $w_0/\sqrt{2} = \cos(\theta/2) w$. The conservation is also visible in the differential equations that can be derived using the described formalism:

$$\frac{dw_1}{d\alpha} = -\frac{\eta}{\sqrt{2\pi}}[1-\cos(\theta)] + \frac{\eta^2 \theta}{2 w_1 \pi}, \tag{4}$$

$$\frac{dw_2}{d\alpha} = -\frac{\eta}{\sqrt{2\pi}}[1-\cos(\theta)] + \frac{\eta^2 \theta}{2 w_2 \pi}, \tag{5}$$

$$\frac{dR}{d\alpha} = \frac{\eta}{\sqrt{2\pi}}[1-\cos(\theta)](w_1 + w_2) - \eta^2 \frac{\theta}{\pi}. \tag{6}$$

If the right-hand side of Eqs. (4) and (5) vanish, so does Eq. (6). There is a curve of fixed points of the system given by the equation

$$w = \frac{\eta}{\sqrt{2\pi}} \frac{\theta}{1-\cos(\theta)}. \tag{7}$$

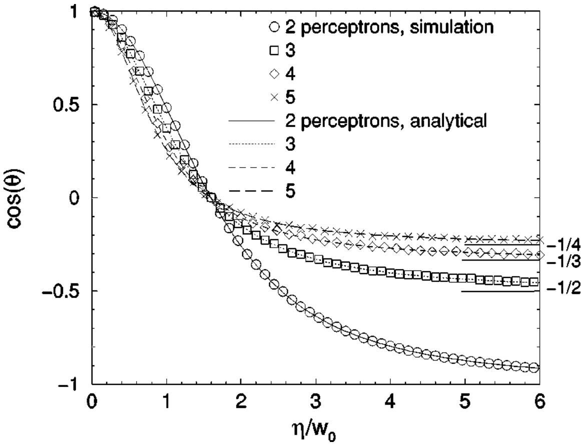


FIG. 1. Mutual learning with rule P (see Secs. I A and I C): comparison between Eqs. (7) and (10) and the stationary state in simulations with $N=100$ and $\alpha > 75$.

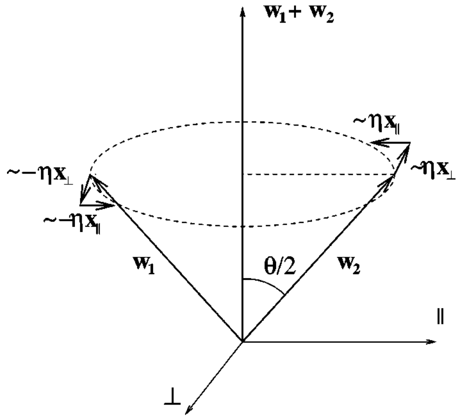
Using the relation $w = w_0/[\sqrt{2}\cos(\theta/2)]$, this can be solved numerically to give the fixed point of $\cos(\theta)$ as a function of the scaled learning rate η/w_0, as shown in Fig. 1. For small learning rates, the perceptrons come to good agreement, while large η leads to antiparallel vectors.

Geometrically, this can be understood as follows: each learning step has a component parallel to the plane spanned by $\mathbf{w}_1$ and $\mathbf{w}_2$, which decreases the distance between the vectors, and a perpendicular component, which increases the distance (see Fig. 2). Equilibrium is reached when a typical learning step no longer changes the angle, i.e., the vectors stay on a cone around $\mathbf{w}_1 + \mathbf{w}_2$. The radius of this cone increases with growing η.

B. Perceptron learning with normalized weights

A similar calculation can be done for the perceptron learning rule with normalized weights (P_N), where the length w_i of the weight vectors is set to 1 after each step. The perceptrons move on a hypersphere of radius 1; in equilibrium, the average learning step leads back onto that sphere before the vectors are normalized again.

We derive the following differential equation for $R = \cos(\theta)$:



$\mathbf{w_1}+\mathbf{w_2}$
$\sim\eta\mathbf{x}_\parallel$
$\sim\eta\mathbf{x}_\perp$
$\sim-\eta\mathbf{x}_\perp$
$\sim-\eta\mathbf{x}_\parallel$
$\mathbf{w_1}$
θ/2
$\mathbf{w_2}$
∥
⊥

FIG. 2. Mutual learning with rule P: sketch of the geometrical interpretation. See Sec. I A.

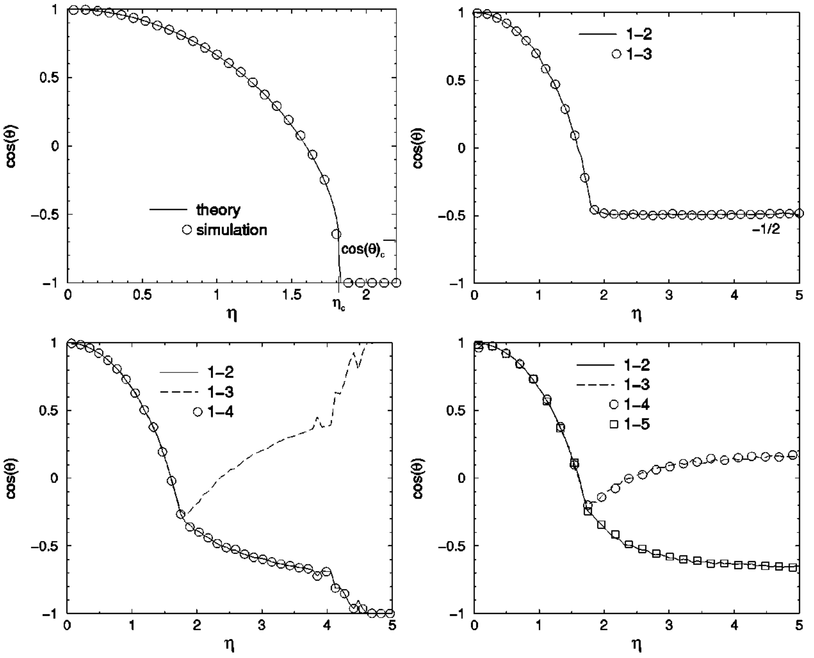


FIG. 3. Mutual learning with rule P_N (cf. Secs. I B and I C): the system follows Eq. (9) for $\eta < \eta_c$. Simulations used $N = 100$.

$$\frac{dR}{d\alpha}=(R+1)\left(\sqrt{\frac{2}{\pi}}\,\eta(1-R)-\eta^2\frac{\theta}{\pi}\right). \tag{8}$$

Fixed points are $R=1$, $R=-1$, and

$$\frac{\eta}{\sqrt{2\pi}}\,\frac{\theta}{1-\cos(\theta)}=1. \tag{9}$$

It is not a coincidence that this is equivalent to Eq. (7) if w is set to 1. The fixed point of Eq. (9) at $R=1$ is repulsive; the one at $R=-1$ is unstable for $\eta<4/\sqrt{2\pi}\cong 1.60$. A solution of Eq. (9) can only be found for $\eta \leqslant \eta_c \cong 1.816$, which corresponds to $\cos(\theta)\cong -0.689$.

