Neuron
Article

Sparseness and Expansion in Sensory Representations

Baktash Babadi[1] and Haim Sompolinsky[1,2,*]
[1]Swartz Program in Theoretical Neuroscience, Center for Brain Science, Harvard University, Cambridge, MA 02138, USA
[2]Edmond and Lily Safra Center for Brain Sciences, Hebrew University, Jerusalem 91904, Israel
*Correspondence: haim@fiz.huji.ac.il
http://dx.doi.org/10.1016/j.neuron.2014.07.035

SUMMARY

In several sensory pathways, input stimuli project to sparsely active downstream populations that have more neurons than incoming axons. Here, we address the computational benefits of expansion and sparseness for clustered inputs, where different clusters represent behaviorally distinct stimuli and intracluster variability represents sensory or neuronal noise. Through analytical calculations and numerical simulations, we show that expansion implemented by feed-forward random synaptic weights amplifies variability in the incoming stimuli, and this noise enhancement increases with sparseness of the expanded representation. In addition, the low dimensionality of the input layer generates overlaps between the induced representations of different stimuli, limiting the benefit of expansion. Highly sparse expansive representations obtained through synapses that encode the clustered structure of the input reduce both intrastimulus variability and the excess overlaps between stimuli, enhancing the ability of downstream neurons to perform classification and recognition tasks. Implications for olfactory, cerebellar, and visual processing are discussed.

INTRODUCTION

Sensory processing in the brain is implemented through a sequence of representations. Often, transformations from a primary representation to a secondary representation are characterized as expansive, indicating that the number of neurons in the secondary representation is much larger than that in the primary one. This expansion in the dimensionality is often accompanied by a change in the firing activity levels from a dense pattern in the primary area to a sparse representation downstream, in which only a few neurons respond to any given stimulus, and each stimulus activates only a small fraction of the population.

The rodent olfactory bulb projects to the piriform cortex (Mombaerts et al., 1996), which hosts millions of pyramidal neurons, roughly three orders of magnitude more than the number of glomeruli in the bulb. While the response of the neurons in the olfactory bulb to odorant stimuli is quite dense (Vincis et al., 2012), only about 10% of the neurons in the piriform cortex show an evoked response to each odorant (Stettler and Axel, 2009; Poo and Isaacson, 2009). In the fly olfactory system, the antenna lobe consisting of 50 glomeruli projects to the mushroom body containing about 2,500 Kenyon cells. In response to an odorant stimulus, 59% of the projection neurons and only 6% of the Kenyon cells fire (Turner et al., 2008). In cat visual cortex, there is an approximate 25:1 expansion ratio between the number of axons leaving V1 and the axons that enter this area from lateral geniculate nucleus (LGN), but only 5%–10% of V1 neurons respond to any natural scene stimulus (Olshausen and Field, 2004). Similar ratios have been observed in the somatosensory system (Brecht and Sakmann, 2002), the auditory system (DeWeese et al., 2003), and the electrosensory system of electric fish (Chacron et al., 2011).

The ubiquity of this phenomenon suggests that sparse and expansive transformations entail a fundamental computational advantage for sensory processing. Indeed, one of the early brain theories in modern time, the Marr-Albus theory of the cerebellum (Marr, 1969, Albus, 1971), explained expansion in this system by the well-known relation between the maximum number of classifications of generic inputs implementable by a simplified neuron model, the perceptron, and the number of its input afferents (Cover, 1965). These early theories also propose that sparseness of the activity patterns of the cerebellar granule cell layer improves their separability. Models of associative memory in recurrent networks also show that sparseness increases memory capacity (Cortes and Vapnik, 1995; Tsodyks and Feigelman, 1988). However, a careful analysis reveals that, for large random patterns, capacity is improved by sparseness of their class membership but is unaffected by sparseness of the inputs to the classifier (Gardner, 1988; Gütig and Sompolinsky, 2006). Efficient coding theories of sensory processing have explained the emergence of sparse V1 (Gabor-like) representations as reflecting the sparse, statistically independent components of naturalistic images (Olshausen and Field, 1996, Bell and Sejnowski, 1997), and recent compressed sensing research has devised efficient sparse coding algorithms for recovering sparse signals that underwent linear compression (Ganguli and Sompolinsky, 2012, Rozell et al., 2008).

The purpose of this article is to address the computational benefits of expansion and sparseness in generic ensembles of clustered stimuli. We focus on relatively simple and biologically plausible architectures and dynamics. We examine the conditions under which the system can retain adequate functionality

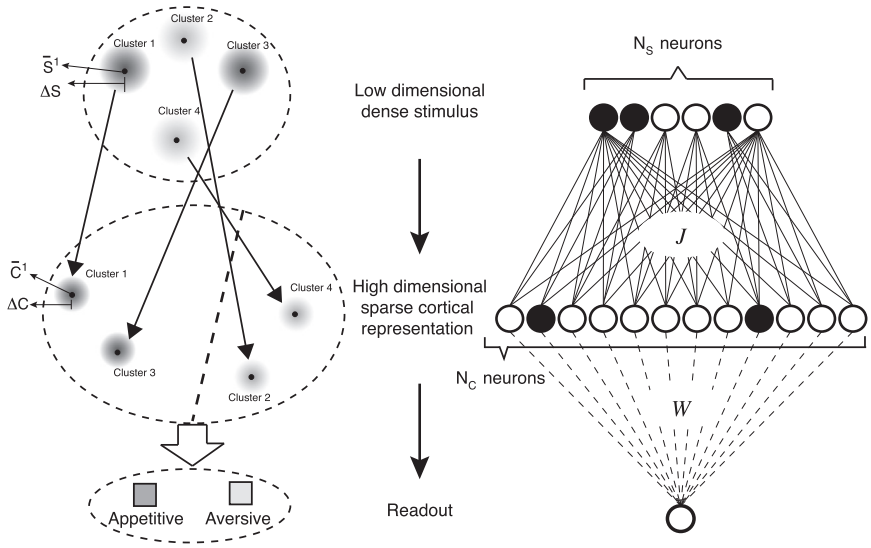

Figure 1. Schematic Description of Sparse and Expansive Representation

Neural activity patterns in the stimulus layer (top) are organized as P clusters of size ΔS, around central patterns $\bar{S}^m$, $m = 1,\ldots,P$. Through feed-forward synaptic projections J, patterns in the stimulus layer are mapped onto patterns in the cortical layer (middle) that, in turn, become organized into clusters of size ΔC, around central patterns $\bar{C}^m$. The size of the stimulus layer is N_S, and the size of cortical layer is N_C. While the representations in the stimulus layer are dense and low dimensional, cortical representations are expansive ($N_C > N_S$) and sparse, i.e., a small fraction f of cortical neurons are active in each pattern. Each cluster of stimulus patterns belongs to a class, e.g., appetitive (dark) or aversive (light) in the olfactory context. A downstream readout neuron (bottom) learns the binary classification of the clusters through the synaptic weights W.

in the presence of substantial variability in the input responses to the same underlying stimulus. This variability may reflect natural variability in the sensory environment and/or noise present in upstream neuronal processing. The required robustness to variability yields surprising results regarding the role of both expansion and sparseness and yields insight into the character of neuronal computation across multiple sensory stages.

RESULTS

Model of Sensory Layers

Our model of early sensory processing is composed of three layers of neurons arranged in a feed-forward structure (Figure 1, right). The first layer (with N_S neurons) is the "stimulus layer," which stands for an early dense and relatively low dimensional neural representation of the sensory stimuli, such as the glomeruli layer in the olfactory bulb and the antenna lobe, the mossy fibers in the cerebellum, or the relay cells in LGN. The second layer (with N_C neurons, $N_C > N_S$) is where the sparse and expansive representation takes place. We call it the "cortical layer," as it often represents sensory cortices or cortical-like structures, such as piriform cortex, V1 area, or the mushroom body. The third layer is the "readout layer," which represents a downstream neural population that receives input from the cortical layer and performs a specific computation, such as recognition of a specific stimulus or classification of stimuli. For concreteness, we will assume a single readout neuron that performs a binary classification of the stimuli. For simplicity, all neurons in the network are binary units, i.e., the activity level of each neuron is either 0 (silent) or 1 (firing).

We further assume that the input patterns are organized as clusters so that the center of each cluster represents a prototypical representation of an underlying stimulus such as a specific odor (Figure 1, left). Other members of the cluster are noisy variants of the central pattern, representing natural variations in the stimulus representation due to changes in stimulus physical features, input noise, or neural noise in afferent stages. For example, in the olfactory system, cluster centers might represent the response of the olfactory bulb to a pure ethologically relevant odorant with a given concentration, while the other members of the cluster are responses to fluctuations in the odor's concentration, contamination by other chemicals, or noise induced by the olfactory receptors. There are P different clusters, and the activity of the i th neuron in the stimulus layer ($i = 1,\ldots,N_S$) corresponding to the center of the m th cluster ($m = 1,\ldots,P$) is denoted by $\bar{S}_i^m$, which are chosen as independent and identically distributed binary patterns, with ½ probability for $\bar{S}_i^m = 1$. Other patterns in the cluster, denoted as S^m, are generated by flipping at random the state of the neurons in $\bar{S}^m$ with a probability that we denote as $\Delta S/2$. Thus, ΔS quantifies the size (or radius) of the clusters. This quantity also equals the average distance of patterns from their corresponding cluster center: $\Delta S = 2\langle \sum_{i=1}^{N_S} \left| S_i^m - \bar{S}_i^m \right| \rangle / N_S$, where the angular brackets denote average over all patterns S^m belonging to the cluster m. This distance is normalized so that random patterns have distance 1 from any cluster center. Thus, $\Delta S = 0$ corresponds to clusters that contain only the central patterns, and $\Delta S = 1$ corresponds to clusters so large that they encompass most of the patterns in the stimulus layer.

