Systems/Circuits

Neural Classifiers with Limited Connectivity and Recurrent Readouts

Lyudmila Kushnir[1,2] **and Stefano Fusi**[2,3,4]

[1]LNC2, Departement d'Etudes Cognitives, Ecole Normale Superieure, Institut National de la Santé et de la Recherche Médicale, PSL Research University, 75005 Paris, France, [2]Center for Theoretical Neuroscience, College of Physicians and Surgeons, [3]Mortimer B. Zuckerman Mind Brain Behavior Institute, and [4]Kavli Institute for Brain Sciences, Columbia University, New York, New York 10027

For many neural network models in which neurons are trained to classify inputs like perceptrons, the number of inputs that can be classified is limited by the connectivity of each neuron, even when the total number of neurons is very large. This poses the problem of how the biological brain can take advantage of its huge number of neurons given that the connectivity is sparse. One solution is to combine multiple perceptrons together, as in committee machines. The number of classifiable random patterns would then grow linearly with the number of perceptrons, even when each perceptron has limited connectivity. However, the problem is moved to the downstream readout neurons, which would need a number of connections as large as the number of perceptrons. Here we propose a different approach in which the readout is implemented by connecting multiple perceptrons in a recurrent attractor neural network. We prove analytically that the number of classifiable random patterns can grow unboundedly with the number of perceptrons, even when the connectivity of each perceptron remains finite. Most importantly, both the recurrent connectivity and the connectivity of downstream readouts also remain finite. Our study shows that feedforward neural classifiers with numerous long-range afferent connections can be replaced by recurrent networks with sparse long-range connectivity without sacrificing the classification performance. Our strategy could be used to design more general scalable network architectures with limited connectivity, which resemble more closely the brain neural circuits that are dominated by recurrent connectivity.

Key words: attractor networks; classifier; committee machines; perceptron; sparse connectivity

Significance Statement

The mammalian brain has a huge number of neurons, but the connectivity is rather sparse. This observation seems to contrast with the theoretical studies showing that for many neural network models the performance scales with the number of connections per neuron and not with the total number of neurons. To solve this dilemma, we propose a model in which a recurrent network reads out multiple neural classifiers. Its performance scales with the total number of neurons even when each neuron of the network has limited connectivity. Our study reveals an important role of recurrent connections in neural systems like the hippocampus, in which the computational limitations due to sparse long-range feedforward connectivity might be compensated by local recurrent connections.

Introduction

The performance of a neural circuit is often evaluated by determining the number of input–output functions that can be implemented or, equivalently, by the number of inputs that can be classified correctly by the neural circuit. Theoretical studies on perceptrons (Rosenblatt, 1957) and recurrent neural circuits (Amit, 1992) have shown that typically the performance of a neural circuit scales with the number of synaptic connections that individual neurons receive, and not with the total number of synapses or with the total number of neurons (Roudi and Latham, 2007). This is clearly a problem in the biological brain in which the connectivity is sparse, especially when long-range connections are considered (Bullmore and Sporns, 2012). One striking example is the mammalian hippocampus (Drew et al., 2013). A typical pyramidal neuron in rodent CA3 receives only 50 synapses from the upstream area (Amaral et al., 1990), the dentate gyrus (DG), which contains around 10^6 neurons. Not only is the connectivity sparse, but also the neural activity (Jung and Mc-

Received Dec. 11, 2017; revised Aug. 16, 2018; accepted Sept. 10, 2018.

Author contributions: L.K. and S.F. designed research; L.K. performed research; L.K. analyzed data; L.K. and S.F. wrote the paper.

S.F. is supported by the Gatsby Charitable Foundation, the Simons Foundation, the Schwartz Foundation, the Kavli Foundation, and the National Science Foundation NeuroNex Program Award DBI-1707398. L.K. was supported by Grants ANR-10-LABX-0087, ANR-10-IDEX-0001-02 and ERC grant PrediSpike, ERC - 312227.

Correspondence should be addressed to Stefano Fusi, Center for Theoretical Neuroscience, College of Physicians and Surgeons, Jerome L. Greene Science Center, 3227 Broadway, 5th and 6th Floors, Quad D, Columbia University, New York, NY 10027. E-mail: sf2237@columbia.edu.

https://doi.org/10.1523/JNEUROSCI.3506-17.2018

 0270-6474/18/389900-25$15.00/0

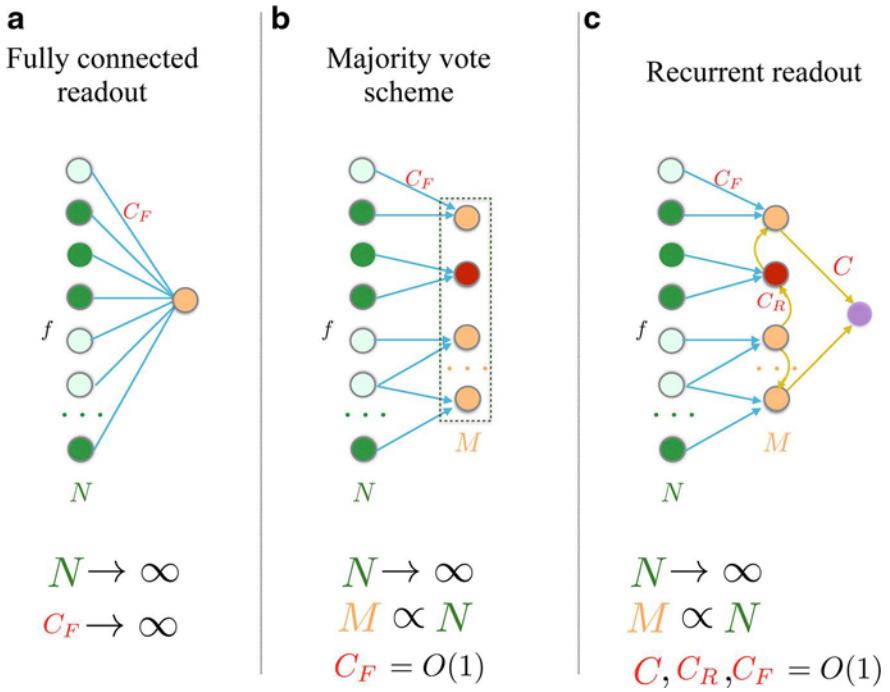

Figure 1. Architectures of the three network classifiers considered in the study and their scaling properties. ***a***, Fully connected readout, considered in the subsection Fully connected readout. The capacity of this classifier grows linearly with the number of input units *N*; however, the number of afferent connections C_F grows as quickly as *N*. ***b***, Committee machine of partially connected perceptrons (section Committee machine). The collective decision is made using a majority vote. Even though the number of connections per perceptron can be kept constant as the number of input neurons *N* increases, the number of readouts *M* has to grow with *N* to match the performance scaling of ***a***. The majority vote strategy requires another downstream readout, whose connectivity grows with *M* and hence with *N*. ***c***, The recurrent readout that we propose in section Committee machine with recurrent connections. As $N \to \infty$, the number of feedforward connections per perceptron C_F, the number of recurrent connections per perceptron C_R, as well as the number of connections of the downstream readout stay constant when *N* increases.

Naughton, 1993; Chawla et al., 2005), which seems paradoxical because a very limited connectivity could be compensated by denser neural activity.

One possible way to overcome the limitations of sparse connectivity is to adopt the strategy of "committee machines" (Nilsson, 1965), which are basically populations of classifiers. Each classifier is weak, as a perceptron with limited connectivity, but the output is generated by reading out a large number of weak classifiers and by combining them using a majority vote or some more sophisticated strategies. The final classification performance is significantly better than the one of each individual classifier, provided that the errors of the individual classifiers are sufficiently independent. The term committee machines goes back to the 1960s (Nilsson, 1965), but they have also been a focus of more recent studies (Parmanto et al., 1996; Bishop, 2007); basically, they are all based on strategies that in machine learning are known as ensemble methods or hypothesis boosting (Kearns M, unpublished observations; Zhou, 2012), strategies that are often adopted also in statistics (Rao and Subrahmaniam, 1971; Efron and Morris, 1973; Rubin and Weisberg, 1975; Green and Strawderman, 1991). Some of the examples include stacking (Wolpert, 1992; Breiman, 1996b), bagging (Breiman, 1996a), arcing (L. Breiman, unpublished observations), and AdaBoost (Adaptive Boosting; Freund et al., 1996; Freund and Schapire, 1997).

One class of committee machines is implemented using populations of neurons, each essentially behaving as a neural classifier, like a perceptron (Mitchison and Durbin, 1989; Monasson and Zecchina, 1995; Kwon and Oh, 1997; Urbanczik, 1997). Classifiers with limited connectivity are weak classifiers. It is possible to compute the classification capacity when each neural classifier has sparse connectivity (Kwon and Oh, 1997). The connections between the N input neurons and the $M < N$ neural classifiers are assumed to be nonoverlapping (N/M connections per "perceptron") and plastic. The final response of the committee machine is obtained by majority vote of the M neural classifiers, which can be easily implemented by introducing a readout neuron that is connected to all the neural classifiers with equal weights. The maximum number of correctly classified inputs is proportional to $N\sqrt{\log M}$, whereas each neural classifier would not go beyond N/M inputs. This is a favorable scaling, and it is similar to the one obtained in other committee machines. However, one has to keep in mind that in these implementations the neural classifiers have sparse connectivity, but the readout neuron performing the majority vote should have a number of connections that scale with *N*.

Here we propose a network architecture that overcomes the restrictions imposed by the limited connectivity, as in the committee machines, but replaces the readout neuron that has extensive connectivity with a more biologically plausible recurrent network in which all of the neurons have a number of connections that remains finite when the number of classifiable patterns grows unboundedly. More specifically, we show that the number of random inputs that can be correctly classified scales linearly with the number of input neurons N, even when the number of connections per neural classifier C_F does not increase with N. The number of neural classifiers M is assumed to be proportional to N.

Interestingly, under certain conditions the recurrent scheme has larger classification capacity than the majority vote scheme. This happens for sparse input representations, the regime that is relevant for the mammalian hippocampus and that we investigate in detail.

Materials and Methods

The following sections describe the models in detail and cover all the analytical calculations. They can be skipped if one is not interested in the technical details because in the Results we reintroduce all the important concepts, although in a less detailed format.

Fully connected readout

In this section, we derive the classification capacity of a single fully connected linear threshold readout, or perceptron (Fig. 1*a*), achieved with a simple learning rule that we use throughout this work. We assume that the input patterns and labels are random and uncorrelated, meaning that the activity of each input unit as well as the label for each pattern is chosen independently, which makes calculations analytically tractable. We use a simple Hebbian-like learning rule, which is not optimal and thus leads to a lower capacity than the $2N$ result in the study by Cover (1965). However, the scaling of the maximal number of learned input patterns $P_{\max}$ with the number of input units N is still linear, as is shown below.

Input statistics

We assume that pairs (ξ^μ, η^μ) of a pattern ξ^μ and a label η^μ are drawn from a random ensemble of P pairs (pattern, label). The pattern components ξ_i^μ on all N input units and labels η^μ are random independent variables. We assume that each component ξ_i^μ ($i = 1 \ldots N$ is the unit index and $\mu = 1 \ldots P$ is the pattern index) is activated to 1 with probability f, called "coding level," and otherwise is 0, and that the label η^μ takes one of the following two values: $\eta^\mu = +1$ with probability y, called the "output sparseness," and $\eta^\mu = -1$, otherwise:

$$\xi_i^\mu = \begin{cases} 1, & \text{with probability } f \\ 0, & \text{with probability } 1\text{-}f \end{cases}$$

$$\eta^\mu = \begin{cases} 1, & \text{with probability } y \\ -1, & \text{with probability } 1\text{-}y \end{cases} \quad (3.1)$$

We have chosen different representations for the input and output variables for mathematical convenience. One can go from $\{1, 0\}$ representation to $\{1, -1\}$ and vice versa by changing the threshold of readout neurons.

Learning rule and the synaptic current

The linear threshold readout, or perceptron, classifies its inputs based on the sign of the weighted sum of the input components. This sum is sometimes called the "synaptic current," as it is viewed as modeling the synaptic current into a biological neuron, as follows:

$$h = \sum_{i=1}^{N} w_i \xi_i.$$

We say that the network has learned the association between P input patterns $\vec{\xi}^\mu$, and P labels η^μ if for any pattern μ, as follows:

$$\text{sign}(h^\mu - \theta) = \text{sign}\left(\sum_{i=1}^{N} w_i \xi_i^\mu - \theta \right) = \eta^\mu,$$

where θ is the threshold, which we further assume to be equal to zero.

Training the network means finding the set of weights w_i that satisfies the above expression for all P patterns.

The Hebbian-like learning rule, which we use to train the weights $\{w_i\}$ of the classifier is as follows:

$$w_i = \frac{1}{\sqrt{P}} \left(\sum_{\mu=1}^{P} (\xi_i^\mu - f)(\eta^\mu + 1 - 2y) - (1-f)(1-2y) \right). \quad (3.2)$$

In the case when patterns are equally likely to belong to either class ($y = \frac{1}{2}$), the learning rule simplifies to the following:

$$w_i = \frac{1}{\sqrt{P}} \sum_{\mu=1}^{P} (\xi_i^\mu - f)\eta^\mu.$$

Here and in all that follows, we set the threshold θ to zero.

After training, the synaptic current in response to a test pattern $\vec{\xi}^\nu$ is as follows:

$$h^\nu = \sum_{i=1}^{N} w_i \xi_i^\nu = \sum_{i=1}^{N} \frac{1}{\sqrt{P}} \left(\sum_{\mu=1}^{P} (\xi_i^\mu - f)(\eta^\mu + 1 - 2y) - (1-f)(1-2y) \right) \xi_i^\nu. \quad (3.3)$$

If $\vec{\xi}^\nu$ together with its label η^ν was part of the training set, we can split the sum over patterns into the contribution from the presented pattern $\vec{\xi}^\nu$ and the contribution from other learned patterns as follows:

$$h^\nu = \frac{1}{\sqrt{P}} \left((1-f)\eta^\nu \sum_{i=1}^{N} \xi_i^\nu + \sum_{i=1}^{N} \left[\sum_{\mu \neq \nu}^{P} (\xi_i^\mu - f)(\eta^\mu + 1 - 2y) \right] \xi_i^\nu \right). \quad (3.4)$$

Here we used $(\xi_i^\nu)^2 = \xi_i^\nu$ because ξ_i^ν takes value 0 or 1.

We denote the number of active input units for the pattern ν by n^ν, as follows:

$$n^\nu = \sum_{i=1}^{N} \xi_i^\nu. \quad (3.5)$$

The variable n^ν is drawn from a binomial distribution of N trials with probability f, $\mathbf{B}(N, f)$, for each realization of the random patterns. Its expected value is determined by the number of input units N and the coding level f, as follows:

$$\langle n^\nu \rangle = Nf. \quad (3.6)$$

(Here and throughout this text, the angular brackets denote the mean over the realizations of the input patterns.)

We replace the sum in the square brackets of Equation 3.4 with $2\sqrt{Pf(1-f)y(1-y)}z_i^\nu$, where we have introduced a noise random variable, z_i^ν, with zero mean and unit variance. The coefficient is concluded from the fact that each individual term $(\xi_i^\mu - f)(\eta^\mu + 1 - 2y)$ has variance, as follows:

$$[f(1-f)^2 + (1-f)f^2][y(2-2y)^2 + (1-y)4y^2] = 4f(1-f)y(1-y), \quad (3.7)$$

and the fact that the ξ_i^μ variables are mutually independent. By the central limit theorem, the noise variables z_i^ν can be approximated as Gaussian in the limit $P \to \infty$ with finite f. The sum $\sum_{i=1}^{N} z_i^\nu \xi_i^\nu$ is also Gaussian, with the variance equal to n^ν, $\sum_{i=1}^{N} z_i^\nu \xi_i^\nu = \sqrt{n^\nu} z^\nu$, with z^ν being a Gaussian random variable with zero mean and unit variance.

In terms of z^ν and n^ν, the synaptic current is written as follows:

$$h^\nu = \frac{1}{\sqrt{P}}(1-f)n^\nu \eta^\nu + 2\sqrt{f(1-f)y(1-y)n^\nu} z^\nu. \quad (3.8)$$

If a pattern belongs to either class with equal probability ($y = 1/2$), this expression simplifies to the following:

$$h^\nu = \frac{1}{\sqrt{P}}(1-f)n^\nu \eta^\nu + \sqrt{f(1-f)n^\nu} z^\nu. \quad (3.9)$$

Note that the first term is the one that reflects the correct classification of the input pattern, and the second one represents the noise caused by the interference from other patterns that were learned by the perceptron. The important parameter is the ratio of the two, which is proportional to $\sqrt{\frac{n^\nu}{P}}$.

Integrating over the distribution of z^ν in the appropriate limits gives the probability of h^ν to have the same sign as η^ν. Requiring this probability to exceed $1 - \epsilon$, where ϵ is the tolerated error rate, leads to the capacity of a fully connected readout, as follows:

$$P = \frac{1-f}{2[\text{erf}^{-1}(1-2\epsilon)]^2} N.$$

The capacity is extensive in (grows linearly with) the number of input neurons; however, this architecture requires the number of converging connections that also grows linearly with N (see Fig. 3).

Committee machine

We now turn to deriving the classification capacity of a committee machine, the network shown on the Figure 1*b*, where each of M perceptrons

receives feedforward connections from C_F input units. The connectivity C_F does not scale when the number of input units N increases.

The final decision is the majority vote of the classifiers. In other words, if classification is accurate:

$$\mathrm{sign}\left(\frac{1}{M}\sum_{k=1}^{M}\mathrm{sign}\left(\sum_{i\in I_k} w_i^k \xi_i^\mu - \theta\right)\right) = \eta^\mu.$$

Here $I \in I_k$ stands for all the input units (there are C_F of them) that are connected to the readout k, and w_i^k is the strength of the connection from the input unit i to the readout k (for the learning rule, we consider that w_i^k does not depend on k).

The synaptic current into the readout unit k when pattern $\vec{\xi}^\nu$ is presented is determined by the following:

$$h_k^\nu = \frac{1}{\sqrt{P}}(1-f)n_k^\nu\eta^\nu + \sqrt{f(1-f)n_k^\nu}z_k^\nu. \tag{3.10}$$

The number of active inputs connected to the perceptron k, n_k^ν is drawn from the binomial distribution $\mathbf{B}(C_F, f)$ of now C_F trials with the success rate f; and its expectation value is as follows:

$$\langle n_k^\nu\rangle = C_F f. \tag{3.11}$$

Since the number of connections per readout C_F stays constant as the number of patterns P and the size of the network (N and M) grow, the probability of a single perceptron to classify a pattern correctly approaches the chance level. Indeed, in contrast to the fully connected perceptron, the number of active inputs n_k^ν per readout neuron does not change with the size of the network (Eq. 3.11). Hence, the first term of the expression (Eq. 3.10) decreases in the absolute value as the number of patterns P grows, while the typical value of the second term stays the same. However, there is always a slight tendency toward the correct answer ($\langle h_k^\nu\eta^\nu\rangle > 0$), which can be used by having a growing number of sparsely connected classifiers that take a collective decision by majority vote. This scheme is known by the name of committee machine and has been shown to largely exceed the performance of a single classifier.