Simulations show that the system relaxes to the fixed point given by Eq. (9) for $\eta<\eta_c$ and jumps to $R=-1$ for larger η (see Fig. 3). This behavior shows the characteristics of a first-order phase transition.

Hence for small learning rates the two perceptrons relax to a state of nearly complete agreement, $\theta \approx 0$. Increasing η leads to a nonzero angle between the two vectors up to $\theta \approx 133°$. At this rate the system jumps to complete disagreement, $\theta = 180°$.

C. Mutual learning on a ring

The mutual learning scenario can be generalized to K perceptrons: perceptron i learns from perceptron $i+1$ if they disagree, with cyclic boundary conditions. Under rule P, the total sum of vectors is conserved again: as many perceptrons take a step in one direction as in the opposite.

Performing the necessary averages for the equations of motion would involve Gaussian integrals over $K-1$ correlated variables with Θ functions—it is not clear to us whether this can be done analytically in general cases. However, we find in simulations that the fixed point for rule P is completely symmetric: there is only one angle θ between all pairs of perceptrons. Assuming that relation (7) still holds, and using the conservation of $\Sigma \mathbf{w}_i$, one can derive

$$\frac{\eta}{\sqrt{2\pi}}\,\frac{\theta}{1-\cos(\theta)}=\frac{w_0}{\sqrt{1+(K-1)\cos(\theta)}}. \tag{10}$$

The largest angle that the perceptrons can take is $\cos(\theta) = -1/(K-1)$, corresponding to a K-cornered hypertetrahedron. This happens when $|\Sigma \mathbf{w}_i|$ is negligible with respect to w_i. Simulations confirm that Eq. (10) holds, as can be seen in Fig. 1

Similar to the case of two networks, all perceptrons agree with each other for small learning rate $\eta \to 0$. For larger rates the system relaxes to a state of high symmetry where all mutual angles between the K weight vectors are identical, $\theta_{ij}=\theta$. Note that the symmetry is higher than the topology of the flow of information (the ring). For high rates $\eta \to \infty$ the system relaxes to a state of maximal disagreement, i.e., the largest possible mutual angle θ that is still compatible with a symmetric arrangement.

For rule P_N, the sum of the weights is not preserved. The fixed point of the dynamics follows the curve for two normalized weights described by Eq. (9) in a completely symmetric configuration. When the hypertetrahedron angle is reached and $\Sigma \mathbf{w}_i$ vanishes, the symmetry is partly broken. There are now different angles to nearest neighbors, next nearest neighbors, etc., so the angles split up into $(K-1)/2$ different branches for odd K and $K/2-1$ for even K. Note that the system still has the symmetry of the ring.

With odd K, increasing η increases the angle between nearest neighbors, up to some limit value. This angle is not the maximum nearest-neighbor angle allowed for by the geometric constraints, but seems to decrease with increasing K.

In the case of even K, simulations show a second transition at some higher value of η, where the vectors split into two antiparallel clusters, thus maximizing the nearest-neighbor angle. The learning rate at which this transition typically appears during the run of the program increases with N. The conclusion is that the antiparallel fixed point is not stable in the $N \to \infty$ limit, but de facto stable in simulations because the self-averaging property of the ordinary differential equations (ODEs) breaks down at this point.

One may ask which symmetries survive if the perceptrons are allowed to have different individual learning rates. A close look reveals that for rule P, there is a more general conserved quantity: $\Sigma_i^K \mathbf{w}_i/\eta_i$. Simulations show that the angles θ_{ij} again relax to a completely symmetric configuration depending on the average η and the initial value of the new conserved quantity, while the norms w_i are proportional to the respective learning rates η_i. For rule P_N, variations in the learning rates not only lead to slightly different curves for each of the angles with individually different η_c, they also suppress the transition to the antiparallel state that is observed for even K.

D. Hebbian learning

The reason why P and P_N lead to antiparallel orientation of the weight vectors for larger learning rates is that they concentrate on cases where the networks disagree. Algorithms that reinforce what both networks agree on are more successful, as can be seen for rule H for two perceptrons.

The differential equations are

$$\frac{dw_i}{d\alpha}=\eta\sqrt{\frac{2}{\pi}}\cos(\theta)+\frac{\eta^2}{2w_i},$$

$$\frac{dR}{d\alpha}=\eta\sqrt{\frac{2}{\pi}}(w_1+w_2)+\eta^2\left(1-\frac{2\theta}{\pi}\right). \qquad (11)$$

This system has no common fixed point, which means that the w_i grow without bounds. The asymptotic behavior can be seen from the equation for cos(θ). Assuming that $w_1=w_2=w$, we find

$$\frac{d\cos(\theta)}{d\alpha}=\frac{\eta}{w}\frac{4}{\sqrt{2\pi}}[1-\cos(\theta)^2]+\frac{\eta^2}{w^2}\left(1-\cos(\theta)-\frac{2\theta}{\pi}\right). \qquad (12)$$

By taking $w\approx\sqrt{2/\pi}\,\eta\alpha$, the ODE leads to $1-\cos(\theta)\propto\alpha^{-4}$ for $\alpha\rightarrow\infty$. This means that $\theta\propto\alpha^{-2}$.

Simulations agree with the numerical integration of Eqs. (11), with the exception of very large α and correspondingly small θ (see Fig. 4). This is not surprising, since the α^{-2} decay is an effect of patterns that are classified differently. As long as the perceptrons give the same output on all patterns, w_1 and w_2 grow linearly, but the difference $\mathbf{w}_1-\mathbf{w}_2$ does not change, leading to $\theta\propto\alpha^{-1}$. This is observed in simulations for small angles, where no patterns happened to be classified differently on the considered timescale. Mathematically, this is related to a breakdown of the self-averaging properties of Eqs. (11) at the point $\theta=0$.

II. MUTUAL LEARNING, COMPETITION

In the previous section, all of the neural networks behave in the same way. Each perceptron tries to learn the output of its neighbor, and only the initial weight vectors are chosen randomly and differ from each other. Now we investigate a scenario where two networks behave differently. Network 1 is trying to simulate network 2 while 2 is trained on the opposite of the opinion of 1. This scenario describes a competition between two adaptive algorithms. If 2 is completely successful, the overlap is cos(θ)$=-1$, and perceptron 1 always fails in its prediction, and vice versa. A motivation from game theory can be drawn from the game of penny matching, where both players make a binary decision simultaneously. One player wins if the decisions are the same, the other if they are different.