Each neuron in the cortical layer receives a weighted sum of the inputs from the stimulus layer, with a synaptic weight matrix J, and compares it against a threshold T (see Experimental Procedures). The cortical representation of the center of the m th input cluster is denoted as $\bar{C}_j^m, j = 1,\ldots,N_C$. The level of sparseness of the cortical layer, i.e., the fraction f of cortical neurons that fire in response to each stimulus, is set by tuning the threshold T (see Experimental Procedures). A number of possible mechanisms, such as feed-forward inhibition (Koulakov and Rinberg, 2011), lateral inhibition (Sachdev et al., 2012), or intrinsic properties of neurons (Demmer and Kloppenburg, 2009) might perform this function.

A key question is what should be the "design principle" for the synaptic matrix J. The simplest scenario is to assume that each synaptic weight J_{ji} is an independent and identically distributed (i.i.d.) random variable. We implement this scenario by choosing

$J_{ji} = \mathcal{N}(0, 2/\sqrt{N_S})$. Later on, we also consider alternative schemes with more structured synapses.

Increased Cluster Sizes in Cortical Representation

Ideally, the transformation of the signals from the stimulus to the cortical layer should enhance the robustness to noise and variations of the input signal in order to facilitate unambiguous processing by downstream structures. Thus, in the cortical layer, distances between different clusters should be large, while distances among patterns belonging to the same cluster should shrink. In fact, with random projections to the cortical layer, distances between clusters are large (≈ 1) in both layers; therefore, we focus on the intracluster distances, or alternatively, on the cluster sizes (schematically shown in Figure 2A). The cortical cluster sizes are defined as:

$\Delta C = \langle \sum_{j=1}^{N_C} \left| C_j^m - \overline{C}_j^m \right| \rangle / (2N_C\ f(1-f))$, where C^m is the cortical representation of S^m. Here again, the normalization is such that a random cortical pattern with sparseness f is of distance 1 from any cluster center. Thus, our measure of the cortical cluster size can be interpreted as the average squared Euclidean distance between cortical patterns and the center of their cluster, relative to the average distance between two cluster centers. Thus, if $\Delta C = 1$, the cluster structure is completely lost in the cortical representation. As we show later, this normalization of the squared Euclidean distances is also justified by the fact that the readout signal-to-noise ratio depends on these distances through the factor $1 - \Delta C$ (Equation 5).

We have evaluated analytically (and confirmed by simulations) the cortical cluster sizes. Notably, we found that $\Delta C > \Delta S$, namely, the transformation from the stimulus layer to the cortical layer, causes an increase in the size of the clusters (Figure 2B). For small stimulus clusters ($\Delta S \ll 1$) and sparse cortical representations ($f \ll 1$), the cortical cluster size can be approximated as follows (see Experimental Procedures):

$$\Delta C \approx \sqrt{\frac{2}{\pi} |\log f| \Delta S}. \qquad \text{(Equation 1)}$$

This implies that, in the regime of small ΔS, $\Delta C \gg \Delta S$ indicating that under random projections, the variability inherent in stimuli belonging to the same cluster is amplified in the cortical layer.

Notably, our results reveal that sparseness of cortical representations increases the size of the cortical clusters, i.e., smaller f leads to larger ΔC (Figures 2B and 2C). In fact, Equation 1 implies that, for high sparseness ($f \ll 1$), even small stimulus clusters, of sizes $\Delta S > 1/|log\ f|$, are dispersed in cortex essentially across the entire space of sparse patterns. Figure 2D provides a schematic explanation of the adverse effect of sparseness. Let us denote by $\overline{h}$ the net inputs to cortical neurons induced by a central pattern $\overline{S}^m$. For large N_S, $\overline{h}$ obeys a Gaussian distribution (Figure 2D, solid curves). The area to the right of the threshold represents the fraction of active neurons in the cortical pattern $\overline{C}^m$, i.e., f (Figure 2D, gray areas). For small ΔS, a typical pattern S^m induces a random change in the input that, in turn, causes some neurons with $\overline{h} \approx T$ to cross the threshold and change their state, yielding a cortical pattern C^m, which is slightly different from $\overline{C}^m$ (Figure 2D, shaded areas). ΔC is roughly the size of the area corresponding to neurons that changed their state from active to inactive, relative to the area of active neurons. As seen in Figure 2D, this relative area is larger in the sparse case (left) than in the dense case (right).

Excess Overlaps between Cortical Clusters

While, on average, the distance between pairs of cortical clusters is 1, there are important deviations in intercluster distances, which vary from one realization of J to another and from one cluster pair to another (schematically shown in Figure 3A). To quantify this effect, we define the overlap between the central patterns of distinct cortical clusters m and n as the normalized dot product between them, i.e., $O^{m,n} = \sum_{j=1}^{N_C} \overline{C}_j^m\ \overline{C}_j^n / N_C$. For $m \neq n$, $O^{m,n} = f^2 + r^{m,n}$, where the first term represents the average overlap between two random sparse vectors and the second term is the deviations of $O^{m,n}$ from this value. As is shown later, overlaps between cluster centers are detrimental for the downstream readout, as they render the distinction of cortical patterns more difficult. When averaged over all realizations of J, $\langle r^{m,n} \rangle = 0$, but for any fixed J, it has a nonzero variance, which can be written as:

$$\left\langle (r^{m,n})^2 \right\rangle = f^2 (1-f)^2 \left(\frac{1}{N_C} + \frac{Q^2}{N_S} \right). \qquad \text{(Equation 2)}$$

Imagine that, instead of generating the cortical patterns via the connections J, they would be chosen randomly (i.e., each $\overline{C}_j^m$ is independently chosen to be 1 with probability f). In this case, the fluctuations in their overlaps $O^{m,n}$ would yield $\langle (r^{m,n})^2 \rangle = f^2 (1-f)^2 / N_C$, which is the same as the first term in the righthand side of Equation 2. Hence, we denote this contribution as the squared amplitude of the random overlap of clusters. The extra term in Equation 2 is the consequence of the feed-forward projections, i.e., the fact that the cortical patterns are generated not randomly but by filtering the random stimulus centers $\overline{S}^m$ and $\overline{S}^n$ through the same projection matrix J. It is important to note that this contribution scales as $1/N_S$, reflecting the square amplitude of the random overlap of the stimulus patterns, with a size-independent prefactor Q^2. We refer to Q as the amplitude of the excess overlap between cortical clusters. The different scaling of the random overlap and the excess overlap (Figure 3B) implies that the excess overlap is particularly important when the stimuli undergo a large expansion, i.e., $N_S \ll N_C$. Figure 3C shows that Q vanishes as $f \to 0$, i.e., sparser representations lead to smaller excess overlaps between clusters in the cortical layer. This is also implied by the analytical expression for Q when $f \ll 1$ (see Experimental Procedures):

$$Q \approx 2f |\log f|. \qquad \text{(Equation 3)}$$

An interesting signature of the excess overlaps between cortical representations of different stimuli is the eigenvalue spectrum of the covariance matrix of the cortical patterns. This can be done by principal-component analysis (PCA) of the matrix $O^{m,n}$. If the cortical patterns are random, the resulting spectrum obeys the well-known Marchenko-Pastur (MP) distribution. Because the individual excess overlaps are small (of order $1/\sqrt{N_S}$), the deviation from the MP distribution is pronounced when the dimensionality of the covariance matrix is large, i.e., the number of patterns P is larger than N_S. In this case, the cortical spectrum shows an enhanced power in the first N_S eigenvalues, whereas