It is important to note that in order for the capacity of a committee machine to keep increasing as new classifiers (committee members) are added, the responses of different classifiers should stay sufficiently independent from each other. In the case of limited connectivity, which we consider here, the correlations automatically become smaller and smaller as we increase the number of input units. This happens because the probability of a typical pair of readout neurons to have a common input unit, and thus correlated responses, decreases. In order for the correlations not to be a limiting factor of the classification capacity, we need to increase the number of input units linearly with the number of perceptrons. If one introduces some other mechanism of reducing the correlations between the responses of the classifiers with common input units (e.g., making different perceptrons learn different sets of patterns), a sublinear scaling of the number of input units N with the number of perceptrons M might be sufficient.

Nonoverlapping case

The majority vote of M linear threshold classifiers is given by the average vote, as follows:

$$r^\nu = \frac{1}{M}\sum_{k=1}^{M} r_k^\nu, \qquad r_k^\nu = \mathrm{sign}(h_k^\nu), \tag{3.12}$$

where h_k^ν is given in Equation 3.10. Positive $r^\nu\eta^\nu$ means that the pattern ν is classified correctly.

The expectation value of r^ν follows from Equation 3.10 after integrating over the noise variable z_k^ν, which is approximated to be normally distributed. We make an assumption $Pf \gg n_k^\nu$, which is justified for a large number of patterns and allows us to use the approximation of the error function for small arguments to get the following:

$$\langle r^\nu\rangle = \langle \mathrm{sign}(h_k^\nu)\rangle_{n_k^\nu, z_k^\nu} = \left\langle \mathrm{erf}\frac{\sqrt{(1-f)n_k^\nu}\eta^\nu}{\sqrt{2Pf}}\right\rangle = \sqrt{\frac{2(1-f)}{\pi Pf}}\langle\sqrt{n_k^\nu}\rangle\eta^\nu. \tag{3.13}$$

The expectation value $\langle\sqrt{n_k^\nu}\rangle$ is computed over the binomial distribution $\mathbf{B}(C_F, f)$ as follows:

$$\langle\sqrt{n_k^\nu}\rangle = \sum_{n=0}^{C_F}\binom{C_F}{n} f^n(1-f)^{C_F-n}\sqrt{n}. \tag{3.14}$$

In the dense regime $C_F f \gg 1$, it can be approximated by the following:

$$\langle\sqrt{n_k^\nu}\rangle = \sqrt{C_F f}, \tag{3.15}$$

and, in the extremely sparse case, when $C_F f \gg 1$ and only $n_k^\nu = \{0, 1\}$ are encountered substantially often, by the following:

$$\langle\sqrt{n_k^\nu}\rangle = C_F f. \tag{3.16}$$

To proceed with deriving the classification capacity, let us start with independent classifiers first. The independence of the responses can be achieved either by forcing the connections to be nonoverlapping or by assuming an additional mechanism that, for example, causes different classifiers to update their incoming connections in response to different subsets of the input patterns.

In this case, r^ν can be thought of as drawn from a Gaussian distribution with the mean given by Equation 3.13 and the variance, as follows:

$$\mathbf{cov}(r^\nu, r^\nu) = \frac{1}{M}(1 + \mathbb{O}(P^{-1})). \tag{3.17}$$

The Gaussian assumption is justified by the law of large numbers.

From here on we ignore the contributions of the subleading order, $\mathbb{O}(P^{-1})$ in this case.

The probability p_{correct} to classify a pattern correctly ($r^\nu\eta^\nu > 0$) can then be easily computed.

Fixing the tolerated error rate ϵ and requiring $p_{\mathrm{correct}} > 1-\epsilon$ leads to the expression for the maximal number of input patterns that can be classified with the accuracy $1-\epsilon$, as follows:

$$P_{\max} = \frac{\langle\sqrt{n}\rangle^2}{f}\frac{1-f}{\pi(\mathrm{erf}^{-1}(1-2\epsilon))^2}M. \tag{3.18}$$

Here $\langle\sqrt{n}\rangle$ denotes the average over binomial distribution, $n \sim \mathbf{B}(C_F, f)$.

This result only holds for the case of nonoverlapping connections or in the presence of a decorrelation mechanism. In the following section, we generalize it to random connectivity.

Correction to classification capacity due to overlap in the connections

To derive an analogous expression for the overlapping case without a decorrelation mechanism, we need to compute the variance, as follows:

$$\mathbf{cov}(r^\nu, r^\nu) = \langle(r^\nu - \langle r^\nu\rangle)^2\rangle, \tag{3.19}$$

of the average vote r^ν, defined by Equation 3.12, taking into account the correlations of individual votes r_k^ν.

We start by splitting the covariance into diagonal and nondiagonal contributions, as follows:

$$\mathbf{cov}(r^\nu, r^\nu) = \frac{1}{M^2}\sum_{k=1}^{M}\sum_{l=1}^{M}\mathbf{cov}(r_k^\nu, r_l^\nu) =$$

$$= \frac{1}{M^2}\sum_{k=1}^{M}\mathbf{cov}(r_k^\nu, r_k^\nu) + \frac{1}{M^2}\sum_{k=1}^{M}\sum_{l=1}^{M}(1-\delta_{kl})\mathbf{cov}(r_k^\nu, r_l^\nu) \overset{M\to\infty}{=}.$$

$$= \frac{1}{M} + \mathbf{cov}(r_k^\nu, r_l^\nu)_{k\neq l} \tag{3.20}$$

We assume that M and N scale linearly with P and M, $N, P \to \infty$. The leading terms are thus of the order $\frac{1}{M} \sim \frac{1}{N} \sim \frac{1}{P}$, and we ignore all the subleading contributions.

When the classifiers k and l share input units, the correlation between their responses is positive and is closely related to the correlation of the input currents h_k^ν and h_l^ν (see Eq. 3.10).

Let n_{kl}^ν be the number of input units that are connected to both the classifier k and the classifier l and are active in the pattern $\vec{\xi}^\nu$. For a large number of input units N and finite connectivity C_F, we can assume that n_{kl}^ν can be either 0 or 1, but not more. Including the terms corresponding to $n_{kl}^\nu > 1$ would lead to corrections that scale as $1/N$ and become negligible in the limit for large N. The probability of n_{kl}^ν being 1 is given by the following:

$$\mathrm{Prob}(n_{kl}^\nu = 1) = f\frac{C_F^2}{N}.$$

The number of active units that are connected to only one of the two classifiers are denoted by $\tilde{n}_k^\nu$ and $\tilde{n}_l^\nu$, respectively. In the current approximation, both of them can be assumed to be distributed according to a binomial distribution $\mathbf{B}(C_F f)$.

Then, the currents can be written as follows (see Eq. 3.10):

$$\begin{aligned} h_k^\nu &= \frac{1}{\sqrt{P}}(1-f)(\tilde{n}_k^\nu + n_{kl}^\nu)\eta + \sqrt{f(1-f)n_{kl}^\nu}z_{kl}^\nu + \sqrt{f(1-f)\tilde{n}_k^\nu}z_k^\nu \\ h_l^\nu &= \frac{1}{\sqrt{P}}(1-f)(\tilde{n}_l^\nu + n_{kl}^\nu)\eta + \sqrt{f(1-f)n_{kl}^\nu}z_{kl}^\nu + \sqrt{f(1-f)\tilde{n}_l^\nu}z_l^\nu \end{aligned}, \tag{3.21}$$

where z_k^ν, z_l^ν and $z_k^\nu l$ are all independent Gaussian variables with zero mean and unit variance.

To compute the covariance:

$$\mathbf{cov}(r_k^\nu, r_l^\nu) = \langle \mathrm{sign}(h_k^\nu), \mathrm{sign}(h_l^\nu)\rangle - \langle \mathrm{sign}(h_k^\nu)\rangle\langle \mathrm{sign}(h_l^\nu)\rangle, \tag{3.22}$$

we start by integrating over the variables z_k^ν and z_l^ν to get the following:

$$\begin{aligned} \langle \mathrm{sign}(h_k^\nu)\rangle_{z_k^\nu} &= \mathrm{erf}\left(\frac{z_{kl}^\nu}{\sqrt{2}}\sqrt{\frac{n_{kl}^\nu}{\tilde{n}_k}}\right) \\ \langle \mathrm{sign}(h_l^\nu)\rangle_{z_l^\nu} &= \mathrm{erf}\left(\frac{z_{kl}^\nu}{\sqrt{2}}\sqrt{\frac{n_{kl}^\nu}{\tilde{n}_l^\nu}}\right). \end{aligned} \tag{3.23}$$

Then, Equation 3.22 can be evaluated using the following table integral (Geller and Ng, 1971, their Eq. 18, p. 158):

$$\int_0^\infty \mathrm{erf}(az)\mathrm{erf}(bz)e^{-c^2z^2}dz = \frac{1}{c\sqrt{\pi}}\tan^{-1}\frac{ab}{cD}\, D = \sqrt{a^2+b^2+c^2}. \tag{3.24}$$

In the leading order, we get the following:

$$\mathbf{cov}(r_k^\nu, r_l^\nu)_{k\neq l} = \mathbf{cov}(\mathrm{sign}(h_k), \mathrm{sign}(h_l))_{k\neq l} = \frac{1}{N}\varphi_{C_F f}$$

$$\varphi_{C_F f} = \frac{2fC_F^2}{\pi}\left\langle \tan^{-1}\frac{1}{\sqrt{(\tilde{n}_k+1)(\tilde{n}_l+1)-1}}\right\rangle_{\tilde{n}_k,\tilde{n}_l\in\mathbf{B}(C_F f)}. \tag{3.25}$$

In the dense regime ($C_F f \gg 1$), the expression for $\varphi_{C_F f}$ in Equation 3.25 can be approximated as follows:

$$\varphi_{C_F f} = \frac{2C_F}{\pi}, \tag{3.26}$$

which leads to the following:

$$\mathbf{cov}(r_k^\nu, r_l^\nu)_{k\neq l} = \frac{2C_F}{\pi N}, \tag{3.27}$$

while in the sparse approximation ($C_F f \ll 1$):

$$\varphi_{C_F, f} = fC_F^2. \tag{3.28}$$

and

$$\mathbf{cov}(r_k^\nu, r_l^\nu)_{k\neq l} = \frac{fC_F^2}{N} \tag{3.29}$$

Plugging this result into Equation 3.20, we get for the variance of the majority vote r^ν in the overlapping case, as follows:

$$\mathbf{cov}(r^\nu, r^\nu) = \frac{1}{M} + \frac{1}{N}\varphi_{C_F f},$$

which together with Equation 3.13 leads to the maximal number of input patterns that the committee machine can learn to classify with the accuracy $1-\epsilon$, as follows:

$$P_{\max} = \frac{\langle\sqrt{n_k}\rangle^2}{f}\frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{M}{1+\frac{M}{N}\varphi_{C_F f}}. \tag{3.30}$$

Here $\varphi_{C_F f}$ is given in Equation 3.25 and is approximated by Equation 3.26 or 3.28.

If both the number of input units N and the number of classifiers M increase in proportion to each other, the capacity P increases linearly with N (or M).

In the case of dense representations $C_F f \gg 1$, the last expression simplifies to the following:

$$P_{\max} = \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{C_F M}{1+\frac{M}{N}\frac{2C_F}{\pi}}, \tag{3.31}$$

and in the ultrasparse limit $C_F f \ll 1$ to the following:

$$P_{\max} = \frac{1}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{MC_F^2 f}{1+\frac{M}{N}C_F^2 f}. \tag{3.32}$$

Committee machine with recurrent connections

The majority rule scenario already overcomes the limitations of the connectivity of a single perceptron, but this is not the final answer to constructing a classifier with limited connectivity. The reason is that we still need to implement the majority rule and bring the classification signal to the level of a single unit. The naive way to do it would require another final readout that would have to sample the entire population of M intermediate layer perceptrons. Since M has to scale linearly with the number of learned patterns P, the connectivity of the final readout would also have to scale linearly with P (see Eq. 3.30) and would exceed any predetermined limit for a sufficiently large number of learned patterns.

To implement the majority vote of the intermediate perceptrons while keeping the connectivity of any unit in the network limited, we introduce the recurrent connectivity in the layer of perceptrons. Our goal is to have two attractor states of the intermediate layer dynamics that correspond to the two classes. The feedforward input through the connections $\{w_i^k\}$, trained in the same way as before, will be slightly biased in the positive direction for one class of the input patterns and in the negative for the other. This slight bias determines which attractor state the network will choose. It is essential that the attractors are far away from each other and do not become closer when the number of learned patterns P increases. This implies that the final readout will be able to discriminate between these states, and thus indicates the class of the presented pattern, even if its connectivity does not scale with P. It turns out that for binary classification it is enough to have random recurrent connectivity with sufficiently large but not increasing with P number of connections per unit. The weights of these recurrent connections do not have to be tuned (no learning required for recurrent connections).

We compute the probability of the network of recurrently connected readouts to go to the correct attractor (the one assigned to the class of the input pattern presented) as a function of the number of input units N, the number of perceptrons M, and various parameters of the recurrently connected network of perceptrons.

Network topology

The recurrent readout network shown on the right of Figure 1*c* consists of the input layer (green), the intermediate layer of perceptrons (orange), and the final readout unit (purple).

As before, the input layer of N neurons is presented with a random and uncorrelated pattern $(\xi_i^\nu)_{i=1\ldots N}$ from a set of P patterns $(\vec{\xi}^\mu)_{\mu=1\ldots P}$ that the network has learned to classify.

The layer of perceptrons we now call the intermediate layer. It consists of M linear threshold readouts, each of which is connected to a randomly chosen C_F of N input units. Hence, the feedforward connectivity C_F is the number of feedforward inputs that each perceptron receives. The C_F is an important parameter in the problem as it determines the classification capacity of a perceptron considered in isolation. The intermediate layer is recurrently connected. For the case of binary classification, the probability that two units are connected is the same for each pair. The recurrent connections are not plastic and can be chosen to be all of equal strength α.

The recurrent connectivity matrix J_{kl}, $k,l \in [1\ldots M]$ is constructed randomly, as follows:

$$J_{kl} = \begin{cases} \alpha & \text{with probability} \quad \dfrac{C_R}{M} \\ 0 & \text{with probability} \quad 1 - \dfrac{C_R}{M} \end{cases}, \tag{3.33}$$

with a constraint of being symmetric, $J_{kl} = J_{lk}$ for all k, l in $[1\ldots M]$.

Here, C_R is the expected number of recurrent connections per unit, as follows:

$$\left\langle \sum_{l=1}^{M} J_{kl} \right\rangle = \alpha C_R, \qquad \forall\, k \in [1\ldots M], \tag{3.34}$$

where the average is taken over different realizations of the connectivity matrix.

The final layer consists of a single readout unit that is connected to a randomly chosen subset of C perceptrons in the intermediate layer, with the strength of all connections taken to be equal.

We will keep the connectivity parameters C_F, C_R, and C and coding level f at fixed constant values, while sending the number of input units N, the number of intermediate perceptrons M, and the number of patterns P to infinity, as follows:

$$P, M, N \to \infty; \qquad f, C_F, C_R, C \quad \text{are constant.} \tag{3.35}$$

We want to recover the linear scaling of the maximal number of patterns $P_{\max}$ that the network can learn to classify with the number of input units N, which is known to hold for the fully connected perceptron (Cover, 1965).

Discrete time dynamic model

We model the recurrent dynamics as a probabilistic dynamic process in discrete time t with the probabilistic transition rule from a network state at time t to a network state at time $t+1$. Let $s_k(t) \in [1\ldots M]$ be the dynamic variable describing the state of unit k at time t in a recurrent network.

Let $\tilde{h}_k^\nu$ be the total current into the readout unit k, as follows:

$$\tilde{h}_k^\nu(t) = \sum_{l=1}^{M} J_{kl} s_l^\nu(t) + h_k^\nu, \tag{3.36}$$

where the first term corresponds to the recurrent contribution, and the second term represents the feedforward current from the input layer (Eq. 3.10), which is constant in time.

The probabilistic transition rule from the state at time t to the state at time $t+1$ is as follows:

$$s_k(t+1) = \begin{cases} 1, & \text{with probability} \quad \dfrac{1}{1+e^{-2\beta\tilde{h}_k^\nu(t)}} \\ -1, & \text{with probability} \quad \dfrac{e^{-2\beta\tilde{h}_{k(t)}}}{1+e^{-2\beta\tilde{h}_k^\nu(t)}} \end{cases} \tag{3.37}$$

Here β is the inverse temperature parameter for the statistical model of the recurrent dynamics, and it characterizes the level of noise.

We approximate this probabilistic recurrent dynamics with a mean field method.

Mean field analysis of the recurrent dynamics

To compute the capacity of such a recurrent classifier, we analyze the recurrent dynamics in the mean field approximation. The activities of the recurrently connected units are represented by the variables $s_k = \{+1, -1\}$ with $k = 1\ldots M$. The average activation of the recurrently connected intermediate layer in response to the pattern ν is defined as follows:

$$m^\nu = \frac{1}{M}\sum_{k=1}^{M} \langle s_k \rangle_\beta,$$

where $\langle\rangle_\beta$ is the average over the recurrent noise. The mean field equation for the average activation reads as follows:

$$m^\nu = \frac{1}{M}\sum_{k=1}^{M} \tanh(\beta(C_R \alpha m^\nu + h_k^\nu)). \tag{3.38}$$

This equation can be obtained by averaging s_k over the distribution (Eq. 3.37) and by using the self-consistent expression for the recurrent part of the total synaptic current $\tilde{h}_k(t)$. It can also be derived more rigorously by following the standard calculation for the overlaps in the Hopfield network (Amit, 1992). The stored patterns of the Hopfield network are replaced by the eigenvectors of the connectivity matrix J, as every symmetric matrix can be expressed as $J = \sum_{i=1}^{N} e_i e_i^T$, where e_i represents the (non-normalized) eigenvectors. We are interested in the overlap with the eigenvector $e_k^1 = 1$ for all k in $[1\ldots M]$. That this is the eigenvector of the chosen connectivity matrix J can be seen from Equation 3.34. The external current h_k^ν can be easily included in the derivation. The average activation m^ν is close to zero if the amount of active and inactive units is approximately the same. If the majority of the units is in the active state, m^ν will be close to 1, and if the majority is inactive, m^ν will be close to -1.

Here C_R is the average number of connections per unit, α is the strength of recurrent synapses (we assume they are all excitatory and of equal strength), β is the inverse temperature parameter, and h_k^ν is the feedforward input current given by Equation 3.10.