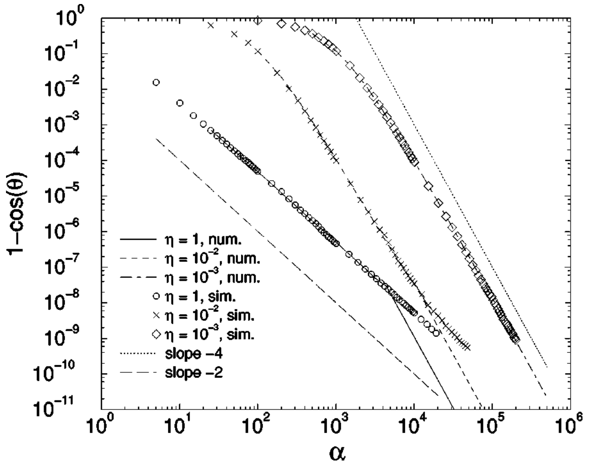


FIG. 4. Mutual learning with rule H: simulations with $N=100$ show good agreement with Eqs. (11), except for very small angles θ.

A. Rule *P*

If both perceptrons use rule P for their respective learning aim, the update rules are

$$\mathbf{w}_1^+=\mathbf{w}_1+(\eta_1/N)\mathbf{x}\sigma_2\Theta(-\sigma_1\sigma_2),$$

$$\mathbf{w}_2^+=\mathbf{w}_2-(\eta_2/N)\mathbf{x}\sigma_1\Theta(\sigma_1\sigma_2). \qquad (13)$$

The corresponding differential equations for the order parameters are

$$\frac{dw_1}{d\alpha}=-\frac{\eta_1}{\sqrt{2\pi}}[1-\cos(\theta)]+\frac{\eta_1^2}{2w_1}\frac{\theta}{\pi},$$

$$\frac{dw_2}{d\alpha}=-\frac{\eta_2}{\sqrt{2\pi}}[1+\cos(\theta)]+\frac{\eta_2^2}{2w_2}\left(1-\frac{\theta}{\pi}\right), \qquad (14)$$

$$\frac{dR}{d\alpha}=\frac{\eta_1 w_2}{\sqrt{2\pi}}[1-\cos(\theta)]-\frac{\eta_2 w_1}{\sqrt{2\pi}}[1+\cos(\theta)].$$

The common fixed point for these equations is $w_i=\sqrt{2\pi}\,\eta_i/4$, cos($\theta$)=0. This is hardly surprising, since none of the perceptrons has a better algorithm than the other. The learning rate only rescales the weight vectors; the ratio η_i/w_i, which determines how fast the direction of $\mathbf{w}_i$ in weight space can change, is independent of η at the fixed point.

B. Rule *H*

The picture is slightly different if both perceptrons learn from every pattern they see. The resulting differential equations are

$$\frac{dw_1}{d\alpha}=\sqrt{\frac{2}{\pi}}\eta_1\cos(\theta)+\frac{\eta_1^2}{2w_1},$$

$$\frac{dw_2}{d\alpha}=-\sqrt{\frac{2}{\pi}}\eta_2\cos(\theta)+\frac{\eta_2^2}{2w_2}, \tag{15}$$

$$\frac{dR}{d\alpha}=\sqrt{\frac{2}{\pi}}\eta_1 w_2-\sqrt{\frac{2}{\pi}}\eta_2 w_1-\eta_1\eta_2(\pi-2\theta).$$

The fixed point of R is reached if $\theta=\pi/2$ and $\eta_1/w_1=\eta_2/w_2$, i.e., the vectors are perpendicular and the scaled learning rates η_i/w_i are the same for both perceptrons. Under these conditions, the equations for w_i can be solved: $w_i=\eta_i[\alpha+(w_{i,0}/\eta_1)^2]^{1/2}$, so w_i shows the $\sqrt{\alpha}$ scaling typical for random walks. Geometrically, the Hebb rule adds corrections to the weight vector that are on average parallel to the teacher vector. Since the teacher is moving at the same angular velocity as the student, the movement of both vectors resembles a random walk. Again, η only sets the temporal and spatial scale.

C. Rule *P* versus rule *H*

The result of the competition becomes more interesting when both perceptrons use different algorithms. For example, we let perceptron 1 use rule *P*, while 2 uses *H*. The derivation of the differential equations is again straightforward,

$$\frac{dw_1}{d\alpha}=-\frac{\eta_1}{\sqrt{2\pi}}[1-\cos(\theta)]+\frac{\eta_1^2}{2w_1}\frac{\theta}{\pi},$$

$$\frac{dw_2}{d\alpha}=-\sqrt{\frac{2}{\pi}}\eta_2\cos(\theta)+\frac{\eta_2^2}{2w_2}, \tag{16}$$

$$\frac{dR}{d\alpha}=-\sqrt{\frac{2}{\pi}}\eta_2 w_1+\frac{\eta_1 w_2}{\sqrt{2\pi}}[1-\cos(\theta)]+\frac{\eta_1\eta_2\theta}{\pi}.$$

They have a common fixed point defined by

$$\theta\frac{\cos(\theta)^2}{[1-\cos(\theta)]^2}=\frac{\pi}{4},$$

$$w_1=\frac{\eta_1}{\sqrt{2\pi}}\frac{\theta}{1-\cos(\theta)}, \tag{17}$$

$$w_2=\frac{\sqrt{2\pi}}{4}\frac{\eta_2}{\cos(\theta)}.$$

These equations can be solved numerically and yield $\cos(\theta)\cong 0.459$, $w_1\cong 0.806\eta_1$, and $w_2\cong 1.37\eta_2$. Although perceptron 1 makes no use of many of the presented patterns, whereas perceptron 2 incorporates all of then, 1 wins the competition: the perceptron using rule *H* has a smaller η/w ratio and is thus less flexible.

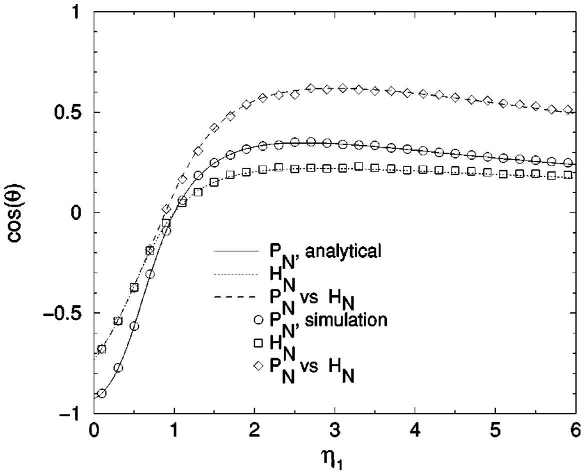


FIG. 5. Competing learning aims with normalized weights: η_2 is set to 1 while η_1 is varied. The analytical curves are fixed points of Eqs. (18), (19), and (20), respectively.