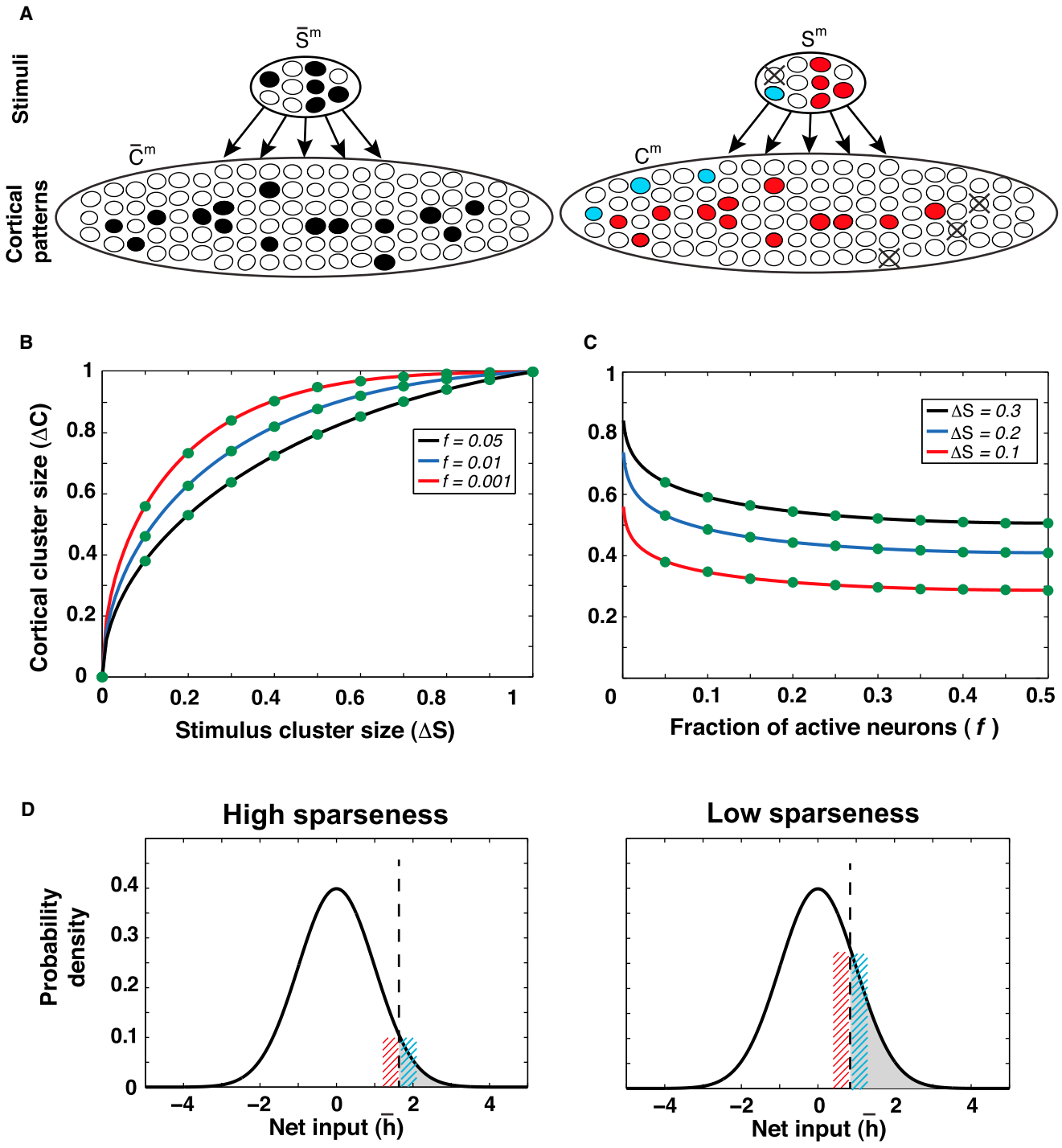

Figure 2. Size of Cortical Clusters in the Case of Random Projections

(A) Schematic description of the cluster sizes. The cluster center stimulus $\overline{S}^m$ induces cortical pattern $\overline{C}^m$ (left). A typical stimulus S^m, which is slightly different from $\overline{S}^m$, induces cortical pattern C^m that is, in turn, slightly different from $\overline{C}^m$ (right). The red neurons (right) are those that are active both in central and typical patterns; the blue neurons are those that are active in the typical but not in the central pattern; and the crossed neurons are those that are active in the central but not in the typical pattern. These variations of the typical patterns compared to the central patterns are quantified by cluster sizes ΔS and ΔC.

(B) Increase of the size of cortical clusters, ΔC, compared to the size of stimulus clusters, ΔS. In this and all subsequent figures, the filled circles are results of numerical simulations, and the solid curves are analytical results. Different curves correspond to different activity levels, f, of the cortical layer. Other parameters of the network are $N_S = 1000$, $N_C = 10000$, and $P = 1000$.

(legend continued on next page)

the rest of the spectrum has a profile similar to that of the MP spectrum (with a slightly smaller amplitude). An example is shown at the top of Figure 3D. It is interesting that, if P is further increased, there is a discontinuity in the spectrum, signaling the existence of a gap between the first N_S eigenvalues and the rest (Figure 3D, middle). The branch of the largest N_S eigenvalues is a stark signature of the approximate low dimensional character of the cortical representations, inherited from its low dimensional stimulus origin. The shape of the spectrum also depends on the level of sparseness. As noted earlier, low values of f also imply smaller Q and, correspondingly, a spectrum that is similar to the MP shape (Figure 3D, bottom).

Classification by a Readout Neuron

With the aforementioned analyses, we are now equipped to address the effect of sparseness and expansion on the performance of a downstream readout neuron. The task of the readout neuron is to classify the clusters of the input patterns into two categories. For example, in the context of the olfactory system, the two categories can be appetitive and aversive odors (Figure 1). Here, we assume that each cluster m is randomly associated with a label L^m, which is either 1 (e.g., appetitive) or −1 (e.g., aversive), each with ½ probability. As shown in Figure 1, we consider a readout neuron that performs a linear classification via a set of synapses W. In order to perform a given classification, these weights must be trained by a supervised learning rule that provides information about the desired categories of inputs. We concentrate on a simple Hebbian learning rule for training W, because of its biological plausibility and its amenability to exact analytical study. Qualitatively similar behavior is found in more elaborate linear classification schemes, i.e., perceptron, pseudoinverse rule, and support vector machine (SVM) (see Supplemental Information available online). In the Hebb scheme, the synaptic weight W_j from neuron j in the cortical layer to the readout neuron is given by

$$W_j = \sum_{m=1}^{P} \left(\overline{C}_j^m - f \right) L^m. \qquad \text{(Equation 4)}$$

Each cluster contributes a term to this weight that is equal to the product of the desired label of this cluster and the activity of the presynaptic cortical neuron in this cluster (relative to the mean activity, f). We find that the average classification error, ε, of the Hebbian readout is given by $\varepsilon = H(\sqrt{SNR})$, where $H(\cdot)$ is the tail probability of the standard normal distribution (the Q function), and SNR denotes the signal-to-noise ratio of the synaptic input to the readout neuron (see Experimental Procedures):

$$SNR = \frac{(1 - \Delta C)^2}{(\alpha_C + \alpha_S Q^2)}, \qquad \text{(Equation 5)}$$

where $\alpha_C = P/N_C$ and $\alpha_S = P/N_S$. The numerator measures the square difference between the mean inputs $\sum_{j=1}^{N_C} W_j (C_j^m - f)$ induced by +1 and −1 clusters. Note that this term decreases as the mean cortical cluster size ΔC increases. The denominator measures the variance of the inputs to the readout neurons and consists of two contributions: the first one is generated by the random overlaps between the classified input cluster and other cluster centers, and the second one originates from the excess overlaps between them (see Equation 2 and Experimental Procedures). This expression allows us to analyze the effect of sparseness and expansion on the performance of the readout.

Increasing the number of cortical neurons, N_C, causes the first term in the denominator of Equation 5 to decrease, thereby increasing the SNR and improving the readout performance (Figures 4A and 4B; Figure S1). Nevertheless, the SNR remains finite, even for arbitrarily large N_C, because the second term in the denominator, the contribution of the excess overlaps, is independent of N_C. Furthermore, there exists a characteristic size, N_C^{sat}, beyond which further expansion yields negligible improvement in performance. This saturation occurs when the first term in the denominator of Equation 5 becomes smaller than the second term, yielding

$$N_C^{sat} = \frac{N_S}{Q^2}. \qquad \text{(Equation 6)}$$

Since Q decreases with f, expansion is more effective when the representation is sparse. Equation 3 implies that $N_C^{sat} \propto N_S (f \log f)^{-2}$ for small f. Therefore, sparseness of cortical representations renders the expansion more effective (Figures 4 and S1).

Despite the fact that N_C^{sat} increases with sparseness, the overall performance of the readout is nonmonotonic as a function of the fraction of active neurons, f. This is because decreasing f decreases not only the noise (Equation 5, denominator) but also the signal (Equation 5, numerator), resulting in an optimal finite level of activity, f_{opt}, for which the readout error is minimized (Figure 4C; Figure S3, left):

$$f_{opt} \propto \left(\frac{N_S}{N_C} \right)^{1/2} (\Delta S)^{1/4}. \qquad \text{(Equation 7)}$$

Note, however, that the increase in error is rather shallow as f increases beyond f_{opt}.

Sparse Expansion through Structured Synapses

Our analysis of sparse and expansive cortical representations revealed some of the limitations of these representations: First, nearby patterns in stimulus layer are mapped into distal patterns in cortical layer, making the cortical representations sensitive to variations of the input. Second, excess overlaps between clusters, which are induced in the cortical layer, limit the benefit

(C) The size of cortical clusters, ΔC, decreases as a function of the fraction of active cortical neurons, f. Different curves correspond to different stimulus cluster sizes, ΔS. Other parameters of the network are the same as in (A).