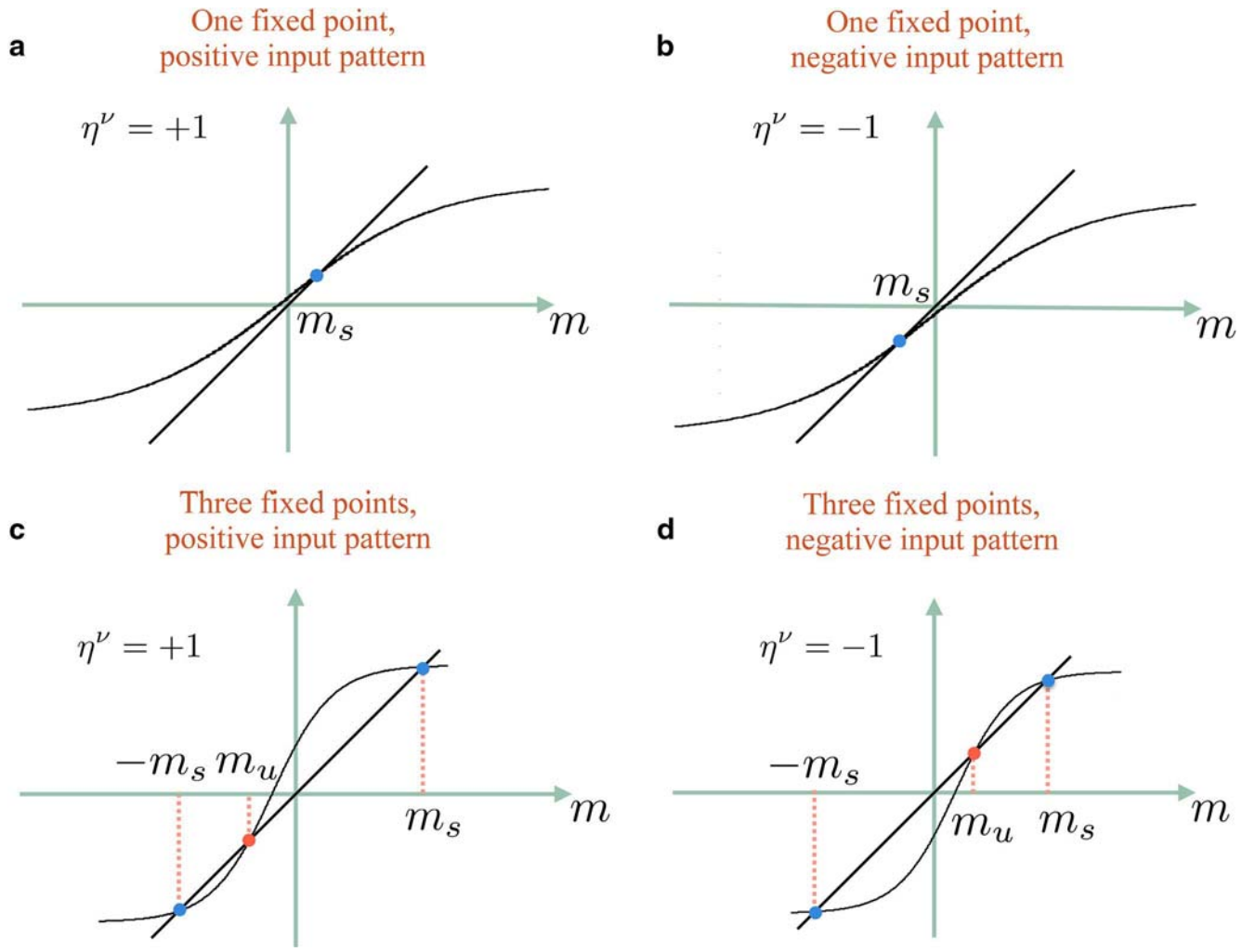

Figure 2. Graphical representation of the mean field equation (Eq. 3.38). The left-hand side of the equation is represented by the line, and the right-hand side, by the sigmoidal curve. The slope of the sigmoidal curve is determined by the amount of noise relative to the strength of the recurrent connections ($\beta C_R \alpha$), and the shift relative to $m = 0$ is based on the expected value of the feedforward input h_k^ν. ***a***, When the input pattern belongs to the positive class and the noise is high, there is only one solution to the equation, which corresponds to the small but positive value $m = m_s$. This solution is stable. ***b***, For the "negative" input pattern, the solution is negative, $m_s < 0$. ***c***, ***d***, In the case of low noise, there are three solutions to the mean field equation, with two extreme solutions m_s and $-m_s$ being stable, and the middle one m_u, which is close to zero, being unstable. For the case of the positive input pattern, $m_u < 0$, and for the case of negative pattern $m_u > 0$.

We proceed by analyzing the above equation graphically. The plot of the right-hand side is a sigmoid curve, and the left-hand side is a line at 45°. The intersections of these two lines determine the solutions to the equation. There are two possible situations that correspond to two different scenarios of the network dynamics.

The first scenario, shown in Figure 2, *a* and *b*, is characterized by having only one point of intersection of the line and the sigmoid. In this case, there is only one solution to the mean field equation (Eq. 3.38) and only one stable state of the recurrent network. The right-hand side of the equation is almost but not quite an odd function of its argument m^ν, so the sigmoidal curve representing it is slightly shifted to the left if $\frac{1}{M}\sum_{k=1}^{M}\tanh(\beta h_k^\nu) > 0$ and to the right if $\frac{1}{M}\sum_{k=1}^{M}\tanh(\beta h_k^\nu) < 0$. If the curve is shifted to the left, the single point of its intersection with the strait line passing thorough the origin will be in the right half-plane. So, for the positive input pattern ($\eta^\nu = +1$ and h_k^ν is more likely to be positive), the mean activity of the intermediate layer in the stable state m^ν will usually be positive, while for the negative input patterns it will be negative. Even though there is a relation between the sign of the mean activity of the intermediate layer in the stable state and the class of the input pattern, this is not helpful for our purposes. The reason is that we encounter exactly the same problem as for the case of no recurrent connections: the absolute value of the average activity m^ν will decrease with the number of learned patterns P, which means that the number of active and inactive units in the intermediate layer will become more and more similar. Consequently, to sample this small imbalance we would require larger and larger connectivity of the final readout. In short, the regime with one stable solution (Fig. 2*a*,*b*) is not much different from the case of no recurrent connections. Not surprisingly, this regime corresponds to relatively weak recurrent connections.

It is the other situation, shown in Figure 2, *c* and *d*, that is actually of interest. There are three points of intersection of the sigmoid curve of the right-hand side of Equation 3.38 and the straight line of the left-hand side. The stable states of the network correspond to the rightmost and the leftmost solutions, which are both characterized by a large imbalance between active and inactive units ($|m^\nu| \sim 1$). Most importantly, these solutions are virtually insensitive to the distribution of h_k^ν, and hence to the number of learned patterns P. So, if we postulate that the right solution corresponds to the positive input patterns and the left solution to the negative ones, it will be easy for a downstream readout with connectivity that does not increase with P to distinguish between them.

The middle intersection point m_u corresponds to the unstable solution. When the network is initialized at the state $\{s_k^0\}$ with $m_0 = \sum_{k=1}^{M} s_k^0$ on the left of the unstable solution $m_0 < m_u$, the recurrent dynamics will most likely evolve to the left stable state; and if initialized at $m_0 > m_u$; it will evolve to the right stable state. As shown in Figure 2, *c* and *d*, the

point of unstable equilibrium will be to the left of the origin for a positive input pattern, and to the right of the origin otherwise (due to the difference in the mean of the distributions of h_k^ν). Hence, initiating the network at $m_0 = 0$ will serve the purpose of biasing the evolution of the network toward the stable state that corresponds to the class of the input pattern. If the number of learned patterns P is large, the point of unstable equilibrium is very close to zero $|m_u| \sim \frac{1}{\sqrt{P}}$, this is the manifestation of the same problem as before, namely the decrease of the signal-to-noise ratio with the increasing number of learned patterns. Thus, the noise in the initial state of the network m_0 should also decrease as $\frac{1}{\sqrt{P}}$. This is achieved if all of the units in the intermediate layer are initialized at $s_k^0 = \pm 1$ with equal probabilities independent from each other, and the number of units M is linear in P (the same scaling as for the committee machine discussed earlier). We use this initialization process to derive the classification capacity and to run the simulations. In the section The initial condition of the recurrent network, we suggest a biologically plausible way to initialize the network at the desired point.

To summarize, the information about the class of the input pattern is contained in the feedforward input to the intermediate recurrently connected layer. In the case of a single stable state (Fig. 2*a*,*b*), although the average activity of the network reflects this information, the signal is very small and a fully connected downstream readout is required. In the case of two stable states (Fig. 2*c*,*d*), this small signal biases the network to choose the one corresponding to the class of the input pattern, and by doing so, the network amplifies the feedforward signal, making it easy to read out by a sparsely connected downstream unit.

Number of classifiable inputs

As discussed in the previous section, the requirement for the correct classification of an input pattern by means of a recurrently connected committee machine is that the average activity of the network at the initial moment m_0^ν is on the correct side of the point of unstable equilibrium m_u^ν, namely the following:

$$(m_0^\nu - m_u^\nu)\eta^\nu > 0, \tag{3.39}$$

where η^ν is the desired output ($\eta^\nu = \{\pm 1\}$).

In what follows we drop the pattern index ν.

The statistics of m_0 over random initializations of the network follows from its definition, as follows:

$$m_0 = \sum_{k=1}^{M} s_k^0,$$

where each unit is initialized at $s_k = +1$ or $s_k = -1$ with equal probability, as follows:

$$\langle m_0 \rangle = 0$$

$$\mathbf{cov}(m_0, m_0) = \langle (m_0 - \langle m_0 \rangle)^2 \rangle = \frac{1}{M}.$$

Since M is a large number, we approximate the distribution of m_0 by a Gaussian distribution with these mean and variance values.

The position of the unstable equilibrium point m_u, corresponding to one of the three solutions (the one that is close to zero) of the mean field equation (Eq. 3.38), cannot be computed analytically in the general case. However, there are parameter regimes in which we can compute the approximate first- and second-order statistics of m_u over random realizations of the input patterns. These parameter regimes and corresponding approximations are discussed in the following section. Once the mean μ_u value, which depends on the number of learned patterns P, and the variance σ_u^2 of m_u are known, the requirement to classify P input patterns with accuracy $1 - \epsilon$ can be written (assuming the distribution of m_u to be also Gaussian), as follows:

$$1 - \epsilon = \frac{1}{2} + \frac{1}{2}\mathrm{erf}\frac{-\mu_u(P)\eta}{\sqrt{2\left(\frac{1}{M} + \sigma_u^2\right)}}. \tag{3.40}$$

The expected number P of correctly classified patterns can be found by inverting the above equation.

In the following sections, we consider different parameter regimes that lead to different approximations for μ_u and σ_u.

The uniform regime

In the current study, among other issues we are interested in the consequences of the sparsity of input representations. Since we consider the feedforward connectivity C_F to be a constant number and not to scale with the size of the network, for sparse representations there will be a substantial number of perceptrons that receive zero feedforward input. Unless the dynamic noise is very high, these units should be considered separately, and in the mean field approximation an additional order parameter should be introduced to describe their average activity [in the derivation of the mean field equation, the overlap with the uniform eigenvector is never the only one with a macroscopic value]. We call these units "free units."

The uniform regime is the parameter regime under which it is not necessary to analyze the free units separately, and Equation 3.38 is valid without modifications. Obviously, when the input representations are dense, $C_F f \gg 1$, the network of the intermediate layer is in the uniform regime, since there are not enough free units to make a difference. However, it is valid to assume the uniform regime even for sparse representations, when the dynamic noise is sufficiently large. To be more precise, the dynamic noise should be large when compared with the typical feedforward input (see the next section).

The conditions defining the uniform regime are as follows:

Sparse input representations and high noise $C_F f \lesssim 1 \quad \beta^{-1} \gg \sqrt{f}$
or
Dense input representations $C_F f \gg 1$

Uniform regime, high noise

One approximation we can make to find the unstable solution m_u of the mean field equation (Eq. 3.38) is the high-noise approximation, which is defined by the following requirement:

$$\beta h_k^\nu \ll 1 \quad \text{for most readouts k and patterns } \nu. \tag{3.41}$$

It follows from Equation 3.10 for the feedforward current that this requirement is met if:

$$\beta \sqrt{f(1-f)\langle n \rangle_{n\neq 0}} \ll 1, \tag{3.42}$$

where $\langle n \rangle_{n\neq 0}$ stands for the mean of the number of active inputs per readout n over the binomial distribution $n \sim \mathbf{B}(C_F, f)$, with the instances of $n = 0$ excluded. For large values of C_F, it can be approximated as $\langle n \rangle_{n\neq 0} = \frac{C_F f}{1 - e^{-C_F f}}$, and the above condition becomes the following:

$$\beta^{-1} \gg \sqrt{\frac{C_F f^2 (1-f)}{1 - e^{-C_F f}}}. \tag{3.43}$$

The condition for having three solutions of Equation 3.38 rather than one (Fig. 2) is as follows:

$$C_R \alpha > \beta^{-1}.$$

Since we are looking for the solution, which is close to zero and Equation 3.42 is satisfied for most of the terms, Equation 3.38 can be approximated by replacing the hyperbolic tangent by its argument, as follows:

$$m_u^\nu = \frac{1}{M}\sum_{k=1}^{M} \beta(C_R \alpha m_u^\nu + h_k^\nu),$$

(note that this approximation is also valid for the terms with $h_k^\nu = 0$).

Solving this equation leads to the mean μ_u and the standard deviation σ_u of m_u:

$$\mu_u = -\frac{1}{C_R\alpha - \beta^{-1}}\mu_h$$

and

$$\sigma_u = \frac{\sigma_h}{C_R\alpha - \beta^{-1}}\sqrt{\frac{1}{M} + \frac{C_F}{N}}. \tag{3.44}$$

The mean μ_h and the standard deviation σ_h of the feedforward current h_k^ν are computed from Equation 3.10, as follows:

$$\begin{aligned}\mu_h &= \frac{1}{\sqrt{P}} f(1-f)C_F\eta \\ \sigma_h &= \sqrt{C_F f^2(1-f)}\end{aligned}. \tag{3.45}$$

The C_F/N term in Equation 3.44 comes from the correlations between the feedforward currents h_k^ν into different readouts k due to overlapping connections (Kushnir and Fusi, 2017, their Appendix A1).

Now the maximum number of learned patterns for the classifier in the uniform regime for high-noise approximation can be computed from Equation 3.40 and is given by the following:

$$P = \frac{1-f}{2[\mathrm{erf}^{-1}(1-2\epsilon)]^2}\frac{C_F M}{1 + \frac{M}{N}C_F + \frac{(\beta C_R\alpha - 1)^2}{C_F f^2(1-f)\beta^2}}. \tag{3.46}$$

We note that because of the applicability condition (Eq. 3.43) making the last term in the denominator small requires fine tuning of the parameter β.

Uniform regime, low noise

The other approximation in which Equation 3.38 can be solved is as follows:

$$\beta h_k^\nu \gg 1, \tag{3.47}$$

which is true for most neurons if:

$$\beta^{-1} \ll \sigma_h = \sqrt{f^2(1-f)C_F}.$$

Under this condition, assuming that the uniform regime is valid only if the input representations are dense:

$$C_F f \gg 1$$

The condition for having three solutions to the mean field equation in the low-noise approximation becomes (see Eq. 3.51) the following:

$$\sigma_h = \sqrt{f^2(1-f)C_F} < \sqrt{\frac{2}{\pi}}C_R\alpha. \tag{3.48}$$

In this case, the hyperbolic tangent in Equation 3.38 can be approximated by the sign function, as follows:

$$m^\nu = \frac{1}{M}\sum_{k=1}^{M}\mathrm{sign}[\beta(C_R\alpha m^\nu + h_k^\nu)].$$

Let us denote the right side of this equation by $g(m_u)$, where:

$$g(m) = \frac{1}{M}\sum_{k=1}^{M}\mathrm{sign}(C_R\alpha m + h_k^\nu) \tag{3.49}$$

is a stochastic function over different realizations of $\{h_k^\nu\}$.

Note that in this case, having a substantial fraction of terms with $h_k^\nu = 0$ would lead to a discontinuity of the right-hand side at $m_u^\nu = 0$.

The mean $\langle g(m)\rangle$ can be found by integrating over the distribution of h_k^ν (see Eq. 3.10), as follows:

$$\langle g(m)\rangle = \mathrm{erf}\left(\frac{C_R\alpha m + \mu_h}{\sqrt{2}\sigma_h}\right), \tag{3.50}$$

where μ_h and σ_h are the mean and standard deviation of h_k^ν, respectively, which are given by Equation 3.45.

Thus, when averaged over training patterns, the mean field equation becomes the following:

$$m = \mathrm{erf}\left(\frac{C_R\alpha m + \mu_h}{\sqrt{2}\sigma_h}\right), \tag{3.51}$$

and it has three solutions when the derivative of the right-hand side with respect to m at $m = 0$ is larger than 1, which, for $\mu_h \ll \sigma_h$, leads immediately to Equation 3.48.

We now return to estimating the mean and the standard deviation of m_u, which is the unstable solution to the approximated mean field equation

$$m_u = g(m_u), \tag{3.52}$$

where $g(m)$ is defined by Equation 3.49.

For $\mu_h \ll \sigma_h$, which is always the case if the number of stored patterns P is large enough, we assume that $C_R\alpha m_u$ is also small compared with σ_h and check the self-consistency later. Then, we can use the approximation for the error function at small arguments to get the following:

$$\langle g(m)\rangle = \sqrt{\frac{2}{\pi}}\frac{C_R\alpha m + \mu_h}{\sigma_h}, \tag{3.53}$$

in which the variance of $g(m)$ can be written as the sum of the diagonal and the nondiagonal terms, as follows:

$$\mathbf{cov}(g(m), g(m)) = \frac{1}{M} + \mathbf{cov}(\mathrm{sign}(C_R\alpha m + h_k), \mathrm{sign}(C_R\alpha m + h_l))_{k\neq l}, \tag{3.54}$$

which is similar to Equation 3.20 for the variance of $\frac{1}{M}\sum_{k=1}^{M}\mathrm{sign}(h_k)$ computed previously in Equation 3.25. The only difference is that here the distribution of h_k is shifted by $C_R\alpha m$. However, because the mean $\langle h_k^\nu\rangle$ did not affect the result (Eq. 3.25) and $C_R\alpha m_u + \mu_h$ is still negligible compared with σ_h, we can write the following:

$$\mathbf{cov}(g(m), g(m)) = \frac{1}{M} + \frac{\varphi_{C_F f}}{N}, \tag{3.55}$$

where $\varphi_{C_F f}$ is given in Equation 3.25.