D. Normalized weights

By setting the weights to 1 after each learning step, a new length scale is introduced, leading to a more complex dependence of the solution on the learning rates. For brevity, we only give the differential equations for the different learning rules and explain some common features. If both networks use rule P_N, the ODE is

$$\frac{dR}{d\alpha}=\frac{1}{\sqrt{2\pi}}[\eta_1(1-R)-\eta_2(1+R)]$$
$$+\frac{R}{\sqrt{2\pi}}[\eta_1(1-R)+\eta_2(1+R)]$$
$$-\frac{R}{2\pi}[\eta_1^2\theta+\eta_2^2(\pi-\theta)], \tag{18}$$

for rule H_N we find

$$\frac{dR}{d\alpha}=\sqrt{\frac{2}{\pi}}(\eta_2-\eta_1)(R^2-1)-\frac{R}{2}(\eta_1^2+\eta_2^2)$$
$$-\eta_1\eta_2\left(1-\frac{2\theta}{\pi}\right), \tag{19}$$

and if rule P_N is used by perceptron 1 and H_N by 2, the equation is

$$\frac{dR}{d\alpha}=R\left(\sqrt{\frac{2}{\pi}}\eta_2 R-\frac{\eta_2^2}{2}+\frac{\eta_1}{\sqrt{2\pi}}(1-R)-\frac{\eta_1^2}{2}\frac{\theta}{\pi}\right)$$
$$+\frac{\eta_1}{\sqrt{2\pi}}(1-R)-\sqrt{\frac{2}{\pi}}\eta_2+\eta_1\eta_2\frac{\theta}{\pi}. \tag{20}$$

The behavior of the fixed point is similar in all the following cases (see Fig. 5):

(i) If, say, η_2 is fixed and $\eta_1 \rightarrow 0$, R goes to a value $R \neq -1$. This is expected, since both P_N and H_N only achieve finite values of R for fixed teachers.

(ii) If both perceptrons use the same algorithm with the same learning rate, the result is $R=0$, as expected.

(iii) If $\eta_i \rightarrow \infty$ for either i, $R \rightarrow 0$. Infinite learning rate means that in every time step the perceptron discards all the information it previously had, replacing it with the current $\pm \mathbf{x}$. Theoretically, that makes it predictable for the other network; in practice, both agents are confused. The notable exception is the case of P_N vs. H_N, where a nonvanishing R results if both $\eta_i \rightarrow \infty$ with a finite ratio η_1 / η_2.

III. CONFUSED TEACHER

For any prediction algorithm there is a bit sequence for which this algorithm fails completely, with 100% error [13]. In fact, such a sequence is easily constructed: Just take the opposite of the predicted bit at each time step. In Ref. [13] a perceptron was used for the prediction algorithm.

Here we do not consider bit sequences. However, it turns out that many statistical properties of the prediction algorithm are similar when random inputs are used instead of a window of the antipredictable bit sequence. Hence we consider the following scenario: Preceptron 1 is trained on the negative of its own output. Perceptron 2 is trained on the output of perceptron 1.

This is similar to the teacher/student model where the teacher weight vector performs a random walk [8]. But here the teacher is ‘‘confused’’; it does not believe its own opinion and learns the opposite of it.

The update rule of perceptron 1 now only depends on its own output,

$$\mathbf{w}_1^+ = \mathbf{w}_1 - (\eta/N)\mathbf{x}\sigma_1 . \tag{21}$$

Geometrically speaking, the vector performs a directed random walk in which every learning step has a negative overlap with the current vector. An equilibrium length is reached when a typical learning step leads back onto the surface of an N-dimensional hypersphere. This fixed point of w_1 is easily calculated to be

$$w_1 = \sqrt{2\pi}\,\eta/4 \cong 0.6267\eta, \tag{22}$$

and the weight vector typically moves on the surface of a hypersphere of that radius.

A. Rule *H*

What happens if a second perceptron tries to follow the output of the confused teacher? Again, the results depend entirely on the used algorithm. The simplest case, the Hebb rule, also has a geometrical interpretation that is revealed by a look at the update rule,

$$\begin{aligned} \mathbf{w}_1^+ &= \mathbf{w}_1 - (\eta/N)\mathbf{x}\sigma_1 , \\ \mathbf{w}_2^+ &= \mathbf{w}_2 + (\eta/N)\mathbf{x}\sigma_1 . \end{aligned} \tag{23}$$

As in Sec. I A, the sum of both vectors is constant, so there is a class of solutions to the ODEs

$$\begin{aligned} \frac{dw_1}{d\alpha} &= -\sqrt{\frac{2}{\pi}}\eta + \frac{\eta^2}{2w_1}, \\ \frac{dw_2}{d\alpha} &= \sqrt{\frac{2}{\pi}}\eta\cos(\theta) + \frac{\eta^2}{2w_2}, \\ \frac{dR}{d\alpha} &= \sqrt{\frac{2}{\pi}}\eta[w_1 - w_2\cos(\theta)] + \eta^2 \end{aligned} \tag{24}$$

defined by $w_{1,f} = \sqrt{2\pi}\eta/4$ and $w_{2,f} = -\sqrt{2\pi}\eta/[4\cos(\theta)]$. The solution is given by the initial condition, i.e., the initial sum $|\mathbf{w}_1 + \mathbf{w}_2|$. The fixed point angle can be calculated by applying the cosine theorem to a triangle with side lengths $w_{1,f}$, $w_{2,f}$, and $|\mathbf{w}_1 + \mathbf{w}_2|$; starting from perpendicular vectors of norm w_0, one finds

$$\cos(\theta) = -\left[1 + \frac{16}{\pi}\left(\frac{w_0}{\eta}\right)^2\right]^{-1/2}. \tag{25}$$

Geometrically, for large learning rate η both norms become much larger than w_0; the only way to achieve this while keeping the sum constant is a large angle. For small η, w_1 becomes very small compared to the sum, and thus to w_2. So the direction of $\mathbf{w}_2$ stays nearly unchanged while $\mathbf{w}_1$ performs its random walk, leading to nearly perpendicular vectors on average.

B. Rule *P*

If perceptron 2 uses rule *P*, the sum of the vectors is not conserved, and a simple geometrical interpretation is not possible. However, the equations of motion can still be solved,

$$\begin{aligned} \frac{dw_1}{d\alpha} &= -\sqrt{\frac{2}{\pi}}\eta + \frac{\eta^2}{2w_1}, \\ \frac{dw_2}{d\alpha} &= -\frac{\eta}{\sqrt{2\pi}}[1 - \cos(\theta)] + \frac{\eta^2}{2w_2}\frac{\theta}{\pi}, \\ \frac{dR}{d\alpha} &= \sqrt{\frac{2}{\pi}}\eta\cos(\theta) - \frac{w_1\eta}{\sqrt{2\pi}}(1-c) - \eta^2\frac{\theta}{\pi}. \end{aligned} \tag{26}$$

The fixed point of cos(θ) is given by the solution of $4\theta/\pi = [1+\cos(\theta)]^2$, independent from η. The numerical solution is $\theta \cong 0.777\pi$, $\cos(\theta) = -0.761$, $w_2 = 0.552\eta$ (in accordance with Ref. [13], where a special case of this problem was solved). Remarkably, the generalization error is larger than 50%—even the ‘‘smarter’’ perceptron learning rule predicts the behavior of the confused teacher with less success than random guessing would.