(D) Schematic explanation of the effect of sparseness on the cortical cluster size. Solid curves show the distribution of the net input $\overline{h}$ received by cortical neurons in response to the central pattern of a stimulus cluster. The dashed vertical lines show the threshold of cortical neurons. Neurons that receive net inputs higher than the threshold fire (gray areas). Shaded areas show the fraction of neurons that cross the threshold and change their state in response to a typical pattern of the cluster. The shaded areas are large relative to the gray area when the representation is sparse (left), while they are relatively small when the representation is dense (right).

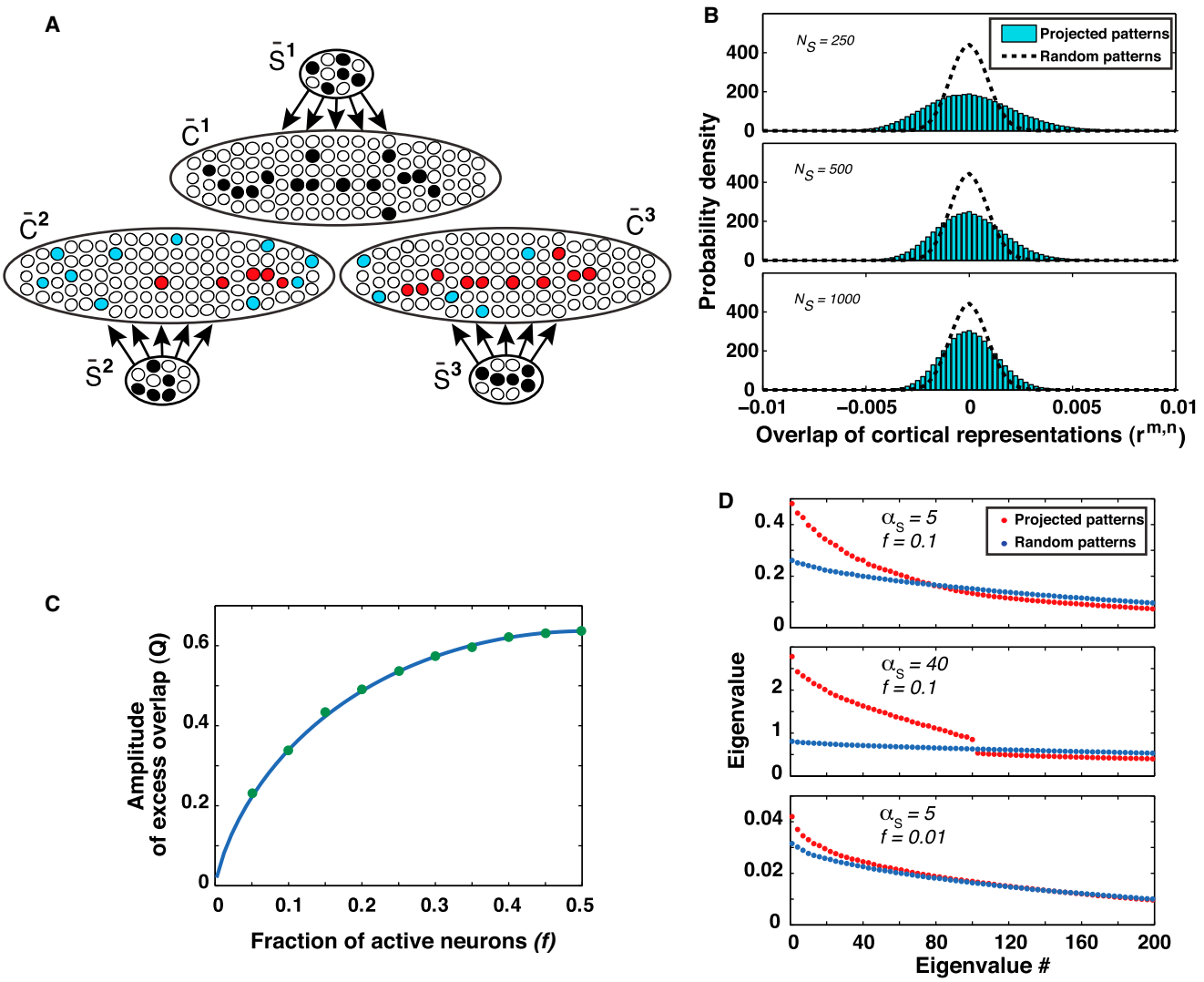

Figure 3. Overlaps between Cortical Representations in the Case of Random Projections

(A) Schematic description of the variance of overlaps between cortical patterns. The red neurons on the left are active in both cortical patterns $\overline{C}^1$ and $\overline{C}^2$, while the blue neurons are active in $\overline{C}^2$ but not in $\overline{C}^1$. The same color code is used to denote the overlap between $\overline{C}^1$ and $\overline{C}^3$ on the right. Although the average overlap is f^2, it deviates from pair to pair, i.e., $r^{1,2}$ and $r^{1,3}$ are different.

(B) The bars show the distribution of the deviations of the overlaps $r^{m,n}$, $m \neq n$ between all pairs of cortical patterns $\overline{C}^m$ and $\overline{C}^n$ projected from corresponding patterns in the stimulus layer through feed-forward random synapses, obtained by numerical simulations. The dotted curves show the distribution of overlaps between putative uncorrelated random cortical patterns. The width of the distribution is larger for the projected cortical patterns; therefore, there exists an "excess overlap" induced by the feed-forward projections. The width of the distribution for projected patterns decreases as the size of the stimulus layer, N_S, increases (top to bottom). Other parameters are $N_C = 10000$, $P = 1000$, and $f = 0.05$.

(C) Amplitude of the excess overlap, Q, increases as a function of the fraction of active cortical neurons, f. Other parameters are $N_S = 1000$, $N_C = 10000$, and $P = 1000$.

(D) Red dots show the principal components of the cortical patterns $(\overline{C}^m - f)$, i.e., eigenvalues of the overlap matrix $r^{m,n}$ (including the diagonal elements), for $N_S = 100$, $N_C = 1000$, and $f = 0.1$. Blue dots show the principal components of putative random uncorrelated cortical patterns for comparison. The first N_S eigenvalues are enhanced. When $\alpha_S = P/N_S$ is very large (middle), there is a gap in the spectrum. Sparser representations lead to a spectrum that is more similar to that of the random uncorrelated patterns (bottom).

of expansion. Finally, increasing sparseness (i.e., decreasing f) improves the performance of the readout up to some finite levels, beyond which further sparseness degrades it.

Some of these limitations are due to the randomness of the synaptic weights that implement the transformation from the stimulus to the cortical layer. As an alternative, it is reasonable to hypothesize that the synaptic weights encode information about the statistics of the inputs, which, in our case, is their clustered structure. Here, we propose a simple scheme based on associating the stimulus clusters with random $f-$ sparse cortical patterns. These associations are encoded in the synaptic weights J_{ji} through the following rule,

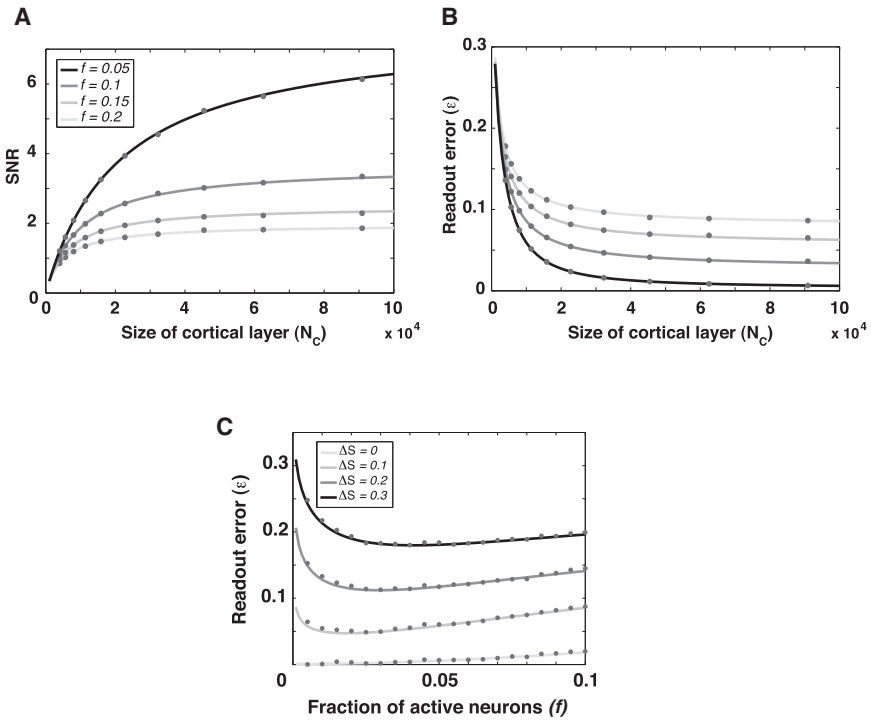

Figure 4. The Effect of Expansion and Sparseness on Readout Performance in the Case of Random Projections

(A) *SNR* of the readout increases as a function of the size of cortical layer, N_C; however, it reaches a plateau. Different curves correspond to different activity levels, f, of the cortical layer.