As a sum of large number M of weakly correlated terms, $g(m)$ can be assumed to be normally distributed and can be written as follows:

$$g(m) = \sqrt{\frac{2}{\pi}}\frac{C_R\alpha m + \mu_h}{\sigma_h} + \sqrt{\frac{1}{M} + \frac{\varphi_{C_F f}}{N}}z^\nu, \tag{3.56}$$

where z^ν is a Gaussian variable with zero mean and unit variance.

Plugging the expression for $g(m)$ into Equation 3.52 and solving for m_u, we get the following:

$$m_u = -\frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sigma_h} - 1}\sqrt{\frac{2}{\pi}}\frac{\mu_h}{\sigma_h} + \frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sigma_h} - 1}\sqrt{\frac{1}{M} + \frac{1}{N}\varphi_{C_F f}}z^\nu, \tag{3.57}$$

where $\varphi_{C_F f}$ is given in Equation 3.25.

So, the expectation value of m_u is as follows:

$$\mu_u = -\frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sigma_h} - 1}\sqrt{\frac{2}{\pi}}\frac{\mu_h}{\sigma_h}$$

$$= -\frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sqrt{C_F f^2(1-f)}} - 1}\sqrt{\frac{2}{\pi}}\frac{\sqrt{C_F(1-f)}}{\sqrt{P}}\eta,$$

and the standard deviation is given by the following:

$$\sigma_u = \frac{1}{\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sqrt{C_F f^2(1-f)}} - 1}\sqrt{\frac{1}{M} + \frac{1}{N}\varphi_{C_F f}}.$$

Because uniform regime and low noise imply dense input representation, we can use the dense approximation (Eq. 3.26) for $\varphi_{C_F f}$. Plugging these results into Equation 3.40 leads to the capacity for the uniform regime, low noise, as follows:

$$P_{\max} = \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{C_F M}{1 + \frac{M}{N}\frac{2}{\pi}C_F + \left(\sqrt{\frac{2}{\pi}}\frac{C_R\alpha}{\sqrt{C_F f^2(1-f)}} - 1\right)^2}.$$

Nonuniform regimes

When the input representation is sparse:

$$C_F f \lesssim 1, \tag{3.58}$$

there is a substantial fraction of perceptrons for which all inputs are silent; we call them the free units. If the noise is not very high $\beta\sqrt{f} \gtrsim 1$, these units are statistically different from those that do receive a nonzero input. To analyze such a system in the mean-field approximation, two order parameters and two coupled mean field equations should be introduced. To avoid this complication, we consider a simpler case, which we refer to as the "two-subnetwork regime." This regime is characterized by the recurrent connections that are relatively weak when compared with the feedforward connections, so that the state of those units that do receive nonzero feedforward input is determined by this input only. Neither recurrent input nor noise can flip them. Only the free units participate in the recurrent dynamics, and their mean activity in the final state reflects the class of the input pattern. Which of the two stable states the subnetwork of free units will go to is biased by the input from the input-receiving units, which have the information about the class of the input pattern from the feedforward input.

This approximation is valid if:

$$\alpha\sqrt{C_R} \ll \sqrt{f} \tag{3.59}$$

$$\beta^{-1} \ll \sqrt{f}. \tag{3.60}$$

To be more precise, this condition does not guarantee that the recurrent input will not be able to flip the input-receiving units close to the final state, when most of the free units are aligned. However, if this is the case, their activity already reflects the correct classification of the input pattern, and the input-receiving units will flip in the right direction.

The mean field equation (Eq. 3.38) should now be seen as describing the subnetwork of free units, and should be modified in several ways.

First, the number of units in the network is

$$M_f = Me^{-C_F f}, \tag{3.61}$$

since for small f the probability of all C_F independent inputs to be silent is $(1-f)^{C_F} \approx \mathrm{e}^{-C_F f}$. Second, only $C_R\mathrm{e}^{-C_F f}$ of C_R recurrent connections per unit come from other free units. Also, the external input to the network now comes from other (input-receiving) units in the intermediate layer, rather than from the input layer.

The modified mean-field equation reads as follows:

$$\tilde{m}^\nu = \frac{1}{M_f}\sum_{k=1}^{M_f}\tanh(\beta(C_R\alpha e^{-C_F f}\tilde{m}^\nu + H_k^\nu)), \tag{3.62}$$

where $\tilde{m}^\nu$ is the average activity of the subnetwork of free units and the index k runs over all the free units.

The external input is as follows:

$$H_k^\nu = \sum_{l=1}^{M_{IR}}\alpha J_{kl}\mathrm{sign}(h_l^\nu), \tag{3.63}$$

where the summation is over the input-receiving units and h_k^ν is the feedforward current of Equation 3.10 with $n_l^\nu \neq 0$. M_{IR} is the number of input-receiving units, as follows:

$$M_{IR} = M(1-\mathrm{e}^{-C_F f}).$$

On average, the free unit k receives C_R inputs, and $(1-e^{-C_F f})C_R$ of them come from input receivers. So, Equation 3.63 will have on average $C_R(1-e^{-C_F f})$ nonzero terms. Assuming that this is a large number, H_k^ν is a Gaussian variable with the mean given (in the leading order) by the following:

$$\mu_H = \alpha C_R(1-e^{-C_F f})\langle\mathrm{sign}(h_l^\nu)\rangle_{n_k\neq 0} = \alpha C_R\langle\mathrm{sign}(h_l^\nu)\rangle, \tag{3.64}$$

which, using Equation 3.13, becomes the following:

$$\mu_H = \sqrt{\frac{2(1-f)}{\pi}}\frac{\langle\sqrt{n}\rangle}{\sqrt{Pf}}C_R\alpha\eta^\nu. \tag{3.65}$$

The number of active inputs n connected to the intermediate unit comes from the binomial distribution, $n \sim \mathbf{B}(N, f)$.

The standard deviation of H_k^ν is as follows:

$$\sigma_H = \alpha\sqrt{C_R(1-e^{-C_F f})}, \tag{3.66}$$

(the corrections due to correlations between different input-receiving units are suppressed as $1/N$ and will become negligible for large networks when C_R does not scale with N).

To find the statistics of $\tilde{m}_u$, the point of unstable equilibrium, we again consider high- and low-noise approximations, but now we should compare the inverse temperature parameter β to the standard deviation of H_k^ν.

What we further call intermediate noise is the noise that is small on the scale of the feedforward input (Eq. 3.60) but large when compared with the typical values of H_k^ν.

Two-subnetwork regime, intermediate noise

The following analysis is valid if, in addition to the conditions described in Equations 3.58, 3.59, and 3.60, the dynamic noise is high compared with the typical external input to the subnetwork of free units:

$$\beta\sigma_H = \beta\alpha\sqrt{C_R(1-e^{-C_F f})} \ll 1.$$

The condition for three solutions to the mean field equation (Eq. 3.62) in this case is as follows:

$$\beta\alpha C_R\mathrm{e}^{-C_F f} > 1.$$

The former inequality allows us to approximate the hyperbolic tangent in Equation 3.62 by its argument when looking for the unstable solution $\tilde{m}_u$, which is close to zero, as follows:

$$\tilde{m}_u^\nu = \beta C_R\alpha e^{-C_F f}\tilde{m}_u + \beta\frac{1}{M_f}\sum_{k=1}^{M_f}\sum_{l=1}^{M_{IR}}J_{kl}\alpha\mathrm{sign}(h_l^\nu). \tag{3.67}$$

Each input-receiving unit l has C_R outgoing connections and approximately $\mathrm{e}^{-C_F f}C_R$ of them terminate on a free unit. Hence, the double sum can be rewritten as follows:

$$\tilde{m}_u^\nu = \beta C_R \alpha e^{-C_E f} \tilde{m}_u^\nu + \beta C_R \alpha e^{-C_E f} \frac{1}{M_f} \sum_{l=1}^{M_{IR}} \text{sign}(h_l^\nu). \quad (3.68)$$

Solving this equation for $\tilde{m}_u$ leads (see Eq. 3.61) to the following:

$$\tilde{m}_u^\nu = -\frac{\beta C_R \alpha}{\beta C_R \alpha e^{-C_E f} - 1} (1 - e^{-C_E f}) \bar{r}^\nu, \quad (3.69)$$

where we have introduced $\bar{r}^\nu$: the sign of the feedforward current averaged over the units for which this current is nonzero, as follows:

$$\bar{r}^\nu = \frac{1}{M_{IR}} \sum_{l=1}^{M_{IR}} \text{sign}(h_l^\nu).$$

The statistics of $\bar{r}^\nu$ are closely related to previously computed statistics of r^ν (see Eq. 3.12), which is the sign of the feedforward current averaged over all of the intermediate units, namely:

$$\langle \bar{r}^\nu \rangle = \frac{1}{1 - e^{-C_E f}} \langle r^\nu \rangle. \quad (3.70)$$

The expression for $\langle r^\nu \rangle$ is given in Equation 3.13, which leads to the following:

$$\langle \bar{r}^\nu \rangle = \langle \text{sign}(h^\nu) \rangle_{n^\nu \neq 0} = \frac{1}{1 - e^{-C_E f}} \sqrt{\frac{2(1-f)}{\pi}} \frac{\langle \sqrt{n} \rangle}{\sqrt{Pf}} \eta^\nu. \quad (3.71)$$

To compute the second-order statistics of $\bar{r}^\nu$, we use the following relation:

$$\mathbf{cov}(\text{sign}(h_k^\nu), \text{sign}(h_l^\nu))_{k \neq l; n_k^\nu, n_l^\nu \neq 0} = \frac{1}{(1 - e^{-C_E f})^2} \mathbf{cov}(\text{sign}(h_k^\nu), \text{sign}(h_l^\nu))_{k \neq l}.$$

The covariance on the right-hand side was also computed in Equation 3.25, which allows us to write the following:

$$\mathbf{cov}(\bar{r}^\nu, \bar{r}^\nu) = \frac{1}{M_{IR}} + \frac{\varphi_{C_F, f}}{(1 - e^{-C_E f})^2} \frac{1}{N}. \quad (3.72)$$

Plugging in Equations 3.71 and 3.72 to Equation 3.69 leads to the expressions for the mean and the standard deviation of $\tilde{m}_u^\nu$:

$$\mu_u = -\sqrt{\frac{2(1-f)}{\pi}} \frac{\beta C_R \alpha}{\beta C_R \alpha e^{-C_E f} - 1} \frac{\langle \sqrt{n} \rangle}{\sqrt{Pf}} \eta,$$

[the mean $\langle \sqrt{n} \rangle$ is computed assuming a binomial distribution for the number of active inputs n connected to a readout $n \sim \mathbf{B}(N, f)$], and the following:

$$\sigma_u = \frac{\beta C_R \alpha}{\beta C_R \alpha e^{-C_E f} - 1} \sqrt{\frac{1}{M}(1 - e^{-C_E f}) + \varphi_{C_F, f} \frac{1}{N}}. \quad (3.73)$$

Now we can use Equation 3.40 to compute the maximum number of learned patterns in the two-subnetwork regime under intermediate noise. The number of units in the network M in Equation 3.40 should be replaced by the number of free units $Me^{-C_E f}$. The result is as follows:

$$P = \frac{\langle \sqrt{n} \rangle^2}{f} \frac{1-f}{\pi [\text{erf}^{-1}(1 - 2\epsilon)]^2} \frac{M}{\gamma + \frac{M}{N} \varphi_{C_F, f}},$$

where:

$$\gamma = 1 - \frac{2\beta C_R \alpha - e^{C_E f}}{(\beta C_R \alpha)^2}, \quad 1 - e^{-C_E f} < \gamma < 1.$$

It is helpful for analyzing this result to rewrite the expression for γ in terms of the following:

$$\Delta_{IT} = e^{-C_E f} \beta C_R \alpha - 1, \quad (3.74)$$

which is the measure of how far the current parameters are from the transition to the one-solution scenario (Fig. 2*a*,*b*), at which the current framework breaks down, as follows:

$$\gamma = 1 - e^{-C_E f} \left(1 - \frac{\Delta_{IT}^2}{(\Delta_{IT} + 1)^2}\right).$$

In the ultrasparse approximation:

$$C_E f \ll 1,$$

we can use Equations 3.16 and 3.28 to get the following:

$$P = \frac{1-f}{\pi [\text{erf}^{-1}(1 - 2\epsilon)]^2} \frac{M C_E^2 f}{\gamma + \frac{M}{N} C_E^2 f}.$$

Two-subnetwork regime, low noise

We now consider the low-noise approximation to the mean field equation for the subnetwork of free units (Eq. 3.62). This approximation is valid when in addition to Equations 3.58, 3.59, and 3.60:

$$\beta \sigma_H = \beta \alpha \sqrt{C_R (1 - e^{-C_E f})} \gg 1. \quad (3.75)$$

In this approximation, the mean field equation has three solutions if:

$$\sqrt{\frac{2}{\pi}} \frac{C_R \alpha e^{-C_E f}}{\sigma_H} = \sqrt{\frac{2}{\pi}} \frac{\sqrt{C_R} e^{-C_E f}}{\sqrt{1 - e^{-C_E f}}} > 1.$$

This condition is derived analogously from Equation 3.47.

Under the assumption (Eq. 3.75), the mean field equation (Eq. 3.62) can then be approximated as follows:

$$\tilde{m}^\nu = \frac{1}{M_f} \sum_{k=1}^{M_f} \text{sign}(\beta (C_R \alpha e^{-C_E f} \tilde{m}^\nu + H_k^\nu)). \quad (3.76)$$

As in the section Uniform regime, low noise, let us introduce a stochastic function $g(\tilde{m})$, as follows:

$$g(\tilde{m}) = \frac{1}{M_f} \sum_{k=1}^{M_f} \text{sign}(C_R \alpha e^{-C_E f} \tilde{m} + H_k^\nu).$$

For small values of the argument $\tilde{m}$, the mean of $g(\tilde{m})$ over different realizations of H_k^ν is approximated as follows:

$$\langle g(\tilde{m}) \rangle = \sqrt{\frac{2}{\pi}} \frac{C_R \alpha e^{-C_E f} \tilde{m} + \mu_H}{\sigma_H},$$

where μ_H and σ_H are given by Equations 3.65 and 3.66.

To compute the variance of $g(\tilde{m})$ we need to know the following:

$$\mathbf{cov}(\text{sign}(C_R \alpha e^{-C_E f} \tilde{m} + H_k), \text{sign}(C_R \alpha e^{-C_E f} \tilde{m} + H_p))_{k \neq p} \approx \mathbf{cov}(\text{sign}(H_k), \text{sign}(H_p))_{k \neq p},$$

which is calculated in the study by Kushnir and Fusi, 2017, their Appendix A2) and, for large absolute values of the recurrent connectivity $C_R e^{-C_E f} \gg 1$), is approximated by the following:

$$\mathbf{cov}(g(\tilde{m}), g(\tilde{m})) = \frac{2}{\pi} C_R \left(\frac{1}{M} + \frac{\varphi_{C_F, f}}{N} \frac{1}{1 - e^{-C_E f}}\right). \quad (3.77)$$

Assuming H_k^ν to be Gaussian, we can write the following:

$$g(\tilde{m}) = \sqrt{\frac{2}{\pi}}\frac{C_R\alpha e^{-C_F f}\tilde{m} + \mu_H}{\sigma_H} + \sqrt{\frac{2}{\pi}C_R\left(\frac{1}{M} + \frac{\varphi_{C_F,f}}{N}\frac{1}{1-e^{-C_F f}}\right)}z^{\nu}, \tag{3.78}$$

where z^{ν} is a Gaussian variable with zero mean and unit variance.

The statistics of the unstable, close to zero, solution of Equation 3.76 can now be found by plugging in Equation 3.78 as the right-hand side of Equation 3.76, and solving for $\tilde{m}^{\nu}$.

After substituting Equations 3.65 and 3.66 for μ_H and σ_H, we get the mean and the variance of the unstable solution $\tilde{m}_u$ (assuming $\sqrt{C_R}e^{-C_F f} \gg 1$), as follows:

$$\mu_u = -\sqrt{\frac{2(1-f)}{\pi}}\, e^{C_F f}\frac{\langle\sqrt{n}\rangle}{\sqrt{Pf}}$$

$$\sigma_u^2 = e^{2C_F f}(1-e^{-C_F f})\left(\frac{1}{M} + \frac{\varphi_{C_F,f}}{N}\frac{1}{1-e^{-C_F f}}\right).$$

Using these expressions and Equation 3.40 with M replaced by the number of free units $M_f = Me^{-C_F f}$, we get the maximal number of classifiable inputs in the following low-noise approximation of the two-subnetworks regime, as follows:

$$P = \frac{\langle\sqrt{n}\rangle^2}{f}\frac{1-f}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2}\frac{M}{1+\frac{M}{N}\varphi_{C_F,f}}.$$

Note, that this is the same expression as Equation 3.30 for the majority vote scenario (see the Results for an intuitive explanation).

For very sparse representations:

$$C_F f \ll 1$$

the expression simplifies to the following:

$$P = \frac{1-f}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2}\frac{MC_F^2 f}{1+\frac{M}{N}C_F^2 f}.$$

Results

The task and the network architecture

To evaluate the performance of different network architectures, we consider a task in which the network is trained to associate a specific response with each input. The response is expressed by the activity of one output neuron, which could represent a decision, the expected value of an input stimulus, or an action. Each input, for example a sensory stimulus, is a pattern of activity across N input neurons. Both input and output neurons are either active or inactive, and hence the variables representing their activity are binary. Moreover, we assume that the inputs and the outputs are random and uncorrelated. Input neurons are active with probability f, whereas the output neuron is active on average for half of the inputs. Performing this task is equivalent to solving a binary classification problem in which each input is assigned to belong to one of two possible classes. As a measure of the performance, we consider the classification capacity and the maximum number of input patterns that can be correctly classified, and determine how it scales with the total number of neurons in the network. We now consider architectures with increasing complexity, and we eventually show that it is possible to design a network in which the number of classifiable inputs is large and scales linearly with the number of neurons while each neuron has limited connectivity (i.e., the number of connections per neuron is fixed in the sense that it does not scale with the number of neurons).