C. Optimal learning rule

This raises an interesting question: is there any ‘‘reasonable’’ algorithm for perceptrons that allows them to track the confused teacher, i.e., an algorithm that is not aware of the teacher's self-avoiding motion but tries to find it assuming it is stationary? If there are such algorithms that achieve a positive overlap, one of them has to be the rule that optimizes

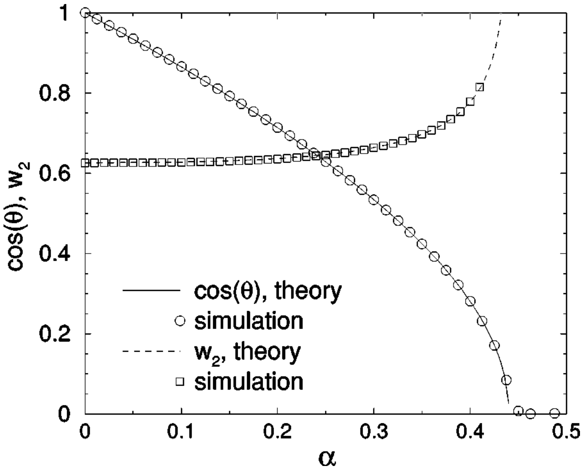


FIG. 6. Confused teacher: Even with the optimal weight function (27) the student only achieves an overlap of $\cos(\theta)=0$. Starting values are $w_1=w_2=\sqrt{2\pi}/4$, $\cos(\theta)=1$, $\eta=1$. Simulations are performed with $N=2000$; the statistical error is smaller than the size of the symbols.

student-teacher overlap in each time step—the optimal weight function derived by Kinouchi and Caticha [14]:

$$f_{\rm opt}=\frac{w_2\tan(\theta)}{\sqrt{2\pi}}\exp\left[-\frac{(\mathbf{x}\cdot\mathbf{w}_2)^2}{2\tan(\theta)^2\;\;w_2^2}\right]$$

$$\times\frac{1}{\Phi\{\sigma_1\mathbf{x}\cdot\mathbf{w}_2/[w_2\tan(\theta)]\}},\tag{27}$$

where $\Phi(x)=\int_{-\infty}^{x}\exp(-z^2/2)/\sqrt{2\pi}\,dz$. If w_1 is set to its fixed point for simplicity's sake, the calculation yields the following ODEs for $\cos(\theta)$ and w_2:

$$\frac{d\cos(\theta)}{d\alpha}=\frac{1}{4\pi}\frac{\sin(\theta)^2}{\cos(\theta)}I-\frac{2}{\sqrt{2\pi}w_1\cos(\theta)},\tag{28}$$

$$\frac{dw_2}{d\alpha}=\frac{w_2}{4\pi}\tan(\theta)^2\,I,\tag{29}$$

where

$$I=\int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}}$$

$$\times\exp\left(-\frac{1+\cos(\theta)^2}{2\sin(\theta)^2}x^2\right)\frac{1}{\Phi[-x\cot(\theta)]\Phi[x\cot(\theta)]}dx.\tag{30}$$

Calculating whether $\cos(\theta)=0$ is in fact a fixed point of the confused teacher and/or optimal student scenario is problematic, since the optimal weight function (27) diverges at $\theta=\pi/2$. However, the numerical solution of Eqs. (28) and (29) shows clearly that even starting from $\cos(\theta)=1$, the system evolves towards $\theta=\pi/2$, which indeed seems to be the upper limit for success. Simulations of the learning process again agree well with our theory (see Fig. 6).

The previous examples have shown that learning rules that work well with a stationary teacher give negative or zero overlap with the confused teacher. However, a shrewd student could use a learning rule that learns the opposite of the teacher's output to achieve a positive overlap, and thus a prediction accuracy of more than 50%.

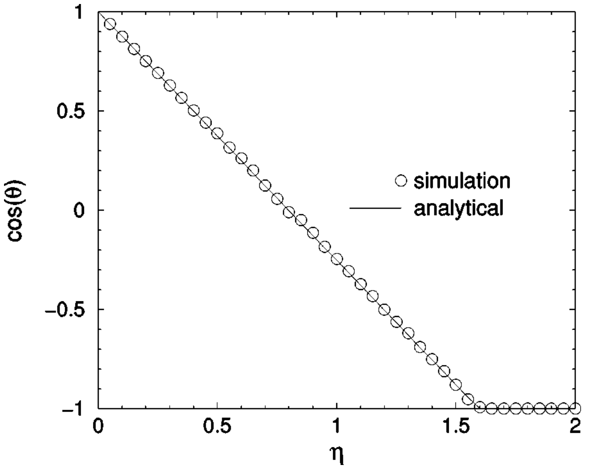


FIG. 7. Confused teacher: If the teacher is slowed down by normalizing its weight, it can be tracked by a student using, e.g., rule H_N. The figure shows the fixed point of Eq. (31) and simulations with $N=100$.

D. Rule H_N

There is another way of achieving a positive overlap with the confused teacher with simple learning rules: if the teacher perceptron is "slowed down" by keeping its weights normalized and setting η to some small value, a student using P_N or H_N can track the teacher nearly perfectly for very small learning rates. For simplicity's sake, let us consider H_N with identical learning rates. The differential equation for R is

$$\frac{dR}{d\alpha}=(R+1)\left(\sqrt{\frac{2}{\pi}}(1-R)\eta-\eta^2\right).\tag{31}$$

The fixed points are $R=-1$ or $R=-\sqrt{2/\pi}\eta+1$. This result is again confirmed by simulations as seen in Fig. 7. The fixed point goes to 1 as $\eta\rightarrow 0$.

IV. PERCEPTRONS IN THE MINORITY PROBLEM

The concept of interacting neural networks can be applied to a problem that has received much attention recently: the El Farol Bar problem [9]. The problem was originally inspired by a popular bar that has a limited capacity: if too many people attend, it becomes crowded, and patrons do not enjoy the evening. In a more special formulation, each agent out of a population of K decides in each time step (each Saturday evening) to take one of two alternatives (go to the bar or stay at home). Those agents who are in the minority win, the others lose. Decisions are made independently; the only information available to agents is the decision of the minority in the last N time steps.

Many papers (see, e.g., [11]) investigated a specific realization of the model called the Minority Game. In this model each agent has a small number of randomly chosen decision tables (Boolean functions) that prescribe an action based on the previous history, and which of the tables is used is decided according to how successful each one was in the course of the game. It turned out that the success of the game depends on the ratio between the number of players and the size of the history window, and general conclusions on the behavior of crowded markets were drawn [15,16].