(B) The error of the Hebb readout decreases as a function of the size of cortical layer, N_C; however, it reaches a plateau. Different curves correspond to different activity levels, f. The stimulus cluster size is fixed at $\Delta S = 0.1$. Other parameters are $N_S = 1000$ and $P = 1000$.

(C) The error of the Hebb readout is nonmonotonic as a function of the fraction of active cortical neurons, f. There is an optimal activity level, f_{opt}, for which the readout error is minimized. Different curves correspond to different stimulus cluster sizes, ΔS. Other parameters are $N_S = 1000$, $N_C = 10000$, and $P = 1000$.

$$J_{ji} = \frac{1}{N_S}\sum_{m=1}^{P}\left(\overline{S}_i^m - \frac{1}{2}\right)\left(R_j^m - f\right), \qquad \text{(Equation 8)}$$

where R^m is a random f – sparse cortical state associated with the m th cluster center, $\overline{S}^m$. This rule can be interpreted as resulting from a sum of Covariance Hebb modifications (Tsodyks and Feigelman, 1988) induced by each pairing of $\overline{S}^m$ and R^m (see Discussion). As in the random weight scenario, cluster center input patterns $\overline{S}^m$ and cluster members S^m induce cortical cluster centers $\overline{C}^m$ and cluster members C^m, respectively, by linear summations with weights J and thresholding with threshold T to ensure the desired sparseness f. Note that, in contrast to the supervised Hebb rule of the output weights (Equation 4), J of Equation 8 does not incorporate information about the behavioral efficacy of the stimuli, i.e., their labels L^m.

In contrast to the random weights, the structured synapses (Equation 8) yield cortical clusters with mean size, ΔC, which is significantly smaller than the stimulus cluster size ΔS (Figure 5A). Furthermore, ΔC decreases rapidly with decreasing f (Figure 5B), as can be seen from the following expression (valid for finite ΔS and $f \ll 1$):

$$\Delta C \approx \frac{\sqrt{2\alpha_S f}}{\sqrt{\pi}(1-\Delta S)}\exp\left(-\frac{(1-\Delta S)^2}{8\alpha_S f}\right) \qquad \text{(Equation 9)}$$

(see Experimental Procedures). In the asymptotic limit of $f \to 0$, the size of cortical clusters approaches zero as long as ΔS is smaller than 1.

A qualitative explanation for this result is presented in Figure 5C. Consider cortical neurons that receive inputs which are highly overlapping with one cluster center, say $\overline{S}^1$. According to Equation 8, the mean total synaptic inputs to neurons with $R_j^1 = 1$ will be positive (proportional to $1 - f$), whereas for those with $R_j^1 = 0$, it will be negative (proportional to $-f$). This results in a bimodal input distribution composed of two Gaussians (Figure 5C, solid curves), as opposed to the unimodal distribution of Figure 2D. The width of both Gaussians (which is proportional to $f(1-f)$; see Experimental Procedures) decreases when the representation becomes sparser (Figure 5C, left versus right). Therefore, sparser representations lead to a better separation of the two Gaussians so that the area that contributes to ΔC is exponentially small when $f \ll 1$. This provides resilience to noise and variations in the input. As suggested by Equation 9, this structure depends not only on f but also on the number of clusters per input neuron, α_S. Thus, when $\alpha_S f \gg 1$, the overlap between the two Gaussians becomes larger than their widths and ΔC behaves similarly to the case of random synapses (Figure 5B, bottom).

Another advantage of the structured connections is the strong suppression of the excess overlaps between cortical clusters, in particular, for low f (Figure 6, top). In the limit of sparse representations ($f \ll 1$), this overlap decreases exponentially with $1/\alpha_S f$:

$$Q \propto f^{-1}\exp\left(-\frac{1}{4\alpha_S f}\right), \qquad \text{(Equation 10)}$$

in contrast to the approximately linear dependence in the case of random weights (Equation 3). The reason for the strong suppression in the overlap is that the cortical patterns $\overline{C}^m$ induced by structured connections are close to the random states R^m (which have zero excess overlap). As before, in the limit of $\alpha_S f \gg 1$, the structured projections behave as random projections, destroying the relationship between individual cluster centers $\overline{C}^m$ and the paired random patterns R^m and yielding a similar excess overlap as in random projections (Figure 6, bottom).

As in the case of random projections, the *SNR* of the readout is given by Equation 5; hence, a finite saturation of performance with expansion, given by Equation 6, holds also for structured expansion. However, here, the saturation size of the cortical

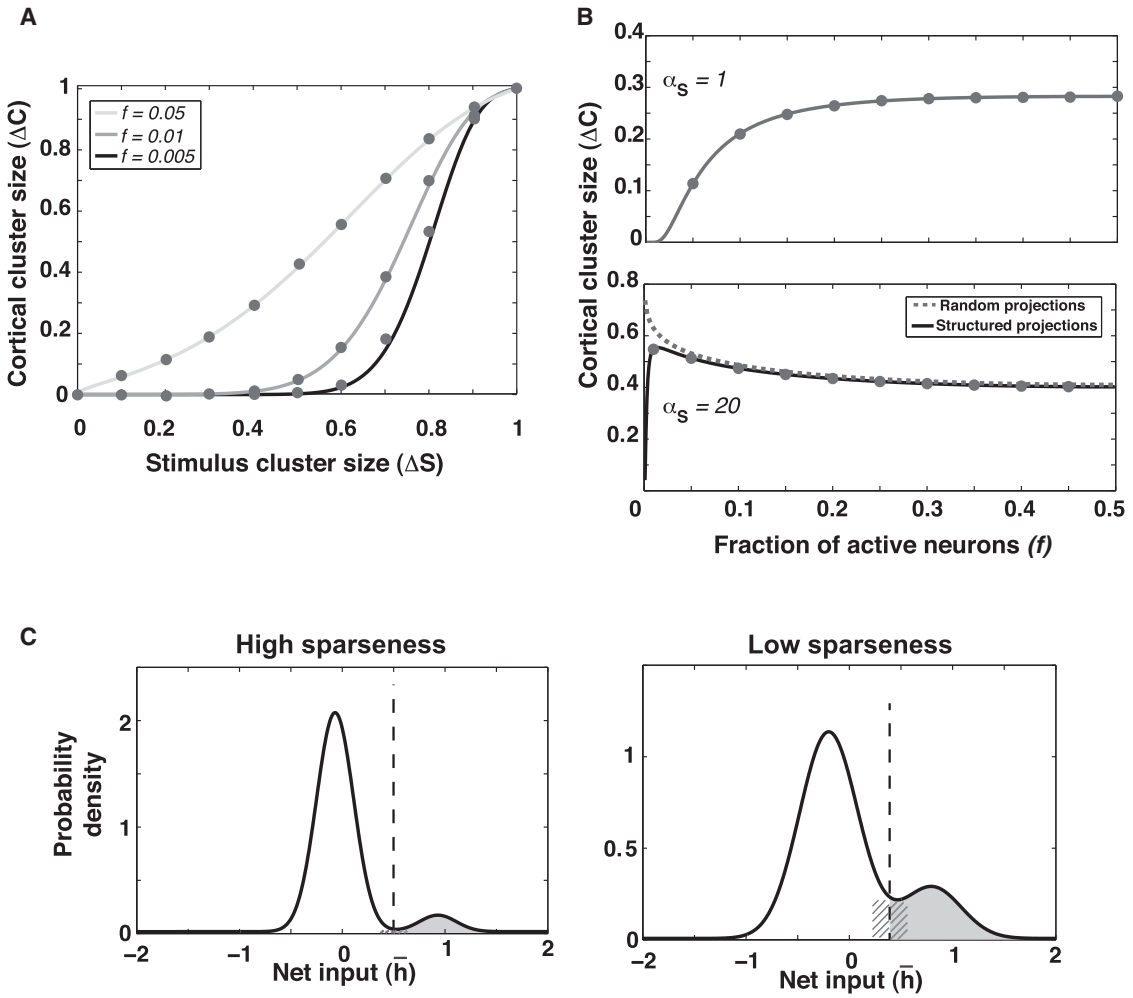

Figure 5. Size of Cortical Clusters in the Case of Structured Projections

(A) Shrinkage of the size of cortical clusters, ΔC, compared to that of the stimulus clusters, ΔS. Different curves correspond to different cortical activity levels, f. Other parameters of the network are $N_S = 1000$, $N_C = 10000$, and $P = 1000$.

(B) The size of cortical clusters, ΔC, as a function of the fraction of active cortical neurons, f. ΔS is fixed at 0.1. When the number of clusters per input neuron, α_S, is large, the behavior of structured synapses converges to the random case (bottom). Other parameters of the network are the same as in (A).