Single fully connected readout

The most basic network that we consider is the one in which the input neurons are directly connected to the output, which is basically the classical perceptron (Rosenblatt, 1957; Figs. 1*a*, 3, the first model on the left). The network is trained by modifying the weights w_i that connect each input neuron i (Fig. 3, green) to the output (Fig. 3, yellow). The output activity o^{μ} in response to stimulus μ is determined by thresholding the weighted sum of the inputs:

$$o^{\mu} = \mathrm{sign}\left(\sum_{i=1}^{N} w_i\xi_i^{\mu} - \theta\right),$$

where θ is a threshold and ξ_i^{μ} is the activity of neuron i when input pattern μ is selected. The weights w_i and the threshold θ are learned to impose that $o^{\mu} = \eta^{\mu}$, where η^{μ} is the desired output in response to stimulus μ. We know from many studies (Cover, 1965; Gardner, 1987) that the maximum number P of random inputs that can be correctly classified scales linearly with the number of input units when $f = 1/2$ ($P \sim N$; Fig. 3, table). This is a very favorable scaling and, actually, is the optimal one in the benchmark that we consider. Unfortunately, the number of connections C_F of the output neuron (Fig. 3, feedforward connectivity) is equal to the number of input neurons, and hence when the number of classifiable inputs grows, the connectivity also has to increase accordingly. This is true also in the case of sparse input representations. Indeed, for an arbitrary value of f, when we used the following simple learning rule inspired by Tsodyks and Feigel'Man (1988):

$$w_i = \frac{1}{\sqrt{P}}\sum_{\mu=1}^{P}(\xi_i^{\mu} - f)\eta^{\mu}, \tag{4.1}$$

and we obtained the limit of a large number of input neurons N, the maximum number of input patterns that can be classified P is given by the following:

$$P = \frac{1-f}{2[\mathrm{erf}^{-1}(1-2\epsilon)]^2}N, \tag{4.2}$$

where ϵ is the maximum tolerated error.

Notice that the factor containing the coding level of the patterns f cannot change the scaling properties of P, even in the case in which the inputs become very sparse (i.e., when $f \to 0$ as $1/N$). This seems to be in contradiction with the results of other studies (Tsodyks and Feigel'Man, 1988; Amit and Fusi, 1994) in which P can scale as N^2 when the inputs are sparse. However, it is important to remember that the N^2 scaling can be achieved only when both the input and the output are sparse, and in the cases that we analyzed here the output is dense (i.e., active in half of the cases).

We now consider a different architecture that partially overcomes the limitation imposed by the limited connectivity assumption.

Committee machines

Consider now the architecture of Figure 1*b* (Fig. 3, Committee machine) in which multiple perceptrons are combined. We assume that each perceptron has limited connectivity, or more precisely, that when the number of input neurons becomes large (mathematically, we consider the limit of $N \to \infty$), the number of input connections per perceptron, C_F, does not increase (i.e., C_F remains finite when $N \to \infty$; Fig. 3, flat line). As a consequence, each perceptron will sample only a small fraction of the input neurons, and, for this reason, it will misclassify most of the inputs when P becomes large ($P \to \infty$). More

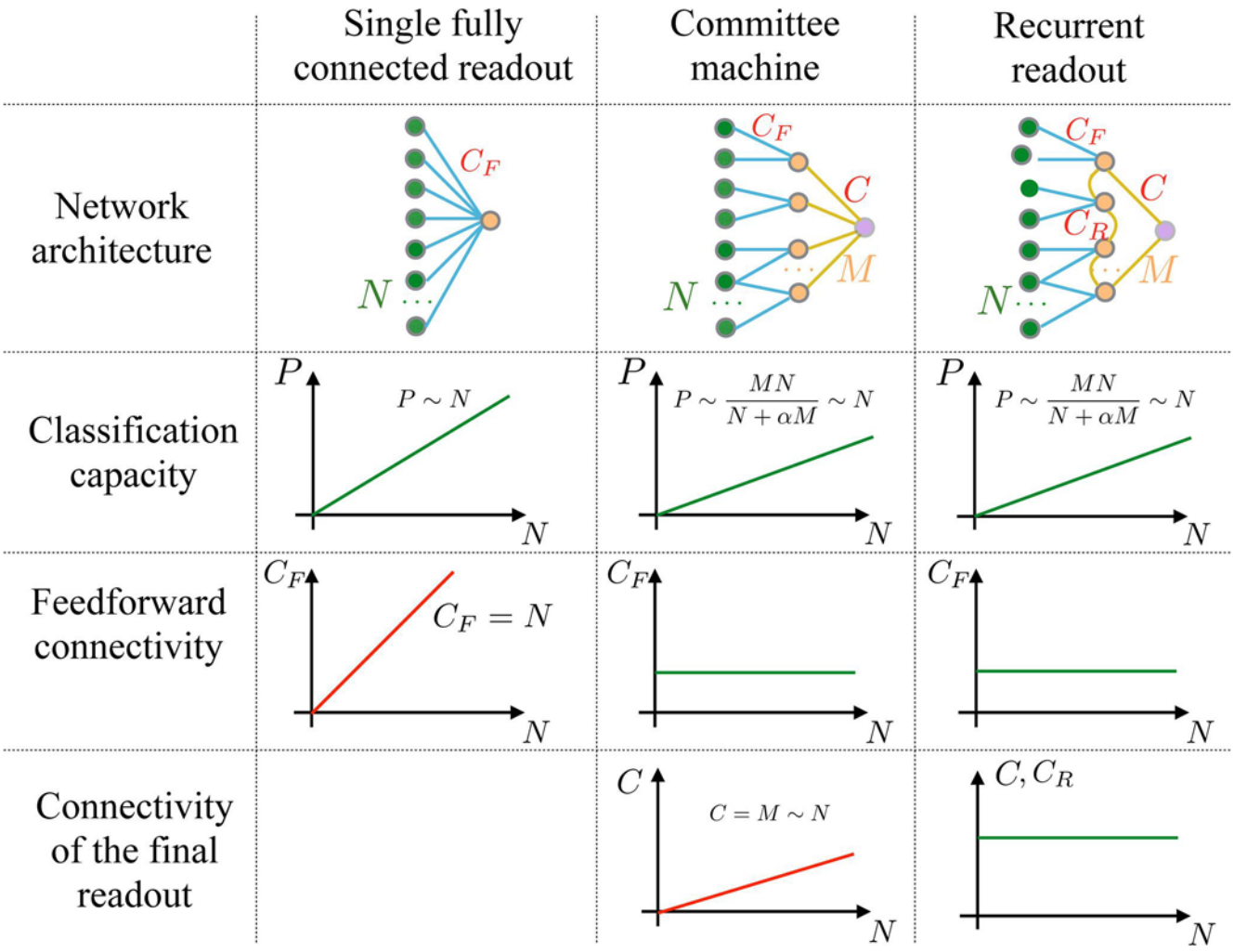

Figure 3. Summary of the scaling properties of the three architectures considered in our study. A single fully connected readout (classical perceptron) achieves a classification capacity P that grows linearly with the number of input neurons N. The input neurons are green and the output neuron is orange. However, the number of feedforward connections that converge onto a single neuron C_F also increases linearly with N. The committee machine with M members (orange neurons) solves this problem by limiting the number of connections C_F per member neuron. This number does not scale with N, and hence it can be relatively small. The classification capacity P still scales linearly with N thanks to the contributions of M partially connected perceptrons, which are combined using a majority vote scheme. The majority vote, however, implies the existence of a final readout, which counts the votes of all the members of the committee. This readout can be implemented with a neuron with C connections, where C is equal to M, and thus scales linearly with N. The suggested recurrent readout architecture on the right achieves the linear growth of the capacity while keeping C_F, C, and the number of recurrent connections per neuron C_R constant as N increases.

quantitatively, the fraction of correctly classified inputs will be slightly above the level of chance (1/2), approximately $1/2 + a/\sqrt{P}$ when P is large (a is a constant).

In this situation, each perceptron is said to be a weak classifier. However, if the responses of different perceptrons are sufficiently independent, they can be combined to perform significantly better than any individual perceptron. The combination of multiple perceptrons makes what is called a committee machine. Typically, the class of an input is decided by the committee using a majority vote rule: if the majority of perceptrons are active, then the output neuron should also be active, otherwise it should be inactive. The majority rule can be easily implemented by summing with equal weights the outputs of all perceptrons.

As mentioned in the Introduction, adding new readouts without increasing the number of input units N cannot increase the classification capacity indefinitely, unless an additional mechanism is introduced to decorrelate the responses of different readouts. Such mechanisms may very well exist in the real brain. For example, one could imagine some local changes of synaptic plasticity during the learning phase, which make different readouts update their connections during the presentation of different subsets of patterns. However, in this article we stick to the simple learning rule (Eq. 4.1) and do not consider any decorrelation mechanisms. So, in the present contexts, the only way of increasing the classification capacity of the network without reaching the saturation is to increase the number of input units N. Also, to satisfy the requirement of limited connectivity, the number of connections converging onto the same readout, C_F cannot increase with N, and we need to add new readouts to connect to the newly added input units. We denote the number of readouts (number of committee members) by M and we derive the classification capacity P under the assumption that N, M, and Pf are large numbers and the C_F connections of every readout (perceptron) are chosen randomly and independently of any other (there will be a random overlap).

If we use the simple local learning rule (Eq. 4.1), the maximum number of classifiable inputs is as follows:

$$P = \frac{\langle\sqrt{n}\rangle^2}{f}\frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{\frac{M}{N}}{1+\frac{M}{N}\varphi_{C_F,f}}, \qquad (4.3)$$

where $\varphi_{C_F,f}$ is of the order of C_F and depends on the coding level f, but not on N or M (see Eq. 3.25). The factor $\langle\sqrt{n}\rangle$ is the mean of the square root of the random variable n over the binomial distribution $\mathbf{B}(C_F, f)$, which is approximately $\sqrt{C_F f}$ in the case $C_F f \gg 1$ (dense approximation) and $C_F f$ in the case of $C_F f \ll 1$ (ultrasparse approximation; in practice, ultrasparse approximation is quite accurate also for $C_F f \approx 1$). As for the single readout, the required classification accuracy is $1-\epsilon$.

Using also the approximations for $\varphi_{C_F,f}$ in these two cases, we get the following:

$$P = \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{C_F\frac{M}{N}}{1+\frac{2}{\pi}C_F\frac{M}{N}}N$$

$$\text{for } C_F f \gg 1 \qquad (4.4)$$

and

$$P = \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2\pi}\frac{C_F^2 f\frac{M}{N}}{1+C_F^2 f\frac{M}{N}}N \qquad (4.5)$$

$$\text{for } C_F f \gg 1.$$

For the dense input representations, $C_F f \gg 1$, if the number of input neurons N is kept constant while the number of perceptrons M is increased, the capacity P will saturate when the total number of feedforward connections $C_F M$ is large compared with N. The value at which P saturates is the same as for the fully connected readout (Eq. 4.2). If the input representations are sparse, $C_F f \lesssim 1$, saturation occurs when $C_F M$ becomes large compared with $N/C_F f$. The saturation value of P in this case differs from the fully connected case by a factor of order one, $2/\pi$.

This implies that the committee machine is less efficient than the fully connected readout when the total number of feedforward connections is considered. This will also be the case for the recurrent readout scheme that we propose in the following section. It should be noted, however, that the difference in the total number of connections is modest (several times) unless the input representations are extremely sparse $C_F f \ll 1$.

The dependence of P on the coding level f is weak, unless $C_F f$ becomes smaller 1. For sparser representations, the capacity becomes proportional to $C_F f$ (unless M is increased). This is not surprising because when $C_F f < 1$, a significant proportion of perceptrons will read out only inactive neurons, which are not informative about the input. However, even for very sparse representations the capacity can be restored by increasing the expansion ratio M/N (see Fig. 5*c*, green line and Fig. 5*f*, low-f region).

When both N and M are increased in proportion to each other, the number of classifiable patterns increases linearly with N, as in the case of the fully connected single perceptron (Fig. 3). However, now the connectivity of each perceptron is C_F, which does not scale with N or M. This means that it is possible to overcome the limitations of sparse connectivity. Unfortunately, this is not a satisfactory solution to the problem of limited connectivity because the readout output neuron, which now has to count the votes of all M members of the committee, needs to be connected to M neurons, and M scales as N. So again, we will need a number of connections per neuron that grows linearly with N. The last row in Figure 3 (connectivity of the final readout) shows this dependence.

Committee machines with recurrent connections

We propose an alternative way of implementing a committee machine, which is based on the use of recurrent connections, and it does not require a fully connected output neuron (Fig. 3). To understand the idea behind it, it is useful to consider a multilayer readout as a way to count the votes of all perceptrons while respecting the limited connectivity constraint: each neuron in the first layer would count the votes of different $\widetilde{C_F}$ perceptrons. The neurons in the second layer would then count the votes of $\widetilde{C_F}$ first-layer neurons, and so on. For this architecture, the number of neurons would decrease by a factor $\widetilde{C_F}$ in every new layer, leading to a total number of neurons that would scale as $\log(M)$ or, equivalently, as $\log(N)$. It is also possible to set up a multilayer network with the same number of layers in which every layer contains the same number of neurons M. This network would require more neurons, but it is functionally equivalent to the first one that we considered. The reason we are considering this network architecture is that it can be interpreted as a recurrent network unfolded in time: if one assumes that the network dynamics is discrete in time, then every layer could be seen as the same recurrent network at a different time step. Importantly, the weights of the synaptic connections should be the same for every layer, as it is always the same network but at different time steps. As this network would also be functionally equivalent to the first multilayer network that we discussed, a recurrent network can in principle replace a complex multilayer readout, which would require significantly more neurons.

These considerations induced us to study the architecture represented in Figure 1*c* (Fig. 3, Recurrent readout), as follows: each perceptron of the committee machine is now connected to a randomly chosen set of the others through recurrent connections, whose weights are all the same and are equal to α. The number of recurrent connections per perceptron is C_R.

The recurrent dynamics basically has the role of stabilizing only two of the following attractor states of the network: one in which all perceptrons are in the active state; and one in which they are all in the inactive state. These two states represent the two possible responses of the output and correspond to the two classes the input could belong to. The recurrent network is an attractor network similar to the one proposed by Hopfield (1982), which in turn took inspiration from the studies on spin glasses in the ferromagnetic state, in which the spins are all aligned to each other in one of two possible directions. These two states of magnetization are analogous to the two decisions of the network that we propose. These attractor models have been used more recently to model decision-making, both in simple, more abstract networks (Usher and McClelland, 2001) and in very detailed biologically plausible networks of spiking neurons (Wang, 2002) in which only the two states corresponding to the possible decisions become stable when a sensory stimulus is presented.

The assumption about the excitatory nature of recurrent connections is crucial to have the two attractor states described above. For the case of binary classification, we assume that recur-

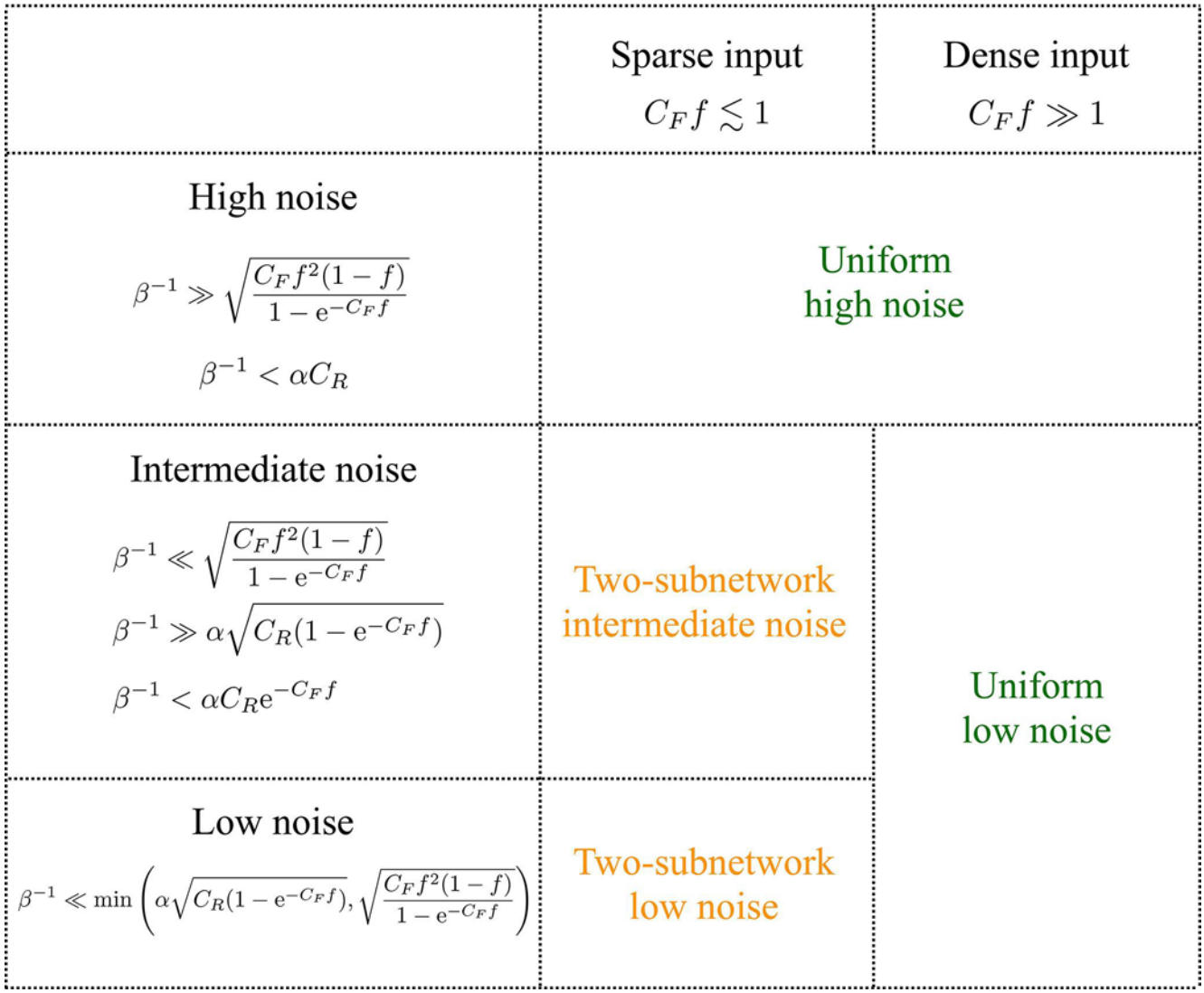

Figure 4. Network regimes depend on the sparseness of the input (determined both by the sparseness of the feedforward connectivity C_F and by the sparsenss f of the input representations) and on the noise level with respect to the recurrent and feedforward inputs. β is the inverse temperature parameter, C_R is the recurrent connectivity, and α is the strength of the recurrent synapses. In the high-noise regime, the network can always be analyzed as a single homogeneous population of neurons (uniform regime). For intermediate and low noise, the network operates in the two-subnetwork regime when the input is sparse, and in the uniform regime when the input is dense. In the two-subnetwork regime, the recurrent neurons should be divided into the following two groups: those that receive a feedforward input and those whose input is zero.

rent connections are either zero or excitatory. For the case of multinomial classification, however, the recurrent connections can also be inhibitory (see subsection Random output).

Once the network has relaxed into one of the two stable states, it becomes easy to determine the class to which the input belongs, as in principle it is sufficient to read out a single perceptron. However, a single neuron readout would not be robust to noise, and hence we will consider the situation in which a number of different perceptrons is read out. We will show that this number remains finite when N and M become large, which is equivalent to saying that it is possible to construct a network in which all of the neurons, including the output neuron, have limited connectivity and the number of classifiable inputs grows linearly with N. These scaling properties are summarized in the last column of Figure 3.