We will discuss a different approach that yields different behavior: Each agent i is represented by a perceptron $\mathbf{w}_i$ that uses the time series $\mathbf{S}_t=(S_t, S_{t-1}, \ldots, S_{t-N+1})$ of past minority decisions to make a prediction on the next time step. It then learns the output of the minority according to some learning rule.

In our approach all of the agents are flexible in their decisions. Each agent uses an identical adaptive algorithm which is trained by the history of the game, the only information available to each of the agents. However, each agent uses a different randomly chosen initial state of its network. If all weight vectors of the networks would collapse, all agents would make the same decision, and all would lose. If all weights remained in the random initial state, each agent would make a random guess which yields a reasonable performance of the system. Our calculation shows that training can improve the performance of the system compared to the random state.

Following Ref. [17], we replace the history $\mathbf{S}_t$ by a random vector $\mathbf{x}$. Simulations show that this changes the results only quantitatively, if at all.

This strategy fulfills the restrictions that the original problem posed: the agents do not communicate except through majority decisions, and individual decisions are based on experience (induction or learning) rather than perfect knowledge of the system (deduction). However, since each player uses only one strategy whose parameters can be fine-tuned to the current environment rather than a set of completely different strategies, no quenched bias in the players' behavior is to be expected.

A. General notes on performance

The commonly used measure of collaboration in the minority problem is the average standard deviation of the sum of outputs of all agents:

$$\frac{\sigma^2}{K}=\frac{1}{K}\left\langle\left(\sum_{i=1}^{K}\sigma_i\right)^2\right\rangle. \tag{32}$$

If each agent makes random decisions, one gets $\sigma^2/K=1$. The probability of two perceptrons i and j giving the same output on a random pattern is $1-\theta_{ij}/\pi$. Any ensemble of vectors $\mathbf{w}_i$ can be thought of as centered around a center of mass $\mathbf{C}=\Sigma_{i=1}^{K}\mathbf{w}_i/K$ with a norm C (for random vectors of length 1, C would be of order $1/\sqrt{K}$). The weights can then be written as $\mathbf{w}_i=\mathbf{g}_i+\mathbf{C}$, with $\Sigma_{i=1}^{K}\mathbf{g}_i=\mathbf{0}$. For the sake of simplicity, we will assume a symmetrical configuration with $g_i=1$ and $\mathbf{g}_i\cdot\mathbf{g}_j=-1/(K-1)$ for $i\neq j$. [An ensemble of randomly chosen vectors of norm 1 would give $g_i^2=1-1/K \pm O(1/\sqrt{N})$ and $\mathbf{g}_i\cdot\mathbf{g}_j=-1/K\pm O(1/\sqrt{N})$.]

The average overlap between different weights is now $R=C^2-1/(K-1)$, their average norm $w_i=\sqrt{\mathbf{C}^2+1}$. With this, Eq. (32) can be evaluated

$$\begin{aligned}\frac{\sigma^2}{K}&=\frac{1}{K}\left\langle\sum_{i=1}^{K}1+\sum_{i=1}^{K}\sum_{j\neq i}^{K}\operatorname{sgn}(\mathbf{x}\cdot\mathbf{w}_i)\operatorname{sgn}(\mathbf{x}\cdot\mathbf{w}_j)\right\rangle_{\mathbf{x}}\\&=1+(K-1)\left[1-\frac{2}{\pi}\arccos\left(\frac{C^2-1/(K-1)}{C^2+1}\right)\right].\end{aligned} \tag{33}$$

If C is set to 0 and K is large, a linear expansion of the arccos term in Eq. (33) gives $\sigma^2_{opt}/K\approx 1-2/\pi\cong 0.363$. The small anticorrelations (of order $1/K$) between the vectors suffice to change the prefactor in the standard deviation.

If C is much larger than g, there is a strong correlation between the perceptrons. Most perceptrons will agree with the classification by the center-of-mass $\operatorname{sgn}(\mathbf{x}\cdot\mathbf{C})$. As $C\to\infty$, σ^2/K saturates at K.

B. Hebbian learning

Now each perceptron is trying to learn the decision of the minority according to rule H. S denotes the majority decision:

$$\mathbf{w}_i^+=\mathbf{w}_i-\frac{\eta}{M}\mathbf{x}\operatorname{sgn}\left(\sum_{j=1}^{N}\operatorname{sgn}(\mathbf{x}\cdot\mathbf{w}_j)\right)=\mathbf{w}_i-\frac{\eta}{M}\mathbf{x}S. \tag{34}$$

As the same correction is added to each weight vector, their mutual distances remain unchanged. Only the center of mass is shifted. We now treat C as an order parameter:

$$\mathbf{C}^+=\sum_{i=1}^{K}\frac{\mathbf{w}_i}{N}-\frac{\eta}{M}\mathbf{x}s, \tag{35}$$

$$C^{2+}=C^2-\frac{2\eta}{N}\mathbf{x}\cdot\mathbf{C}S+\frac{\eta^2}{N}. \tag{36}$$

To average over $\mathbf{x}\cdot\mathbf{C}S$ in the thermodynamic limit, we introduce a field $h=\mathbf{x}\cdot\mathbf{C}$ and average over $\mathbf{x}$ for fixed h:

$$\mathbf{x}\cdot\mathbf{C}S=|h|\operatorname{sgn}\left(\sum_{i=1}^{K}\operatorname{sgn}(h)\operatorname{sgn}(\mathbf{x}\cdot\mathbf{g}_i+h)\right). \tag{37}$$

The quantity $\operatorname{sgn}(h)\operatorname{sgn}(\mathbf{x}\cdot\mathbf{g}_i+h)$ is a random variable with mean $\operatorname{erf}(|h|/\sqrt{2})$ and variance $1-\operatorname{erf}(|h|/\sqrt{2})^2$. In a linear approximation for small $|h|$, we replace this by mean $\sqrt{2/\pi}|h|$ and variance 1.

For sufficiently large K, one can use the central limit theorem to show that $\Sigma_{i=1}^{K}\operatorname{sgn}(h)\operatorname{sgn}(\mathbf{x}\cdot\mathbf{g}_i+h)$ becomes a Gaussian random variable with mean $\sqrt{2/\pi}K|h|$. Since the terms of the sum in Eq. (37) are anticorrelated rather than independent, the variance turns out to be $(1-2/\pi)K$ rather than K, analogously to Eq. (33). This yields

$$\left\langle\operatorname{sgn}\left(\sum_{i=1}^{K}\operatorname{sgn}(h)\operatorname{sgn}(\mathbf{x}\cdot\mathbf{g}_i+h)\right)\right\rangle=\operatorname{erf}[\sqrt{K/(\pi-2)}|h|]. \tag{38}$$

Since h is a Gaussian variable with mean 0 and variance C^2, the average over hS can now be evaluated. We find the following differential equation for the norm of the center of mass:

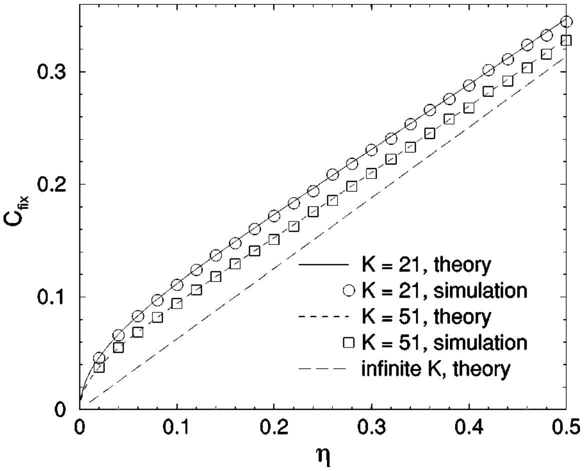


FIG. 8. Fixed point of C vs η: simulations with $N=100$ agree well with Eq. (40). The limit for $K\rightarrow\infty$ is $C=\sqrt{2\pi}\eta/4$.

$$\frac{dC^2}{d\alpha}=-\frac{4\eta}{\sqrt{2\pi}}\sqrt{\frac{2K/(\pi-2)}{1+2K(\pi-2)C^2}}C^2+\eta^2. \qquad (39)$$

The fixed point of C, which can be plugged into Eq. (33) to get $\sigma^2/K(\eta,K)$, is

$$C=\frac{\sqrt{\pi}}{4}\eta\sqrt{1+\sqrt{1+\frac{16(\pi-2)}{\pi K\eta^2}}} \qquad (40)$$

(see Figs. 8 and 9). If C is large, the majority of perceptrons will usually make the same decision as $\mathbf{C}$, which then behaves like the single confused perceptron: $C\rightarrow\sqrt{2\pi}\eta/4$ if $K\eta^2\rightarrow\infty$, compare to Eq. (22).

For small C the majority may not coincide with $\mathrm{sgn}(\mathbf{x}\cdot\mathbf{C})$. In that case, the learning step has a positive overlap with $\mathbf{C}$, leading to $C\propto\sqrt{\eta}$ as $\eta\rightarrow 0$.

The derivation given is only correct if $N\rightarrow\infty$ and K is large. However, simulations show very good agreement even for $K=21$ and $N=100$ (see Fig. 9). For a smaller number of dimensions N, there is even a tendency towards smaller σ^2/K. This can be understood in the extreme case of $N=1$: Each perceptron is characterized by one number; the outcome is decided by whether the majority of numbers is smaller than 0 or larger, regardless of the ''pattern.'' The learning step consists of shifting all numbers up or down by the same amount. In the case of small η, the fixed point is characterized by $(N-1)/2$ players firmly on one side of the origin, $(N-1)/2$ on the other side, and one unfortunate loser who changes sides at every step.

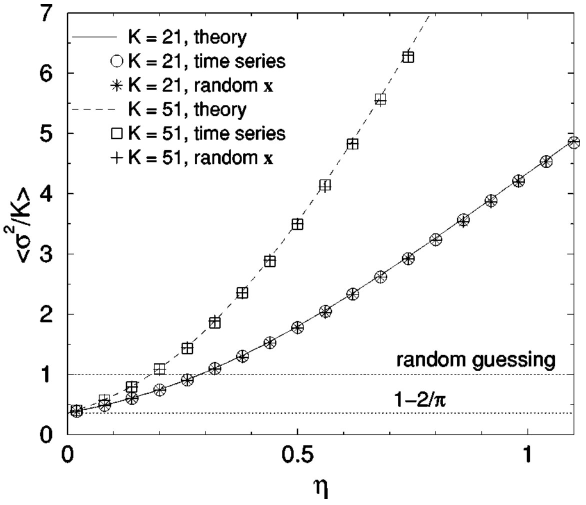


FIG. 9. Fixed point of σ^2/K vs η: the combination of Eqs. (33) and (40) shows that sufficiently small learning rates lead to $\sigma^2/K<1$.

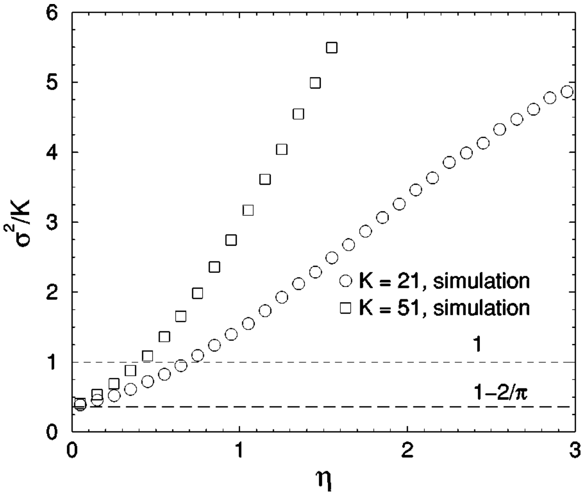


FIG. 10. Using a modified P_N algorithm improves the results, compared to Fig. 9. Simulations again use $N=100$.

Interestingly, if the time series generated by the minority decisions is used as patterns, the functions $\sigma^2(C)$ and $C(\eta)$ are quantitatively different from those found for random patterns. However, in the final result $\sigma^2(\eta)$ no disagreement can be noticed (see Fig. 9).

The presented Hebb algorithm may appear too simplistic and the chosen initial conditions too artificial. It must therefore be emphasized that there are other learning algorithms that lead to the same anticorrelated state. In particular, a variation of rule P_N has proven successful in simulations (see Fig. 10): all perceptrons that are on the minority side take a learning step, and weights are kept normalized. The regular rule P where perceptrons on the majority side move, however, leads to strong clustering and $\sigma^2/K\propto K$.

The absence of scaling behavior if $N>K$ and the fact that smaller dimensions (corresponding to smaller memory of the time series) even improve the results show that the conclusions drawn from the ''conventional'' Minority Game do not apply to all conceivable strategies for the Bar problem. We think that the dependence of σ^2/K on the ratio between available strategies and players is caused by the use of quenched strategies and will not arise in any scenario in which agents stick to one strategy which is fine-tuned by some learning process. The case of $N=1$ implies that there are strategies that give $\sigma^2/K\propto 1/K$. We will elaborate this point in another publication.

V. SUMMARY

We have investigated several scenarios of mutually interacting neural networks. Using perceptrons with well-known on-line training algorithms in the limit of infinite system size, we derived exact equations of motion for the dynamics of order parameters that describe the properties of the system. In the first scenario a system of K perceptrons is placed on a ring. All perceptrons receive the same input and each perceptron is trained by the output of its neighbor on the ring. We have used two well-known training algorithms: the perceptron rule which concentrates on examples where the networks disagree, and the Hebbian rule where each example changes the weights. We find that with unnormalized weights the system relaxes to a stationary state of high symmetry: each perceptron has the same overlap with all others. The overlap depends on the learning rate: with increasing η the perceptrons increase their mutual angle as much as possible.