(C) Schematic explanation of the effect of sparseness on cortical cluster size. Solid curves show the distribution of the net input received by cortical neurons in response to a central pattern of a cluster in the stimulus layer. The dashed lines show the threshold of cortical neurons. Neurons that receive net inputs higher than the threshold fire (gray areas). Shaded areas show the fraction of neurons that cross the threshold and change their state in response to a typical pattern of the cluster. When the cortical activity is sparse (left), the two peaks of the distribution are well separated, and the shaded areas are very small relative to the gray area.

layer (calculated via Equation 10) increases exponentially with f^{-1}, as $N_C^{sat} \propto f^2 \exp(1/2\alpha_S f)$. Thus, for sparse representations, expansion is highly effective at improving the *SNR* (Figures 7A and 7B; Figure S3, right). Additionally, sparseness not only decreases the noise but also increases the signal (Figure 5); hence, the overall performance monotonically decreases with sparseness (Figure 7C; Figures S2, right, and S3, left).

Stimuli with Intrinsic Low Dimensionality

In our model, stimuli are distributed as random N_S dimensional patterns so that different cluster centers are not correlated. However, many natural stimuli exhibit strong correlations, resulting in intrinsic dimensionality that is significantly lower than their dimensionality. By intrinsic dimensionality of a collection of stimulus representations, we mean the dimensionality of the manifold on which they (approximately) lie, whereas their dimensionality refers to the number of pixels or neurons participating in the representation. For instance, in vision, most of the power of natural images lies in the low Fourier components, reflecting the strong correlations between nearby pixels (Field, 1987). In olfaction, behavioral and neuronal data suggest that natural stimuli lie on a low dimensional manifold (Haddad et al., 2010; Koulakov et al., 2011). Intrinsic low dimensional representations may also emerge in association cortices that encode both stimulus and contextual information (Miller and Cohen, 2001). To study the effect of expansion and sparseness on stimulus representations

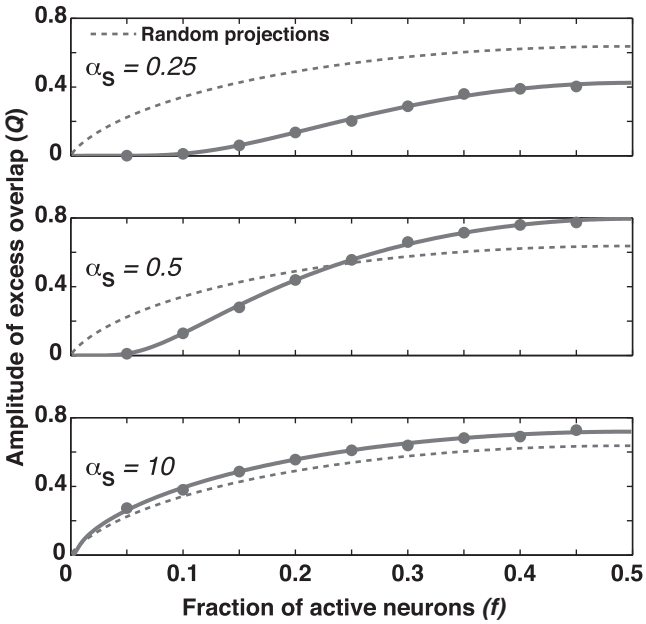

Figure 6. Amplitude of Excess Overlap between Cortical Representations in the Case of Structured Projections

Solid curves show the amplitude of excess overlap as a function of the fraction of active cortical neurons, *f*. Dashed curves show the amplitude of excess overlap in the case of random projections for comparison. The number of clusters (*P* and, hence, α_S) increases from top to bottom. Other parameters are N_S = 1000 and N_C = 10000. Note that the dashed curves do not change by changing the number of clusters.

with intrinsic low dimensional structure, we generated N_S dimensional vectors $\overline{S}^m$ with intrinsic dimensionality *M* with $M < N_S$ (Supplemental Information; Figure S6A). As before, expansion improves the performance with an asymptote to a nonzero error for large N_C (Figure S6D). Random projections yield a performance that is optimal at a finite value of *f* (example shown in Figure S6C). Sparseness considerably improves the performance in the case of structured projections (Figure S6E, right).

Performance in Other Tasks

In the preceding text, we have analyzed the performance of readout in binary classification tasks in which the clusters are labeled as null or target at random with ½ probability. The results of our analysis hold qualitatively for other tasks as well. For instance, consider the case where the number of target clusters (e.g., appetitive stimuli) is much smaller than the number of null stimuli. This can be modeled by assuming that the readout labels have a probability *l* of being one and $1 - l$ of being zero, with small *l*. In the extreme case where $l = 1/P$, the task can be interpreted as an "identification" task, i.e., the readout neuron has to signal the presence (or absence) of one particular stimulus. We find that Equation 5 for the *SNR* holds also for these tasks, except for an overall increase by a factor of $4l(1 - l)$ (see Supplemental Information). Therefore, sparseness of the readout tasks improves the performance, while the effects of cortical sparseness, expansion, and structuring the input weights remain unchanged (Figure S7).

Another interesting task is a reconstruction task. In this case, the readout layer consists of N_S neurons that are required to reconstruct for each noisy stimulus pattern, S^m, its cluster center, $\overline{S}^m$. To perform this task, the cortical layer is connected to the readout layer by an $N_C \times N_S$ dimensional weight matrix *W* (Figure S8A), with weights that are given by a Hebb rule, associating the cortical states $\overline{C}^m$ with the desired output states $\overline{S}^m$. The performance of this reconstruction task is, in general, superior to that of the classification task, particularly when the cortical representation is dense. The qualitative effect of sparseness and expansion remains the same as that of the classification task.

DISCUSSION

Our analysis highlights the subtle effects of sparseness and expansion, two fundamental concepts in the theory of biological and artificial signal processing. It is often argued that sparseness suppresses noise because two *N* -bit patterns with sparseness level *f* can deviate by, at most, *2Nf* bits. We argue, however, that the relevant measure of noise is the distance between two sparse patterns belonging to the same stimulus or category, relative to the distance between two random sparse patterns with the same level of sparseness, a measure that is encapsulated in our normalized distances or cluster sizes ΔS and ΔC. This normalization is ultimately justified by the analysis of the readout *SNR*, which depends on the cluster size through $1 - \Delta C$. As shown here, whether sparseness shrinks ΔC or not depends on the nature of the synaptic weights *J*. Random projections into sparse representations increase the normalized distances, thereby degrading the ability of the system to identify nearby input patterns as equivalent; as a consequence there is a nonzero optimal sparseness, f_{opt} (cf. Equation 7). A finite optimal sparseness has been also found numerically in a model for processing inputs that lie in low dimensional subspace by projecting them with random weights (Barak et al., 2013).

Sparse expansion via projections that incorporate the clustered structure of the inputs shrinks the distance between patterns belonging to the same cluster, conferring enhanced robustness to the system. In addition, by pairing input states with random cortical patterns, these connections greatly suppress correlations between cortical representations, especially when they are sparse. As a result, increasing the sparseness of the representation improves the readout performance, yielding error levels that are substantially lower than those achieved by random projections. Increasing sparseness is beneficial as long as the number of active neurons in the sparse patterns (fN_C) remains large. Otherwise, the fluctuations in the number of neurons that represent a cluster limit the performance of the system.

It is well known that a nonlinear mapping of a set of low dimensional inputs into a high dimensional space enhances their linear separability, namely, the ability of classifying them by downstream linear classifier units. Similarly, in our case, the number of patterns that can be perfectly classified by a suitable linear readout unit increases with the dimensionality of the inputs to the readout, N_C, independent of their sparseness level. Surprisingly, however, we have found that when one considers classifications of the entire clusters (i.e., $\Delta S > 0$), there is a limit to the