The number of classifiable inputs is derived analytically in Materials and Methods using a mean field approach. This number depends on the parameters that characterize the network architecture (i.e., the number and the connectivity of the different types of neurons) and on the statistics of the inputs that have to be classified. Depending on the assumptions about the parameters, there are different regimes that lead to different analytical expressions. These distinct regimes are summarized in Figure 4 and described in the following sections in great detail. The first distinction relates to whether all the recurrently connected neurons can be considered statistically equivalent or not. We call uniform the regime in which all the neurons can be assumed to be equivalent. This assumption is reasonable except when the connectivity and/or the input representations are so sparse that there is a significant fraction of neurons in the readout layer that do not receive any feedforward input. The number of these neurons depends on the product $C_F f$ (Fig. 4). This population of neurons behaves differently from the others, to the point that it can be considered as a different subnetwork that requires a different analysis (for this reason, we call this a two-subnetwork regime). The type of analysis we perform also depends on the amount of noise in the network. In particular, when the noise is large, the network always operates in a uniform regime, even when the

input is sparse. We now first discuss the uniform regime, and we will cover the nonuniform regime later.

The uniform regimes

The behavior of the network in the uniform regime also depends on the amount of noise that is injected into the neurons. We introduced noise as in the Hopfield model: the state of each neuron is stochastic, and its total synaptic current determines the probability distribution of the states. The noise is characterized by a parameter β, which in the language of statistical mechanics would be the inverse temperature parameter. When β is large, the noise is small and the neurons are basically deterministic. As β goes to zero, the neurons become more noisy and less dependent on the total mean synaptic input.

As we know from previous studies on attractor neural networks (Amit, 1992), the noise cannot be too large, otherwise the attractor states remain stable only for a short time (Fig. 2). More specifically, the noise should be smaller than the recurrent input when the network already settled in one of the two attractors and most of the presynaptic neurons of the recurrent network are in the right state. In the uniform regime, this requirement is expressed as $\beta C_R \alpha > 1$. Moreover, to guarantee attractor stability, the recurrent input should also dominate over the feedforward input. More formally, this condition can be expressed as $A < C_R\alpha$, where $A = \sqrt{\frac{C_F f^2(1-f)}{1-e^{-C_F f}}}$ is approximately the range in which the feedforward synaptic input varies when different inputs are presented. It basically determines the selectivity to the inputs in the absence of the recurrent connections (for more details, see Materials and Methods).

The two conditions on noise versus recurrent input and recurrent input versus feedforward input impose constraints on both A and β. However, the range in which these parameters can vary still allows the network to operate in qualitatively different regimes that depend on how large the noise is compared with the typical amplitude of the feedforward input.

The uniform, high-noise regime. In the high-noise regime, the noise is so large compared with the feedforward input ($\beta^{-1} \gg A$) that all of the different recurrent neurons can behave similarly (uniform regime) even when the feedforward input is so sparse ($C_F f \lesssim 1$) that many neurons receive zero input. In this regime, the noise is large compared with A, but still small compared with the recurrent input. The number of classifiable patterns P for the high noise, always uniform regime, is given by the following:

$$P = \frac{1-f}{2[\mathrm{erf}^{-1}(1-2\epsilon)]^2} \frac{C_F \frac{M}{N}}{1 + C_F \frac{M}{N} + \frac{\Delta_{UH}^2}{C_F f^2 (1-f)\beta^2}} N, \tag{4.6}$$

where

$$\Delta_{UH} = \beta C_R \alpha - 1 > 0.$$

The parameter Δ_{UH} should be positive in order for the network to have two stable attractor states. However, increasing Δ_{UH} decreases the number of classifiable inputs P. In the following analysis, we assume that the parameters of the recurrent dynamics are adjusted in such a way that Δ_{UH} is not too close to zero, so that there is no risk of losing the two attractors, but also not too large, so as to not sacrifice too much of the classification performance. A reasonable choice is $\Delta_{UH} = 0.2$.

As Δ_{UH} cannot be too small, and $C_F f^2(1-f)\beta^2 \ll 1$ by the high-noise assumption (Eq. 3.43), it can be seen from Equation 4.6 that the proposed network, when operating in the uniform regime, has a worse performance than a committee machine (Eqs. 4.4 and 4.5) and a fully connected readout (Eq. 4.2). However, in the limit of large $C_F M/N$, the performance becomes the same.

Importantly, if the number of input units N and the number of intermediate readouts M are increased in the same proportion, the number of classifiable inputs scales linearly with M or N (Fig. 5*a*), as in the committee machine case. However, now there is not a single neuron that is required to have a connectivity that scales with N, so the connectivity of each neuron can remain finite even when N and M become arbitrarily large. This supports the claim made in Figure 3 about scaling properties of the proposed network. As we will see below, this is not the only regime in which these scaling properties are valid.

The rate at which P grows with the number of input neurons N, P/N, depends on the expansion ratio M/N (the number of intermediate readouts per input neuron), the coding level f, and the parameters of the recurrent dynamics Δ_{UH} and β. Instead of using M/N as an independent parameter, it is more convenient to express P as a function of the average number of efferent connections per input neuron, as follows:

$$c = \frac{MC_F}{N}.$$

Indeed, C_F/N is the probability that an input neuron is connected to a readout. When multiplied by M, it gives the average number of connections that depart from an input neuron and arrive at the intermediate readouts. We assume that, along with C_F, the number of efferent connections per input neuron should also be minimized, given that these connections could be long range.

The dependence of the growth rate of the capacity P/N on the input coding level f is shown in Figure 5*d* for different values of c. To make these plots, we assumed that the parameters of the recurrent dynamics are chosen anew for every value of f so as to keep $\Delta_{UH} = 0.2$ and to satisfy the high-noise condition (Eq. 3.42) by the same margin. We also assumed that the number of feedforward connections per perceptron is $C_F = 50$, which is consistent with the observations in the mouse hippocampus (see Discussion).

The classification performance of the recurrent readout P/N in the high-noise regime increases with c, for any value of f. In other words, when N and C_F are kept constant, increasing the number of perceptrons M, and hence the total number of connections $C_F M$, will always increase the capacity P. However, the capacity cannot increase indefinitely in this way because of the correlations between the perceptrons that we discussed above. Interestingly, the saturation of the capacity as a function of M (or c) is reached sooner for denser representations.

As can be seen from Figure 5*d*, the classification performance reaches its maximum at $f_{\max} \approx 0.05$, depending on the value of c. When c increases, the maximum moves toward sparser f. The position of the maximum also depends on the number of feedforward afferent connections per perceptron, C_F, as $f_{\max} \propto 1/C_F$.

The uniform, low-noise regime. When the noise is low compared with both the recurrent and the feedforward input, the density of the input representations starts playing a crucial role in determining whether the network is in a uniform or a nonuni-

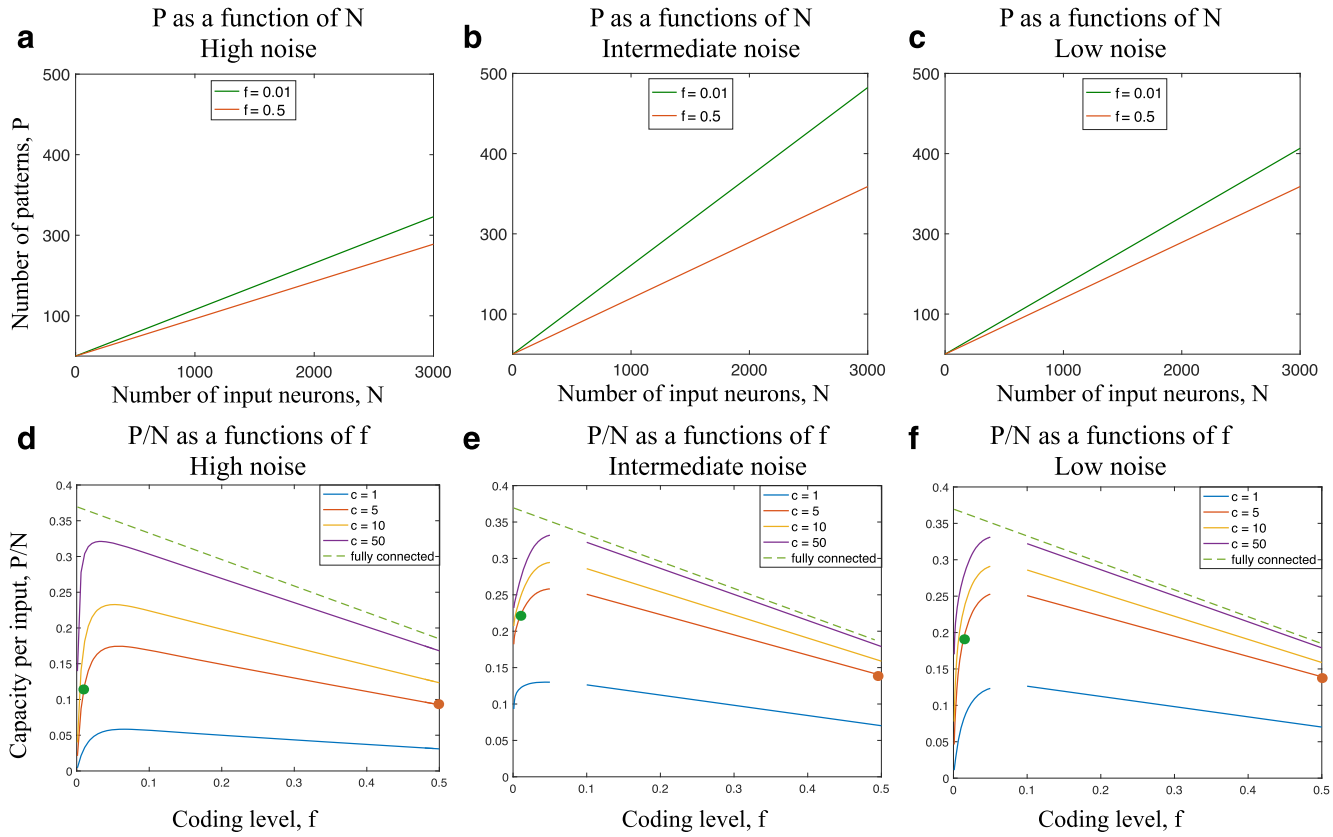

Figure 5. ***a*-*c***, The linear dependence of the classification capacity of the recurrent readout P on the number of input neurons N, when the number of intermediate perceptrons M is increased proportionally to N, so that $c = \frac{C_F M}{N}$ remains constant (we assumed $c = 5$). The red and green lines correspond to dense ($f = 0.5$) and sparse ($f = 0.01$) representations. The number of feedforward connections per perceptron is $C_F = 50$, and the tolerated error rate is $\epsilon = 0.05$. ***a***, High-noise regime: the noise is large compared with the feedforward input. For the dense case (red line), $\beta = 0.04$, and for the sparse case (green line) $\beta = 0.9$; these choices correspond to a ratio of the noise to feedforward input equal to 10. ***b***, Intermediate level of noise: the noise is low compared with the feedforward input, but large when compared with the input from the input receiving to the free neurons in the case of sparse input representations (two-subnetwork regime). The red line corresponds to dense input representations (uniform low-noise regime), and the green line corresponds to the two-subnetwork intermediate-noise regime. ***c***, Low level of noise. The red line corresponds to the uniform low-noise regime, and the green line corresponds to the two-subnetwork low-noise regime (same as majority vote). ***d*–*f***, Change of the slope of the plots from ***a*** to ***c***, P/N with the coding level f for different values of c. ***d***, High-noise regime. Different curves correspond to different numbers of perceptrons M per input neuron, expressed as $c = \frac{C_F M}{N}$. The noise parameter β and the strength of the recurrent synapses α are varied with the coding level f to keep the value of $\Delta_{UH} = 0.2$ and the inequality of Equation 3.42 satisfied by the factor of 10 for every value of f. The last condition implies that the ratio of the noise to the amplitude of the feedforward input is equal to 10 for every point on the curve. ***e***, Intermediate level of noise. The low-f segments of the curves represent the two-subnetwork intermediate-noise regime. Either the noise parameter β or the strength of the recurrent synapses α is varied with f to keep $\Delta_{IJ} = 0.2$. The high-f segments correspond to the uniform low-noise regime, and α is varied with f so that $\Delta_{UL} = 0.2$. ***f***, Low noise. Low-f segments of the curves correspond to the two-subnetwork low-noise regime (same as majority vote), the high-f segments are the same as in panel ***e***. The dashed green line shows the performance of the fully connected readout for comparison. The green and red points on the $c = 5$ curve correspond to the values of f used in ***a*–*c***. The curves on ***e*** and ***f*** are discontinuous because there is no consistent way to analyze the recurrent dynamics in the perceptron layer across the entire range of f for these levels of noise. However, we believe that the capacity changes smoothly across the unexplored region, achieving its maximum at approximately $f \approx 0.05$ for $C_F = 50$.

form regime. If the input representation is dense, $C_F f \gg 1$, the network is again in a uniform regime. As before, all of the neurons have the same average activity, but the main source of inhomogeneity is the feedforward input rather than the noise. The number of classifiable inputs in this uniform low-noise regime is as follows:

$$P = \frac{1-f}{[\mathrm{erf}^{-1}(1-2\epsilon)]^2 \pi} \frac{C_F \frac{M}{N}}{1 + C_F \frac{2}{\pi} \frac{M}{N} + \Delta_{UL}^2} N, \qquad (4.7)$$

with

$$\Delta_{UL} = \sqrt{\frac{2}{\pi}} \frac{C_R \alpha}{\sqrt{C_F f^2 (1-f)}} - 1 > 0.$$

Again, $\Delta_{UL} = 0$ corresponds to the transition from the network with two stable states, which is suitable for classification, to the network where only one state is stable. In this case, it is not the recurrent noise that may destroy the two attractor states, but the variance in the feedforward input (this is why β does not enter into the expression for Δ_{UL}). We assume that the strength of the recurrent connections α is adjusted to keep Δ_{UL} from being too close to zero.

The result (Eq. 4.7) is very similar to the case of dense representations in the high-noise regime. One obvious difference is that the inverse temperature parameter β does not appear since we assumed to be in the low-noise limit $\beta \to \infty$. As before, the capacity per input neuron P/N grows as the expansion ratio M/N or the number of feedforward connections per perceptron C_F increases.

When compared with the performance of the fully connected readout (Eq. 4.2), this regime implies a slightly lower

classification capacity. The difference disappears when c is assumed to be large.

Figure 5, *c* and *d*, summarizes the dependence of the classification performance P/N on the coding level f when the noise in the recurrent network is low on the scale of the feedforward input. The high-f segments of the curves correspond to the uniform low-noise regime.

Nonuniform regimes

When the noise is small compared with the feedforward input and the representations are sparse, the uniform approximation is not valid and the recurrent network behaves in a qualitatively different way; for each input pattern, there would be two distinct populations of neurons: the free neurons, which receive zero feedforward input, and hence are not constrained (free) by the input; and all the others, the input receivers. The two populations would be different for different inputs, they would have different activity distributions, and they would evolve in time differently, although they constantly interact with each other.

Generally, such a regime is intractable with the mean field method, so we need to make the additional assumption that the feedforward synapses are sufficiently strong relative to the recurrent ones, so that the nonzero feedforward inputs are typically larger than the total recurrent inputs in the initial state (before the network reaches the final state, when most of the neurons have the same activity). Furthermore, we need to assume that these feedforward inputs are also much larger than the noise. Under these assumptions, the state of the input receivers is determined by the feedforward input, at least in the initial stages of the dynamics, while the network is deciding which stable state to choose. We then need only to consider the dynamics of the subnetwork of free units, treating the recurrent input from the input receivers as a fixed external input. It is this input that contains the information about the correct classification.

We refer to the described scenario as to the two-subnetwork regime. The classification capacity in the two-subnetworks scenario also depends on the noise. The noise has to be small compared with the feedforward input, otherwise, it might modify the input-receiving neurons. However, it can be either small or large when compared with the amplitude of the recurrent input coming from the input receivers. The noise amplitude determines whether the network operates in a two-subnetworks low-noise regime or in the two-subnetwork intermediate-noise regime.

The two-subnetwork intermediate-noise regime is realized when the representations are sparse ($C_F f \lesssim 1$) and the noise is small relative to the feedforward input but large in the subnetwork of free neurons, namely relative to the input into free neurons from the input receivers. This regime leads to the classification capacity of the following:

$$P = \frac{\langle\sqrt{n}\rangle^2}{f}\frac{1-f}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2}\frac{M/N}{\gamma + \frac{M}{N}\varphi_{C_F,f}}N. \qquad (4.8)$$

Here $\langle\sqrt{n}\rangle$ is the mean of $\sqrt{n}$ over the binomial distribution $\mathbf{B}(C_F,f)$, which is approximated by different functions of C_F and f depending on whether $C_F f$ is small or large (Eqs. 3.15 or 3.16). The term $\varphi_{C_F,f}$ corresponds to the correlations between input receivers (Eq. 3.25). It is of the order of C_F and depends on the coding level f. It is also approximated differently depending on the value of $C_F f$ (see Eqs. 3.26 and 3.28). The quantity γ is given by the following:

$$\gamma = 1 - e^{-C_F f}\left(1 - \frac{\Delta_{TI}^2}{(\Delta_{TI}+1)^2}\right), \qquad (4.9)$$

where

$$\Delta_{TI} = e^{-C_F f}\beta C_R \alpha - 1 > 0$$

As in the uniform regimes, the capacity is maximal (smallest γ) when the network of free units is close to transitioning from three fixed points (Fig. 2*c*,*d*) to one fixed point (Fig. 2*a*,*b*).

In the ultrasparse approximation $C_F f \ll 1$, the expression for P becomes the following:

$$P = \frac{1-f}{\pi[\mathrm{erf}^{-1}(1-2\epsilon)]^2}\frac{\frac{M}{N}C_F^2 f}{\gamma + \frac{M}{N}C_F^2 f}N. \qquad (4.10)$$

Figure 5*b* shows the linear dependence of P on the number of input neurons N for different expansion ratios in the two-subnetwork intermediate-noise regime. As for uniform regimes, the dependence is linear, confirming once more the scaling properties of Figure 3. The relation between the slope of this linear dependence, P/N, and the coding level for $C_F = 50$ and different values of $c = C_F M/N$ is represented by the low-f segments of the curves on Figure 5*e*. The high-f segments of the curves correspond to the low-noise approximation of the uniform regime, which is characterized by the same relationship between the noise and the typical values of the feedforward input. We do not plot the low-f curves beyond $f = 0.05$ because for denser representations the fraction of the free units becomes small (~8%), and it becomes difficult for the randomly and sparsely connected final readout to distinguish between the two states of the free subnetwork. However, we believe that the classification capacity changes smoothly between the two regimes, achieving its maximum for the coding value close to $f = 0.05$. The location of the maximum will change when different values of C_F are assumed, $f_{\max} \propto \frac{1}{C_F}$ for large C_F.