For the perceptron learning rule with normalized weights we find phase transitions with increasing learning rate η. For large values of η, the symmetry is broken, but the symmetry of the ring is still conserved. For the Hebbian rule we find a different behavior. The lengths of the weights diverge, the mutual angles shrink to zero, and the perceptrons eventually come to perfect agreement in the limit of infinitely many training examples.

We furthermore study the behavior of perceptrons that pursue competing learning aims for different learning algorithms. If two perceptrons follow mutually exclusive learning aims using the same algorithm, a draw results. If they use different rules, the outcome depends on factors like the rescaled learning rate η/w. We find that a perceptron that learns the opposite of its own prediction cannot be tracked by a student perceptron that learns the positive output of the confused teacher: all rules achieve a negative overlap.

Finally an ensemble of interacting perceptrons is used to solve a model of a closed market. Each agent uses a perceptron which is trained on the decision of the minority. Our analytic solution shows that the system relaxes to a stationary state which yields a good performance of the system for small learning rates η. In contrast to the Minority Game of Refs. [11] our approach leads to identical profits for all agents in the long run. In addition, the performance of the algorithm is insensitive to the size of the history window used for the decision.

This paper is a first step towards more complex models of interacting neural networks. We have presented analytically accessible cases which may open the road to a general understanding of interacting adaptive systems with possible applications in biology, computer science, and economics.

ACKNOWLEDGMENTS

All authors are grateful for support by the GIF. This paper also benefitted from a seminar at the Max-Planck-Institut komplexer Systeme, Dresden. We thank Johannes Berg, Michael Biehl, Liat Ein-Dor, Andreas Engel, Georg Reents, and Robert Urbanczik for helpful discussions.

APPENDIX

The following averages are used in our calculations to derive deterministic differential equations from the update rules. The angled brackets denote averages over isotropically distributed pattern vectors. In the limit $N\rightarrow\infty$, $\mathbf{w}_1\cdot\mathbf{x}$ and $\mathbf{w}_2\cdot\mathbf{x}$ are correlated Gaussian random variables, and the averages can be calculated by integrating over their joint probability distribution with appropriate boundaries. In many cases, simple geometrical calculations give the same result with less effort:

$$\langle \mathbf{x}\cdot\mathbf{w}_1\, \sigma_2 \Theta(-\sigma_1\sigma_2)\rangle = -\frac{w_1}{\sqrt{2\pi}}[1-\cos(\theta)], \tag{A1}$$

$$\langle \mathbf{x}\cdot\mathbf{x}\,\Theta(-\sigma_1\sigma_2)\rangle = N\frac{\theta}{\pi}, \tag{A2}$$

$$\langle \mathbf{x}\cdot\mathbf{w}_1\, \sigma_1 \Theta(\sigma_1\sigma_2)\rangle = \frac{w_1}{\sqrt{2\pi}}[1+\cos(\theta)], \tag{A3}$$

$$\langle \mathbf{x}\cdot\mathbf{x}\,\Theta(\sigma_1\sigma_2)\rangle = N\left(1-\frac{\theta}{\pi}\right), \tag{A4}$$

$$\langle \mathbf{x}\cdot\mathbf{w}_1\, \sigma_1\rangle = \sqrt{\frac{2}{\pi}}w_1, \tag{A5}$$

$$\langle \mathbf{x}\cdot\mathbf{w}_1\, \sigma_2\rangle = \sqrt{\frac{2}{\pi}}w_1\cos(\theta), \tag{A6}$$

$$\langle f_{\rm opt}\rangle = \frac{2w_2}{\sqrt{2\pi}}\,\frac{\sin(\theta)^2}{\cos(\theta)}, \tag{A7}$$

$$\langle f_{\rm opt}\,\mathbf{x}\cdot\mathbf{w}_2\,\sigma_1\rangle = 0, \tag{A8}$$

$$I = \int_{-\infty}^{\infty}\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1+\cos(\theta)^2}{2\sin(\theta)^2}x^2\right) \times\{\Phi[-x\cot(\theta)]\Phi[x\cot(\theta)]\}^{-1}dx, \tag{A9}$$

$$\langle f_{\rm opt}^2\rangle = \frac{w_2^2}{2\pi}\tan(\theta)^2 I, \tag{A10}$$

$$\langle f_{\rm opt}\,\mathbf{x}\cdot\mathbf{w}_1\,\sigma_1\rangle = \frac{w_1w_2}{2\pi}\,\frac{\sin(\theta)^2}{\cos(\theta)}I. \tag{A11}$$

[1] J. Hertz, A. Krogh, and R. Palmer, *Introduction to the Theory of Neural Computation* (Addison-Wesley, Redwood City, CA, 1991).

[2] M. Opper and W. Kinzel, in *Models of Neural Networks III*, edited by E. Domany, J. van Hemmen, and K. Schulten (Springer-Verlag, Heidelberg, 1995), Chapter on Statistical Mechanics of Generalization, pp. 151–209.

[3] *On-line Learning in Neural Networks*, edited by D. Saad (Cambridge University Press, Cambridge, England, 1998).

[4] M. Biehl and P. Riegler, Europhys. Lett. **28**, 525 (1994).

[5] E. Eisenstein, I. Kanter, D. Kessler, and W. Kinzel, Phys. Rev. Lett. **74**, 6 (1995).

[6] M. Schröder and W. Kinzel, J. Phys. A **31**, 9131 (1998).

[7] D. H. Wolpert and K. Turner (unpublished).

[8] M. Biehl and H. Schwarze, J. Phys. A **26**, 2651 (1993).

[9] W. B. Arthur, Am. Econ. Assoc. Papers Proc. **84**, 406 (1994).

[10] D. Challet and Y.-C. Zhang, Physica A **246**, 407 (1997).

[11] M. Marsili, D. Challet, and R. Zecchina, e-print cond-mat/9908480; D. Challet, M. Marsili, and R. Zecchina, Phys. Rev. Lett. **84**, 1824 (2000); D. Challet and Y.-C. Zhang, Physica A **256**, 514 (1998); R. Savit, R. Manuca, and R. Riolo, Phys. Rev. Lett. **82**, 2203 (1999); D. Challet and M. Marsili, Phys. Rev. E **60**, R6271 (1999).

[12] G. Reents and R. Urbanczik, Phys. Rev. Lett. **80**, 5445 (1998).

[13] H. Zhu and W. Kinzel, Neural Comput. **10**, 2219 (1998).

[14] O. Kinouchi and N. Caticha, J. Phys. A **25**, 6243 (1992).

[15] D. Challet, M. Marsili, and Y.-C. Zhang, e-print cond-mat/9909265.

[16] N. F. Johnson, M. Hart, P. M. Hui, and D. Zheng, e-print cond-mat/9910072.

[17] A. Cavagna, Phys. Rev. E **59**, R3783 (1999).