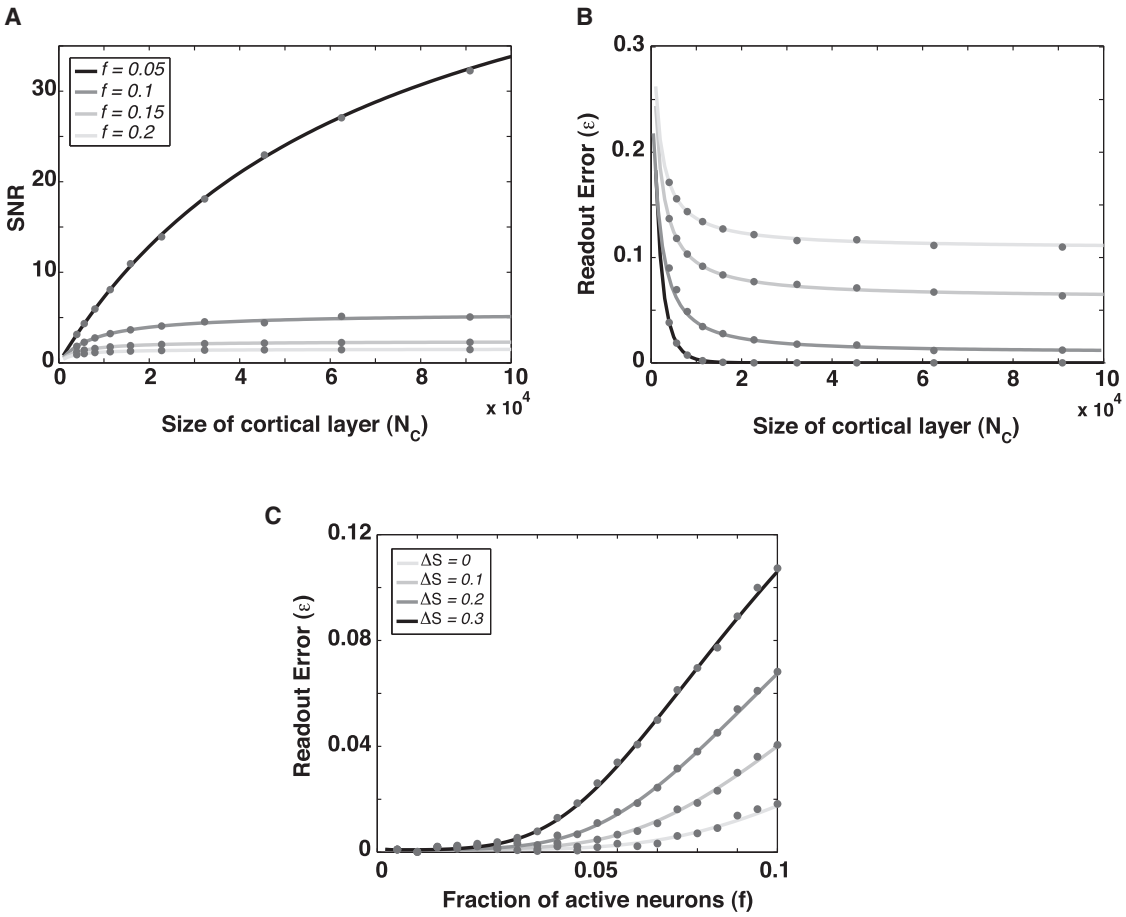

Figure 7. The Effect of Sparseness and Expansion on Readout Performance in the Case of Structured Projections

(A) *SNR* of the readout increases as a function of the size of cortical layer, N_C; however, it reaches a plateau. Different curves correspond to different cortical activity levels, *f*.

(B) The error of the Hebb readout decreases as a function of the size of cortical layer, N_C; however, it reaches a plateau. Different curves correspond to different cortical activity levels, *f*. Sparser cortical representations lead to a higher *SNR* and lower readout error. Note that the *SNR* is considerably higher compared to random projections (Figure 4A), and in the case of highly sparse representations (black curve), it does not saturate in the given range. The stimulus cluster size is fixed at $\Delta S = 0.1$. Other parameters are $N_S = 1000$ and $P = 1000$.

(C) The error of the Hebbian readout is monotonically increasing as a function of the fraction of active cortical neurons, *f*, as opposed to the case of random projections (Figure 4C), i.e., sparseness decreases the error unboundedly. Different curves correspond to different stimulus cluster sizes, ΔS. Other parameters are $N_S = 1000$, $N_C = 10000$, and $P = 1000$.

gain achieved by expansion (as demonstrated by our N_C^{sat}; Equation 6). This saturation stems from the excess overlaps between the cortical patterns inherited from the original low dimensional stimulus layer. These overlaps scale inversely with the input dimension, N_S, and are therefore not affected by the expansion dimension, N_C. This saturation occurs for both random and structured expansions; however, the saturation size for the structured expansion is substantially larger than that of the random expansion when the cortical representation is highly sparse. We have shown that the excess overlaps can be observed by a PCA of the covariance matrix of cortical activity patterns generated by distinct stimuli, provided that the number of stimuli is large compared to stimulus dimensionality, N_S. In such a case, the theory predicts a distinct band of enhanced power of the first N_S modes, reflecting the source low dimensionality of the cortical representations (Figure 3D).

Our conclusions about readout performance are based on the analysis of a supervised Hebb rule for readout weights, which is tuned to classify the central pattern of each cluster. As an alternative training scheme, the readout neuron can be trained to classify samples of the typical, noisy input patterns belonging to the clusters (Supplemental Information). Training the Hebb readout with these inputs reduces both the signal and the noise of the readout (numerator and denominator of Equation 5, respectively). For most parameter ranges, this results in degradation in performance compared to training with the cluster centers. Nevertheless, the qualitative behavior of the readout remains the same (Figure S4A).

The supervised readout Hebb learning rule has the advantage of being amenable to systematic analytic study and is biologically plausible. Linear classifiers with more complex learning rules such as pseudoinverse, perceptron, and SVM, are more

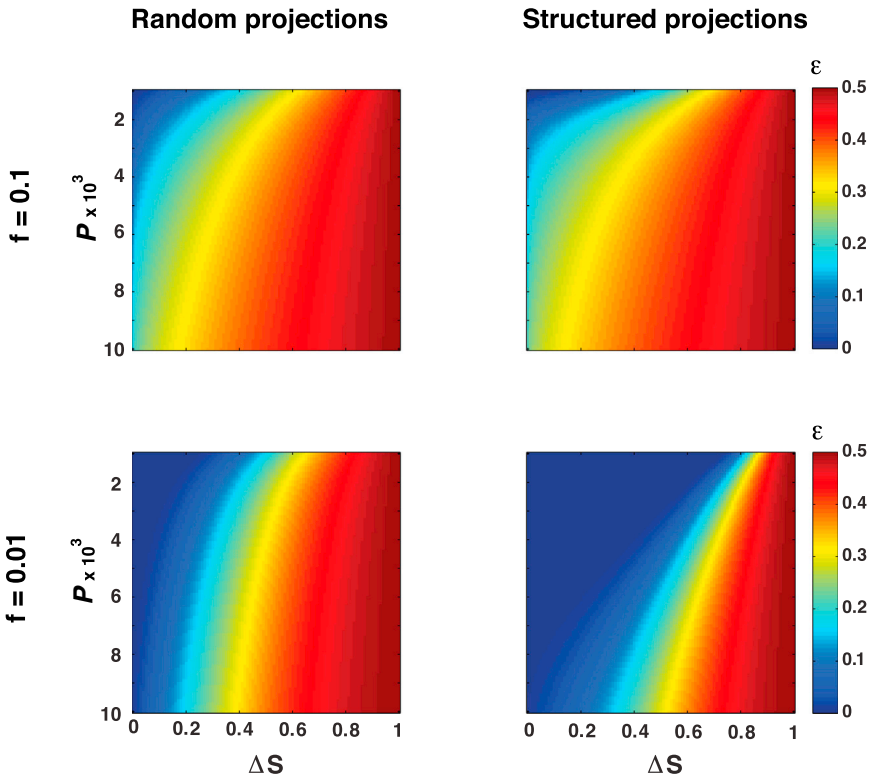

Figure 8. Comparison of Readout Performance between Random Projections, at Left, and Structured Projections, at Right, When the Fraction of Active Neurons is *f* = 0.1, in the Top Row, and *f* = 0.01, in the Bottom Row

Horizontal and vertical axes indicate the stimulus cluster size, ΔS, and the number of clusters, P, respectively. The color scale shows the readout error. The sizes of the stimulus and cortical layers are fixed at N_S = 1000 and N_C = 500,000, respectively. Note that the difference between the two modes of projections is more manifest when the cortical activity is sparse (bottom).

powerful in that they allow for perfect classification of noiseless patterns, up to their capacity limit, whereas the *SNR* of the Hebb rule is small but nonzero, even for $\Delta S = 0$ (compare Figure 4C with Figure S3, left). In addition, in the presence of noise, their performance is generally better than the simpler Hebb readout. Although a systematic analytical study of these classifiers is not available, our numerical simulations of a range of parameters suggest that the behavior of these linear classifiers is qualitatively similar to the Hebb rule (Figures S1–S3 and S4B). In particular, the error of all these classifiers saturates to a finite value as N_C increases, and with random expansion, but not with structured expansion, there is an optimal sparseness level that minimizes the readout error.

Our results are consistent with recent compressed sensing and sparse coding theories (for a review, see Ganguli and Sompolinsky, 2012). While random projections of sparse signals are effective in compressing them into low dimensions, generating an appropriate expanded sparse code requires a nonlinear computation that incorporates information about the hidden structure of the compressed signals (e.g., the "dictionary" of their sparse basis vectors). Our Hebbian model for J plus the subsequent threshold nonlinearity can be viewed as a simple, biologically plausible approximation to the more complex machine learning algorithms for sparse reconstruction (Ganguli and Sompolinsky, 2012; Rozell et al., 2008). To demonstrate the relation of our work to sparse coding in vision, we have applied our scheme to inputs taken from natural images. The stimulus layer of size N_S = 256 consists of neurons that code the analog pixel values of 16 × 16 whitened patches of natural images. Mapping of these inputs to a cortical layer of size N_C = 5000 was implemented using online Hebb learning rule for the synaptic matrix J (Supplemental Information; also discussed later). It is interesting that imposing a high degree of sparseness yields Gabor-like cortical receptive fields (Figure S5), which is the hallmark of sparse coding theory of early vision (Field, 1994; Bell and Sejnowski, 1997). Thus, the combined effect of averaging the input stream using the Hebb learning and competition at the cortical layer imposed by sparseness results in a synaptic matrix that encodes the hidden clustered structure of natural image patches, with Gabor-like cluster centers (Saxe et al., 2011).