The capacity in the intermediate-noise regime can be larger than the capacity of a majority vote committee machine. Interestingly, the capacity in the two-subnetwork intermediate-noise regime (Eq. 4.8) is larger than in the case of a majority vote committee machine for the same coding level f (Eq. 4.3). This result is counterintuitive, but it can be explained, as follows: in the majority vote scenario, both the input-receiving units and the free units contribute to a collective decision, even though the free units carry no information about the class of the input pattern and they actually generate noise as we assume that initially they are in a random state. In contrast, in the recurrent case, the collective state of the network is initially determined mostly by the input-receiving units, which then drive the free units to the right state. The noise contained in the initial state of the free units does not much affect the initial relaxation dynamics, provided that the noise in dynamics is sufficiently large (relatively low β).

In the case of the majority vote committee machine, the class is decided in only one time step and the initially random free units generate a certain amount of noise that depends on their number. In the case of the recurrent dynamics, the connectivity is sparse and each neuron that participates in it samples the noisy neurons a number of times, which depends on the relaxation time. If these neurons can flip randomly at every time step, then their noise is

Optimizing the network architecture

Optimizing the architecture under the constraint that the total number of long-range connections is constant

The expressions for the classification capacity as a function of the parameters of the network (Eqs. 4.6 to 4.12) allow us to determine the optimal network architecture under different constraints. More specifically, we determine the optimal relation among the number of the input neurons N, the size of the perceptron layer M, and the feedforward connectivity C_F. Note that this notion of optimality is independent from tuning the parameters of the recurrent dynamics, such as α, β, and C_R, which can be chosen later to ensure optimal values of Δ_{UH}, Δ_{UL}, or Δ_{TI}.

We first discuss the optimization under the constraint on the total number of long-range connections (i.e., the feedforward connections). More specifically, we assume that the number of inputs N and the total number of long-range connections C_{FM} are fixed (which implies constant $c = C_F M/N$), and we ask what value of C_F (or M) will optimize the capacity P.

For dense input representations in the uniform regimes (high or low noise), rearranging the connections while keeping their total number the same has no effect on the classification capacity. This can be seen from Equations 4.6 and 4.7. Although the parameter C_F enters Equation 4.6 not only in combination with $C_F M$, for dense representations, $C_F f^2(1-f)\beta^2$ in the last term in the denominator represents the ratio between the noise and the typical value of the feedforward input, which we assume to be constant (see Eq. 3.42).

For the case of sparse representations, independent of the level of noise, the situation is different. In Equations 4.8 and 4.11, the parameter C_F enters through the quantities $\langle\sqrt{n}\rangle$ and $\varphi_{C_F f}$, while in Equation 4.6 it enters explicitly in the last term of the denominator and in the scaling of the noise parameter $\beta \propto \frac{1}{\sqrt{f(1-f)}\langle\sqrt{n}\rangle_{n\neq 0}}$ (see Eq. 3.42). It turns out that for constant C_{FM}, when C_F is assumed to be large (which we always do for this study), the capacity P depends only on the product $C_F f$, the average number of active inputs per perceptron, but not on C_F or f individually [apart from the factor $(1-f)$, which is close to 1 for sparse representations]. The dependence of the capacity on the value $C_F f$ will be represented by the same curves as the low-f regions of the curves shown in Figure 5*d–f*. As we can see from these plots, the capacity P increases as a function of $C_F f$.

So, for sparse representations under low or intermediate noise the optimum of the classification capacity for a fixed number of input neurons N and a fixed total number of feedforward connections $C_F M$ is achieved for the value of C_F, which corresponds to the boundary of applicability of the two-subnetwork regime, $C_F f \approx 2$. For high levels of noise, the optimum is also approximately $C_F f = 2$ (its exact position depends on the value of c and the chosen level of noise relative to the feedforward input).

Increasing C_F under these assumptions is equivalent to increasing the coding level f.

Optimizing the architecture under the constraint that the total number of neurons is constant

We now determine the optimal architecture in the case when the total number of neurons is fixed. Basically, we ask how to partition the total set of neurons between the input and the perceptron layer to maximize the classification capacity. This question is sensible if the dimensionality expansion in the input layer is not a limiting factor, as we assume that all N input neurons are independent.

It is straightforward to derive the optimal expansion ratio M/N from Equations 4.6 to 4.12 under the constraint $M + N =$ constant.

In the uniform regime, unless the noise is very high or the representations are very sparse (see Eqs. 4.6 and 4.7), the expansion ratio that maximizes the capacity can be approximated by the following:

$$\frac{M}{N} \approx \frac{1}{\sqrt{C_F}}$$

The number of perceptrons M is much smaller than the number of inputs N (converging architecture) and the number of feedforward connections per input neuron $c \approx \sqrt{C_F}$.

In the uniform, high-noise regime, for very sparse representations or very high levels of noise, $C_F f^2(1-f)\beta^2 \ll \Delta_{UH}^2$, the optimal expansion ratio is given by the following:

$$\frac{M}{N} \approx \frac{1}{\sqrt{C_F}} \frac{\Delta_{UH}}{\sqrt{C_F f^2(1-f)\beta^2}},$$

which implies a higher proportion of the perceptrons $M/(M+N)$ compared with the previous result.

For the two-subnetwork regime with intermediate noise (Eq. 4.8), the result for the optimal expansion ratio is the same unless the input representations are extremely sparse, in which case the optimal proportion of the perceptrons increases.

The optimal expansion ratio for the low noise is identical to the sparse limit of the intermediate-noise case, as follows:

$$\frac{M}{N} \approx \frac{1}{\sqrt{C_F}\sqrt{C_F f}},$$

To summarize, our model predicts that under the constraint on the total number of neurons $M + N$, the optimal expansion ratio is given by $M/N \approx 1/\sqrt{C_F}$ ($c = \sqrt{C_F}$), unless the input representations are very sparse or the noise is very large, in which case the optimal proportion of the perceptrons increases. For the intermediate levels of noise, this increase is less profound (happens for more sparse representations).

Multinomial classification

We now turn to a more difficult problem of classifying the inputs into more than two categories. The scheme presented above can be generalized in a straightforward way to serve as a multinomial classifier. We first present the generalization of the model where instead of a single population of perceptrons in the intermediate layer, we assume multiple subpopulations, each of which is selective to its own class of input patterns. The model requires that the recurrent connectivity in the intermediate layer is restricted to pairs of neurons belonging to the same subpopulation. This scheme implies that the patterns of activity in the intermediate layer, as well as its connectivity structure, have a specific design, which should be imposed to the network and most likely is driven by top–down signals. The architecture that we will describe is probably the result of a learning process, although here we just focus on the already structured network and we do not model explicitly the synaptic modifications that lead to it.

We later discuss a more realistic scenario that supports multinomial classification. Namely, we assume that the desired activity pattern of the intermediate layer in response to each class of the input pattern is chosen randomly. In this case, the only role of the external supervisor is to guarantee that the activity pattern of

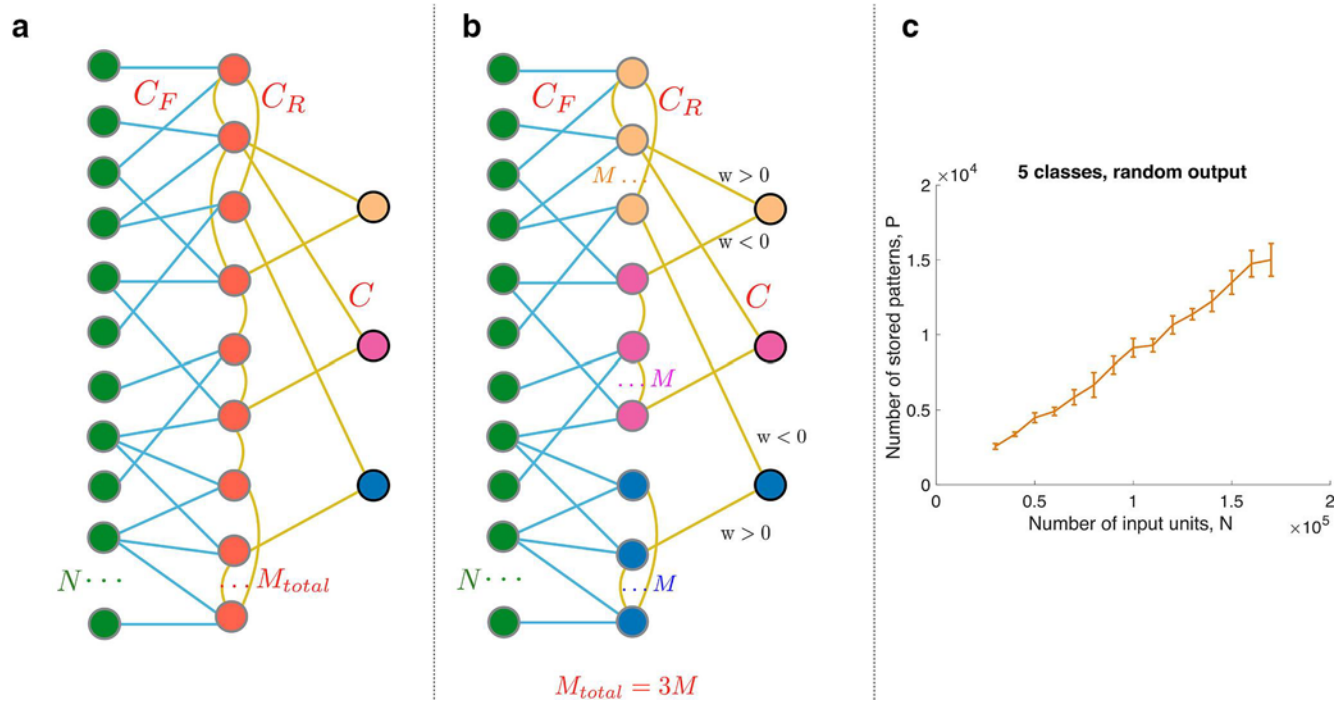

Figure 7. ***a***, Network architecture for the case of structured output (see subsection Structured output). For the case of three-way classification, the intermediate layer of readout neurons is divided into three subpopulations, each selective for its own class of input patterns. The recurrent connectivity is random and excitatory within subpopulations, but there are no recurrent connections between the subpopulations. The final readouts, one for each class, are connected sparsely and randomly, as before, but the sign of the connections is only positive if the presynaptic neuron belongs to the correct subpopulation; the rest are zero or negative. ***b***, Network architecture for the case of random output (subsection Random output). There are no distinct subpopulations in the intermediate layer, and the desired output pattern corresponding to each class of input pattern is chosen randomly. The recurrent connections exist between any pair of readout neurons with equal probability. The strength of these connections, however, is now adjusted according to a Hebbian learning rule (Eq. 4.13). ***c***, The results of the simulation for multinomial classification. The output patterns corresponding to $L = 5$ classes are chosen randomly with the coding level $y = 1/2$. The recurrent connectivity is sparse, and the strength of the synapses are trained with the learning rule (Eq. 4.13). The network of recurrently connected perceptrons is in the high-noise regime with dense input representations ($C_F = 50$, $f = 0.2$, $C_R = 200$, $\alpha = 0.015$, $\beta = 0.5$). The error bars correspond to standard deviations of the capacity over 10 random realizations of the input patterns and network connectivity.

the perceptron layer is the same during the presentation of different input patterns belonging to the same class. In this case, the recurrent connectivity within the perceptron layer is random and sparse, as for binary classification, but unlike the latter, the strength of the existing connections are plastic and are modified by the desired activity patterns via Hebbian plasticity.

This last, more realistic scenario cannot be considered analytically, but we show with simulations that the capacity decrease compared with binary classification is moderate and, most importantly, that the linear scaling with the network size is preserved.

Structured output

The immediate generalization of the recurrent readout scheme to multinomial classification task is to introduce several nonoverlapping populations of intermediate readout neurons, each of which would activate in response to a single class of input stimuli. The recurrent connectivity within a population would be as described before, while no recurrent connections would exist among the neurons belonging to distinct populations. The desired output pattern in response to an input from each class is then structured so that the population corresponding to the given class is active while the others are inactive. The final readout has to contain multiple readout units, one for each class. Their connectivity can still be sparse and random, but the sign of the connections would have to be adjusted based on whether it comes from the neuron in the population selective for the same class as the given final readout or not (Fig. 7*a*).

The classification capacity can now be computed in the same way as above, by noticing that each population is now doing a binary classification, selecting for one of L classes. The only difference is that the proportion of "positive" patterns (the output sparseness) is now $y = 1/L$ instead of 1/2. The capacity formula for the case of sparse output is derived in Materials and Methods, and it differs from the capacity for a dense case by a factor that depends on y, as follows:

$$P_y(N, M) = \frac{1}{4y(1-y)} P_{0.5}(N, M).$$

It should be noted, that the number of intermediate readouts M, entering this formula is the number of units in the population selective for a particular class. So, if the total number of intermediate readout units is M_{total}, and all populations have equal size, it is $M = M_{\text{total}}/L = yM_{\text{total}}$ that should enter the formulas for the capacity. So, in terms of the total number of intermediate units, in the two-subnetwork intermediate-noise regime, for example Equation 4.8, we have the following:

$$P = \frac{L}{4(L-1)} \frac{\langle\sqrt{n}\rangle^2}{f} \frac{1-f}{\pi[\text{erf}^{-1}(1-2\epsilon)]^2} \frac{M_{total}/N}{\gamma + \frac{M_{total}}{LN}\varphi_{C_F,f}} N,$$

where γ is given by Equation 4.9. There are two differences with respect to the binary classification case (Eq. 4.8). The first is the factor $L/4(L-1)$, which is equal to 1/2 for the case of two classes

($L = 2$). This is the reflection of the fact that when only two classes are possible, the current scheme is redundant; when the first population is active, the other is not, and vice versa. In the limit of a large number of classes, the prefactor is equal to 1/4. The other difference is in the second term in the denominator, which rescales N, the number of the input units. Namely, for the number of intermediate readout units, the role of correlations between them is decreased compared with the binary classification case. This is because there is no interference between the readout neurons belonging to different populations.

Random output

Another, more realistic scenario is to assume that each class is represented by a distinct random output pattern. In contrast to the previous scenario, now the output pattern can have a nonzero random overlap. In this case, it is necessary to train the recurrent connections, and we assumed a simple Hebbian learning rule, similar to the one used in the Hopfield model, as follows:

$$J_{kl} = \sum_{a=1}^{L} \zeta_k^a \zeta_l^a, \tag{4.13}$$

where ζ_k^a is the output pattern corresponding to class a ($a = 1 \ldots L$). In this case, there are no structurally distinct subpopulations of the intermediate readout neurons, which are defined a priori (Fig. 7*b*). In contrast, the subpopulations of neurons that represent different classes emerge as a consequence of the learning rule (Eq. 4.13).

Figure 7 shows the simulation results for a five-way classification ($L = 5$) of dense input patterns with high dynamic noise (the parameter values are given in the caption of Fig. 7). As expected, also in this case P increases linearly with N, even when the number of incoming connections is kept constant for all neurons (i.e., it does not scale with N). This result also confirms the validity of our approach in a more realistic case for which it is significantly more difficult to perform analytical calculations.

Notice that in this case, although we did not perform the analysis, we know from previous studies on recurrent neural networks (Amit, 1992) that the number of recurrent connections will have to scale linearly with the total number of classes L. This would explain why the number of recurrent connections could be much larger than the number of feedforward inputs. A future study will address this specific issue.

The initial condition of the recurrent network

An important assumption that we made to implement the majority vote with a recurrent readout is that the recurrent network initial condition is unbiased or, in other words, that $m_0 = 0$ (see subsection Mean field analysis of the recurrent dynamics). This condition might sound difficult to realize in a network that is basically designed to amplify any small deviation from $m_0 = 0$. However, this condition could be realized as follows: assume that before a pattern to be classified is presented, the input layer is spontaneously active. This spontaneous activity generates a feedforward input h_k^{sp}, which causes the disordered state ($m = 0$) to be the only stable state of the recurrent network (Fig. 2*a*). There are two conditions on the statistics of h_k^{sp} that are required to have $m = 0$ as the only stable state of the system in the mean field approximation. The first requirement is that h_k^{sp} has zero expectation value, which is satisfied if the patterns of spontaneous activity are not correlated with the training patterns. The second requirement is that the standard deviation of the distribution is large enough to make the slope of the sigmoidal curve of Figure 2 less than 1. For instance, in the uniform regime (see subsection The uniform regime), the latter requirement is $\sigma_h^{sp} > \sqrt{\frac{2}{\pi}} C_R \alpha$ (which is the opposite of Eq. 3.48), where σ_h^{sp} is the standard deviation of the feedforward current due to spontaneous activity. When the input pattern is presented, then the noise is assumed to decrease to restore the conditions (Eq. 3.48) that allow the recurrent network to have three solutions, two stable, corresponding to the possible classification outcomes, and one unstable, which was the initial state. A reduction in noise during stimulus presentation has been observed in the study by Churchland et al., 2010.

Biological interpretation and testable predictions

One of the important results of our analysis is that the sparseness of the neural activity can play an important role, even in the case in which the connectivity is very limited (see also the Introduction). We showed in Figure 5 that the optimal coding level f (i.e., the average fraction of active neurons) is always approximately $f = 0.05$, when the number of feedforward afferent connections per neuron C_F is assumed to be 50, which would be estimated as the average number of synaptic inputs that each neuron in CA3 receives from the DG. This is a surprising result given that the assumed connectivity is so limited, and it explains why the neural activity in the DG can be so sparse without compromising the computational performance of the hippocampal system.

Intuitively, the optimal sparseness can be explained as follows. One of the reasons why the classification capacity is limited is the noise in the synaptic strengths introduced by the other patterns that have been learned by the classifier. This noise increases as the representations become denser (f increases), which leads to a decrease of the classification capacity for dense representations. However, when the coding level is too small, the fraction of intermediate readouts whose inputs are all silent, and hence not informative, becomes larger and the capacity again decreases.

The optimal coding level also has an interesting dependence on the number of patterns P that should be classified, and this dependence generates a specific prediction on how the richness of the environment can change the sparseness of the representations in the DG.

We assume that the number of input neurons N, which in the application to the mammalian hippocampus corresponds to the number of dentate gyrus granular cells, cannot change substantially. Although adult neurogenesis was observed in this area, the number of adult-born cells is negligible compared with the total number of neurons. It is also reasonable to assume that having higher f (i.e., higher activity in the DG) has a metabolic cost.

When the animal is put into an enriched environment and has to learn to classify more input patterns P than before while keeping N fixed, the classification performance of the network, expressed as P/N should increase. This can be achieved either by increasing the number of neurons in the perceptron layer (the CA3 area in the application to the hippocampus), by increasing the number of feedforward (DG–CA3) connections per perceptron (CA3 pyramidal cell) C_F, or, finally, by changing the coding level f of the input (DG) representations.

In line with the main assumption of our work, the number of connections per perceptron C_F is limited by spatial constraints and cannot be increased further. Increasing the number of cells is also unlikely because the number of required additional neurons would scale as N, and this would require additional wiring and more energy. There is a more efficient alternative, which is to adjust the coding level f.

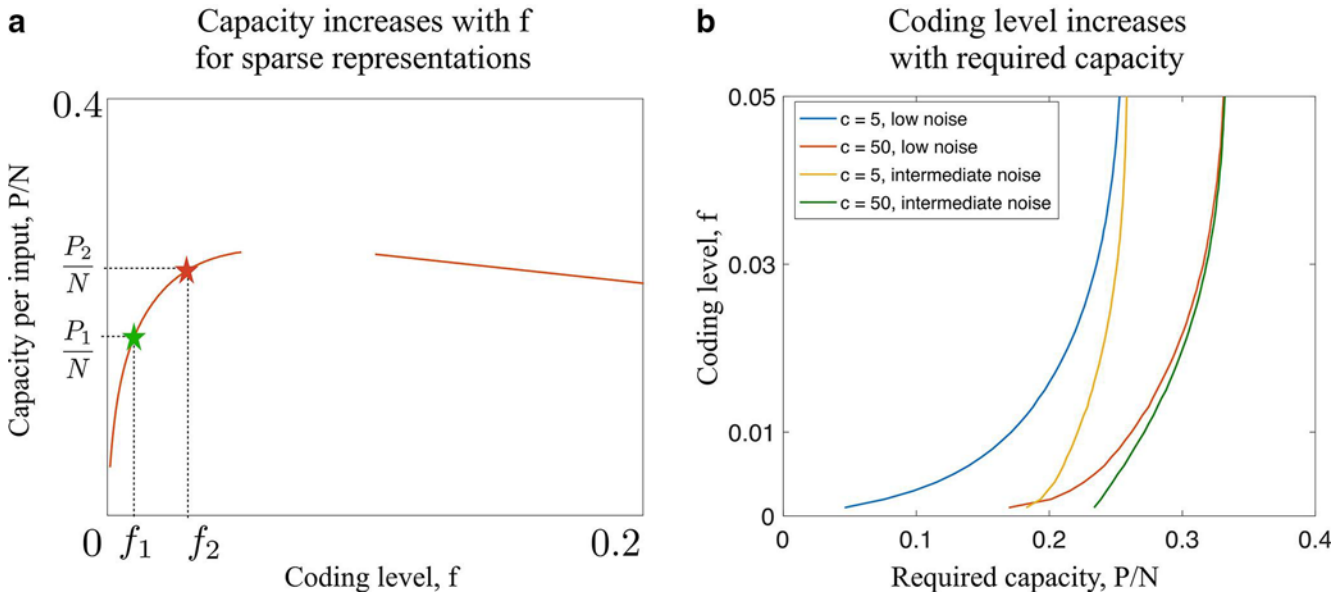

Figure 8. ***a***, Schematic plot demonstrating the increase of the classification capacity with the coding level for sparse input representations. When the initial demand on the number of patterns whose classification the animal has to remember is P_1, the lower level of activity in the dentate gyrus f_1 is sufficient. When the required number of patterns is increased to P_2 (e.g., the environment of the animal is enriched) and neither the connectivity of the network nor its size change, the new classification demand can be met by increasing the coding level up to the value f_2. ***b***, The quantitative predication of the coding level f as a function of the required capacity P/N for two values of the number of feedforward connections per input neuron ($c = 5$ and $c = 50$) in the intermediate- and low-noise regimes.

As neuronal firing has a metabolic cost associated with it, we hypothesize, that f is kept as low as possible so that the demand on the number of patterns P saturates the classification capacity of the network. In other words, if the required number of classifiable patterns is P_1, the coding level f_1 is such that the point $(P_1/N, f_1)$ lies on the curve describing the maximal classification capacity of the network (Fig. 8*a*). There can be two values of f that correspond to the same P_1, of which the network prefers the lower one (on the left from the maximum). When the environment is enriched and the animal needs to classify $P_2 > P_1$ patterns, the coding level should increase to $f_2 > f_1$, so that the point $(P_1/N, f_1)$ is still on the same curve as shown in Figure 8*a* (we assume that $c = \frac{MC_F}{N}$ cannot change).

In Figure 8*b*, we present the predicted relation between the coding level f and the number of patterns P that the animal has to learn to classify correctly (richness of environment), as estimated by our model. Different colors correspond to different values of the feedforward connections per input neuron c (number of DG–CA3 connections per DG granular cell) and the different regimes described above. It should be pointed out that the parameter c is difficult to measure in an experiment, as the number of perceptrons M does not directly correspond to the number of cells in the CA3 area, but rather to the number of cells in the subpopulation whose target activity is assumed to be uniform (see subsection Multinomial classification). However, while the prediction that the coding level should increase with the richness of the environment does not depend on c, it is true for a wide range of values of c and it is valid for both the intermediate- and low-noise regimes, which are likely to be the only two regimes that are relevant for a computationally efficient biological system. The prediction of the model is that the coding level of neural representations in the DG, which could be measured using calcium imaging, would increase with the richness of the environment. If this is observed, then it would be interesting to determine the role of neurogenesis. The number of newborn neurons in adult animals is probably too small to account for the increase in N that would be needed to deal with an enriched environment. However, the newborn neurons could have a significantly larger coding level f and affect the effective coding level of DG in a more substantial way. A new model with mixed coding levels would have to be studied to produce specific quantitative predictions.

Discussion

We presented a model network based on perceptrons in which all the neurons have limited connectivity, and nevertheless the classification capacity grows unboundedly and linearly with the size of the network. The limitations on the classification capacity of the individual perceptrons that are imposed by the limited connectivity are overcome by reading out multiple perceptrons, as in a committee machine. However, the readout mechanism is different from the one normally used in committee machines as it uses a recurrent attractor dynamics of committee members to generate a final vote. Thanks to the recurrent dynamics, it is then possible to read out a small sample of all the committee members to determine the committee decision. This allows for readouts that have a limited connectivity, even when the size of the network becomes very large. The limitations imposed on the number of connections per neuron make the proposed network less efficient than a single readout with unlimited connectivity when the total number of feedforward connections is considered. However, the decrease in classification performance is modest unless the input representations become extremely sparse. Moreover, and most importantly, the scaling properties of our neural system are the same as those of the readout with unlimited connectivity.

Recent theoretical studies (Barak et al., 2013; Cayco-Gajic et al., 2017; Litwin-Kumar et al., 2017) considered a neural architecture that is similar to the one that we analyzed. In all of these studies, the input neurons are connected to an intermediate layer of neurons and then read out by a single cell with unlimited connectivity. The inputs are completely random in the study by Litwin-Kumar et al. (2017), are low-dimensional and correlated

in the study by Barak et al. (2013), and are highly correlated in the study by Cayco-Gajic et al. (2017). The intermediate layer makes the neural representations linearly separable by expanding the dimensionality, so that the readout, which is linear, can be trained. One of the studies (Litwin-Kumar et al., 2017) also shows that this dimensionality expansion can be efficiently implemented using random connectivity, and, surprisingly, the optimal number of random connections per neuron is very small. In our case, we also discuss a situation in which the connectivity is limited, but our work addresses a different computational problem: in the study by Litwin-Kumar et al. (2017), the authors focused on the problem of dimensionality expansion and showed that large connectivity can actually reduce the performance of a downstream classifier that reads out randomly connected neurons. In their case, which nicely applies to the cerebellum, the classifier had unlimited connectivity. Here we focused on the problem of how to implement this downstream classifier under the constraint of limited connectivity. In our case, the constraint on limited connectivity is imposed because of the metabolic and spatial cost of wiring, whereas in the study by Litwin-Kumar et al. (2017) it emerges as a requirement for optimizing the ability to expand the dimensionality of the neural representations. In the study by Cayco-Gajic et al. (2017), the authors showed that sparse connectivity can be beneficial in a problem of pattern separation, but again it is the connectivity of the neurons in the intermediate layer that they refer to, and not the readout.

In our article and in one of the articles discussed in the previous paragraph (Litwin-Kumar et al., 2017), the input patterns were assumed to be random and uncorrelated. For any analysis of the performance of a neural circuit, we need to make an assumption about the nature of the inputs and random patterns is a standard assumption enabling theorists to perform analytical calculations. Although real-world sensory inputs are likely to be highly structured and correlated, it is not completely unreasonable to believe that, at least in some brain areas like the hippocampus, the patterns of activity can be modeled as random and uncorrelated. Indeed, the hippocampus is known to be involved in memory consolidation. The most efficient way of storing real-world correlated memories is to memorize only their uncorrelated component. In other words, if it is possible to compress the memories by taking advantage of their correlations, the resulting compressed representations will look random and uncorrelated (Fusi, 2017). This is a process that is normally not modeled explicitly, although there are some models predicting that the hippocampus might be involved in this compression process (M.K. Benna and S. Fusi, unpublished observations).

One of the results that we discussed in our article is that there are situations in which the proposed recurrent readout scheme can outperform classical readouts that are based on a majority vote despite the fact that the majority vote would require a significantly larger readout connectivity (see subsections Nonuniform regimes and Two-subnetwork regime, intermediate noise). For the majority vote scheme, the classification capacity drops drastically when the input representations are very sparse because the fraction of classifiers whose inputs are all silent becomes substantial and these classifiers just contribute to the noise. Instead, for the recurrent readout the classification capacity can be kept high even for very sparse representations in certain parameter regimes because the recurrent dynamics can align the "free" classifiers to the majority decided by the other, informative classifiers. The lower limit on the coding level f, below which the capacity drops is determined by the amount of noise in the recurrent dynamics, the expansion ratio, and the number of feedforward connections per perceptron (Fig. 5).

In general, the proposed system is robust to both sparse connectivity and sparse representations, which makes it suitable to describe neural circuits like the DG and CA3 area, where the number of connections of downstream neurons (CA3) is much smaller than the number of neurons in the input (DG) and the neural activity in the input can be very sparse. CA3 is known to have the recurrent connections that would implement our proposed readout mechanism. We showed that for intermediate noise levels (see Results, sections Nonuniform regimes and Two-subnetwork regime, intermediate noise), the classification capacity stays within a reasonable range even when the expected number of active units read out by each perceptron is smaller 1 (Fig. 5*e*,*f*). This result nicely complements the findings of the studies by Barak et al. (2013), Litwin-Kumar et al. (2017), and Cayco-Gajic et al. (2017), where the authors show that low-dimensional correlated inputs require an intermediate layer of neurons (randomly connected in the study by Barak et al., 2013; randomly connected or learned in the studies by Cayco-Gajic et al., 2017; Litwin-Kumar et al., 2017). For these neurons in the intermediate layer, there is an optimal sparseness level, which minimizes the generalization error of a single perceptron-like readout. In the study by Cayco-Gajic et al. (2017), the authors also showed that sparse representations in the intermediate layer are important for performing pattern discrimination. Here we showed that there is a readout scheme that would also work for the sparse representations required in the intermediate layer in the studies by Barak et al. (2013), Cayco-Gajic et al. (2017), and Litwin-Kumar et al. (2017), and does not require an unreasonably large number of long-range connections.

Our model is intentionally abstract with binary neurons, and no separation between excitation and inhibition. However, the dynamics of the recurrent network is very similar to the attractor dynamics so widely studied first in abstract models like the model of Hopfield (1982) and then in more realistic models that contain integrate-and-fire neurons with dynamic synapses, sparse representations (Amit and Brunel, 1997; Wang, 1999; Compte et al., 2000; Brunel and Wang, 2001), and also, in some cases, plastic synapses (Amit and Mongillo, 2003). We believe that the path that goes from our abstract model to more realistic models such as those just described is very similar to the one already followed. This will be an interesting future project, which most likely will confirm the scaling properties that we derived and discussed in our article, as it happened in the case of the attractor neural networks based on the pioneering work of John Hopfield.

References

Amaral DG, Ishizuka N, Claiborne B (1990) Neurons, numbers and the hippocampal network. Prog Brain Res 83:1–11. CrossRef Medline

Amit DJ (1992) Modeling brain function: the world of attractor neural networks. Cambridge, UK: Cambridge UP.

Amit DJ, Brunel N (1997) Model of global spontaneous activity and local structured activity during delay periods in the cerebral cortex. Cereb Cortex 7:237–252. CrossRef Medline

Amit DJ, Fusi S (1994) Learning in neural networks with material synapses. Neural Comput 6:957–982. CrossRef

Amit DJ, Mongillo G (2003) Spike-driven synaptic dynamics generating working memory states. Neural Comput 15:565–596. CrossRef Medline

Barak O, Rigotti M, Fusi S (2013) The sparseness of mixed selectivity neurons controls the generalization—discrimination trade-off. J Neurosci 33:3844–3856. CrossRef Medline

Bishop C (2007) Pattern recognition and machine learning. New York: Springer Verlag.

Breiman L (1996a) Bagging predictors. Mach Learn 24:123–140. CrossRef

Breiman L (1996b) Stacked regressions. Mach Learn 24:49–64. CrossRef
Brunel N, Wang XJ (2001) Effects of neuromodulation in a cortical network model of object working memory dominated by recurrent inhibition. J Comput Neurosci 11:63–85. CrossRef Medline
Bullmore E, Sporns O (2012) The economy of brain network organization. Nat Rev Neurosci 13:336–349. CrossRef Medline
Cayco-Gajic NA, Clopath C, Silver RA (2017) Sparse synaptic connectivity is required for decorrelation and pattern separation in feedforward networks. Nat Commun 8:1116. CrossRef Medline
Chawla MK, Guzowski JF, Ramirez-Amaya V, Lipa P, Hoffman KL, Marriott LK, Worley PF, McNaughton BL, Barnes CA (2005) Sparse, environmentally selective expression of arc rna in the upper blade of the rodent fascia dentata by brief spatial experience. Hippocampus 15:579–586. CrossRef Medline
Churchland MM, Yu BM, Cunningham JP, Sugrue LP, Cohen MR, Corrado GS, Newsome WT, Clark AM, Hosseini P, Scott BB, Bradley DC, Smith MA, Kohn A, Movshon JA, Armstrong KM, Moore T, Chang SW, Snyder LH, Lisberger SG, Priebe NJ, et al (2010) Stimulus onset quenches neural variability: a widespread cortical phenomenon. Nat Neurosci 13:369–378. CrossRef Medline
Compte A, Brunel N, Goldman-Rakic PS, Wang XJ (2000) Synaptic mechanisms and network dynamics underlying spatial working memory in a cortical network model. Cereb Cortex 10:910–923. CrossRef Medline
Cover TM (1965) Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. IEEE Trans Electron Comput EC-14:326–334. CrossRef
Drew LJ, Fusi S, Hen R (2013) Adult neurogenesis in the mammalian hippocampus: why the dentate gyrus? Learn Mem 20:710–729. CrossRef Medline
Efron B, Morris C (1973) Combining possibly related estimation problems. J R Stat Soc Series B Stat Methodol 35:379–421.
Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 55:119–139. CrossRef
Freund Y, Schapire RE et al (1996) Experiments with a new boosting algorithm. In: Machine learning: proceedings of the thirteenth international conference (ICML '96): Bari, Italy July 3–6, 1996, Vol 96, pp 148–156. San Francisco, CA: Kaufmann.
Fusi S (2017) Computational models of long term plasticity and memory. Available at arXiv:1706.04946 [q-bio.NC].
Gardner E (1987) Maximum storage capacity in neural networks. Europhys Lett 4:481. CrossRef
Geller M, Ng E (1971) A table of integrals of the error function. II. additions and corrections. J Res Natl Bur Stand 75:149–163.
Green EJ, Strawderman WE (1991) A james-stein type estimator for combining unbiased and possibly biased estimators. J Am Stat Assoc 86:1001–1006. CrossRef
Hopfield JJ (1982) Neural networks and physical systems with emergent collective computational abilities. Proc Natl Acad Sci U S A 79:2554–2558. CrossRef Medline
Jung MW, McNaughton BL (1993) Spatial selectivity of unit activity in the hippocampal granular layer. Hippocampus 3:165–182. CrossRef Medline
Kushnir L, Fusi S (2017) Classifiers with limited connectivity. bioRxiv 157289. CrossRef
Kwon C, Oh J (1997) Storage capacities of committee machines with overlapping and non-overlapping receptive fields. J Phys A Math Gen 30:6273. CrossRef
Litwin-Kumar A, Harris KD, Axel R, Sompolinsky H, Abbott LF (2017) Optimal degrees of synaptic connectivity. Neuron 93:1153–1164.e7. CrossRef Medline
Mitchison G, Durbin R (1989) Bounds on the learning capacity of some multi-layer networks. Biol Cybern 60:345–365.
Monasson R, Zecchina R (1995) Weight space structure and internal representations: a direct approach to learning and generalization in multilayer neural networks. Phys Rev Lett 75:2432–2435. CrossRef Medline
Nilsson NJ (1965) Learning machines: foundations of trainable pattern-classifying systems. New York, NY: McGraw-Hill.
Parmanto B, Munro PW, Doyle HR (1996) Reducing variance of committee prediction with resampling techniques. Connect Sci 8:405–426. CrossRef
Rao JNK, Subrahmaniam K (1971) Combining independent estimators and estimation in linear regression with unequal variances. Biometrics 27:971–990. CrossRef
Rosenblatt F (1957) The perceptron, a perceiving and recognizing automaton Project Para. Buffalo, NY: Cornell Aeronautical Laboratory.
Roudi Y, Latham PE (2007) A balanced memory network. PLoS Comput Biol 3:e141. CrossRef Medline
Rubin DB, Weisberg S (1975) The variance of a linear combination of independent estimators using estimated weights. Biometrika 62:708–709. CrossRef
Tsodyks M, Feigel'Man M (1988) The enhanced storage capacity in neural networks with low activity level. Europhys Lett 6:101–105. CrossRef
Urbanczik R (1997) Storage capacity of the fully-connected committee machine. J Phys A Math Gen 30:L387–L392. CrossRef
Usher M, McClelland JL (2001) The time course of perceptual choice: the leaky, competing accumulator model. Psychol Rev 108:550–592. CrossRef Medline
Wang XJ (1999) Synaptic basis of cortical persistent activity: the importance of NMDA receptors to working memory. J Neurosci 19:9587–9603. CrossRef Medline
Wang XJ (2002) Probabilistic decision making by slow reverberation in cortical circuits. Neuron 36:955–968. CrossRef Medline
Wolpert DH (1992) Stacked generalization. Neural Netw 5:241–259. CrossRef
Zhou Z-H (2012) Ensemble methods: foundations and algorithms. Boca Raton, FL: Taylor & Francis.