Application to Biological Systems

Comparing our results to specific neuronal systems requires an estimate of several key parameters, such as the sparseness level of the cortical layer, f; the number of behaviorally relevant distinct stimuli or objects represented in the system, P; and the magnitude of variability or noise in their primary sensory representation, i.e., ΔS. To provide a concrete plausible setting, we show in Figure 8 the case of $N_S = 10^3$ and $N_C = 0.5 \times 10^6$, an expansion that is roughly the same order of magnitude as in rodent olfactory system (where roughly 10^3 glomeruli project to 10^6 neurons in the piriform cortex) and in cerebellum (where a single Purkinje cell receives input from roughly 10^3 mossy fibers via an expanded layer of 200,000 parallel fibers). For the number of distinct stimuli, we take the range $N_S \leq P \leq 10N_S$. Figure 8 shows that, for a moderate sparseness level $f = 0.1$, which might characterize the activity level in piriform cortex, the performance of the structured expansion is equal or even slightly worse than the random case (Figure 8, top). In contrast, in the case of f = 0.01, which is similar to that of the granular layer in cerebellum (Chadderton et al., 2004; Galliano et al., 2013), the structured projections are superior even for a number of distinct stimuli ten times larger than the size of the input layer (Figure 8, bottom). Additionally, for f = 0.1, our theory predicts that the benefit of expansion saturates at $N_C^{sat} \approx 10^4$, suggesting that the size of the piriform cortex serves other computational purposes than those considered here. In contrast, for f = 0.01, $N_C^{sat} \approx 2 \times 10^5$, which is consistent with the expansion in the cerebellum.

Analyzing the projections from the glomeruli in the antenna lobe to the Kenyon cells in the mushroom body of the fly, Caron et al. (2013) concluded that the observed connectivity is indistinguishable from a random 0,1 matrix. It should be noted, however, that our structured projections may be built from a random 0,1 connectivity by appropriate modification of the magnitude of the nonsynaptic efficacies. Thus, a definitive test of the random versus structured scenarios must require data not only about connectivity but also about the synaptic strengths. An alternative experimental test is to measure intracellularly the distribution of the net synaptic potentials induced by a set of natural stimuli. In the case of structured projections, this distribution should be bimodal, with the firing threshold in between the two modes, separating the preferred from the null stimuli for each neuron.

Our model of structured projections (Equation 8) associates each stimulus cluster with a random sparse cortical pattern, raising the question of what might be the biological mechanism of this allocation process. One possible scenario is that, throughout learning or development, noise is injected to the cortical layer during stimulus presentation, resulting in random cortical activation patterns. A Hebbian plasticity will then associate these cortical patterns with the corresponding stimulus inputs. The notion that noise is injected to cortical neuronal responses during learning, presumably by basal ganglia structures, has gathered support from recent experiments in birdsong learning and motor association learning in primates (Ölveczky et al., 2005; Sheth et al., 2011). It is argued that this variability enhances exploratory behavior, which benefits reinforcement-like learning. Our results suggest that, in addition to this role, enhanced variability during learning is beneficial as an effective mechanism for forming associations between inputs and randomized cortical patterns. A recent theory proposes that the hippocampus plays a central role in the process of allocations of neurons in cortex to form new "concepts" (Valiant, 2012). Alternatively, the feed-forward connections may start initially as random projections and generate initial cortical representations, which are then subjected to online Hebbian modifications (see Supplemental Information). This scheme, which combines elements from our two models of J, results in a performance that is intermediate between the random and structured models studied earlier (see Figure S5). An alternative hybrid architecture is a combination of feed-forward random projections with Hebbian recurrent connectivity in the cortex. Thus, a fuller understanding of the computational principles of cortical sensory transformations will require extending our architecture to include both recurrent and feedback connections.

EXPERIMENTAL PROCEDURES

Numerical Simulations

Stimulus and Cortical Clusters

Central patterns of each cluster, $\overline{S}^m$, in the stimulus layer are generated as arrays of N_S {0,1} bits i.i.d. with equal probabilities. To generate other members of a cluster—say, S^m —the value of each element of $\overline{S}^m$ is independently flipped (zero to one or one to zero) with probability $\Delta S/2$. This guarantees that the cluster size is ΔS.

Cortical responses for each stimulus pattern—say, S^m —are calculated by evaluating the weighted sums $h_j^m = \sum_{i=1}^{N_S} J_{ji}(S_i^m - 1/2)$ for each cortical neuron j. Note that the offset $-1/2$ (which can be incorporated in the definition of the neuron's threshold) enhances the *SNR* of the cortical neurons. The cortical pattern is obtained as $C_j^m = \Theta(h_j^m - T)$, where $\Theta(\cdot)$ denotes the Heaviside step function. The threshold T is set so that a fraction f of all PN_C realizations of h_j^m are larger than T. In order to calculate the cortical cluster sizes ΔS and ΔC, one representative typical pattern is generated from each of the P clusters, and the values are averaged over the P clusters. The numerical results appear as filled circles in Figures 2, 3, 5, and 6.

Linear Readouts

The synaptic weights of the Hebbian readout are obtained according to Equation 4. In addition, we studied the performance of alternative linear readouts (see Supplemental Experimental Procedures). In all cases, the readout error is calculated as:

$$\varepsilon = \left\langle \left| L^m - \mathrm{sign}\left(\sum_{j=1}^{N_C} W_j \left(C_j^m - f \right) \right) \right| \right\rangle \Big/ 2.$$

The average is over P cortical pattern/label pairs (C^m,L^m), with one representative typical pattern from each cluster. In Figures 4, 7, and S1–S7, the readout error averaged over 400 different realizations of the readout is shown, each with its own randomly chosen labeling of the clusters.

Analytical Calculations

All analytical results are derived in the limit of large N_S, N_C, and P, while $\alpha_S = P/N_S$ and $\alpha_C = P/N_C$ are finite. We also assume that $N_C f \gg 1$. Central limit theorem implies that the input potentials to cortical neurons, as well as to the readout neuron, are Gaussian. The first-order and second-order statistics is readily calculated from the statistics of the input patterns and synaptic weights.

Size of Cortical Clusters

The cortical cluster size can be formulated as:

$$\Delta C = \frac{\sum_{j=1}^{N_C} \left(Pr\left(C_j^m = 1, \overline{C}_j^m = 0 \right) + Pr\left(C_j^m = 0, \overline{C}_j^m = 1 \right) \right)}{2N_C f(1-f)} = \frac{Pr\left(C_j^m = 0, \overline{C}_j^m = 1 \right)}{f(1-f)} \quad \text{(Equation 11)}$$

The net input received by j th cortical neuron from m th stimulus cluster center is $\overline{h}_j^m = \sum_{i=1}^{N_S} J_{ji}(\overline{S}_i^m - 1/2)$. Similarly, the net input in response to a typical member of m th cluster is $h_j^m = \sum_{i=1}^{N_S} J_{ji}(S_i^m - 1/2)$. The joint probability in Equation 11 can be written as $Pr(h_j^m < T, \overline{h}_j^m > T)$. In the case of random projections, both $\overline{h}_j^m$ and h_j^m have zero mean and unite variance, and their covariance is $\langle \overline{h}_j^m h_j^m \rangle = (1 - \Delta S)$. Therefore, the size of cortical clusters is:

$$\Delta C = \frac{1}{f(1-f)} \int_T^\infty dh \frac{\exp\left(-\frac{h^2}{2} \right)}{\sqrt{2\pi}} H\left(\frac{(1-\Delta S)h - T}{\sqrt{\Delta S(2-\Delta S)}} \right). \quad \text{(Equation 12)}$$

Threshold T, which imposes sparseness, is obtained from the condition $f = H(T)$, where $H(x) = \int_x^\infty dx \exp(-x^2/2)/\sqrt{2\pi}$. The aforementioned exact form of ΔC is illustrated as solid curves in Figures 2B and 2C, while Equation 1 is its approximation in the limit of small ΔS and small f.

In the case of structured projections, the means of net inputs are $\langle \overline{h}_j^m \rangle = (R_j^m - f)$ and $\langle h_j^m \rangle = (R_j^m - f)(1 - \Delta S)$, respectively; their variance is $\sigma^2 = \alpha_S f(1-f)$; and their covariance is $(1 - \Delta S)\sigma^2$. Therefore, the size of cortical clusters is:

$$\Delta C = \int_{T_0}^\infty dh \frac{f \exp\left(-\frac{(h-(1-f))^2}{2\sigma^2} \right) + (1-f)\exp\left(-\frac{(h+f)^2}{2\sigma^2} \right)}{\sqrt{2\pi}\sigma f(1-f)} \times H\left(\frac{(1-\Delta S)h - T}{\sqrt{\Delta S(2-\Delta S)}\sigma} \right), \quad \text{(Equation 13)}$$

where T_0 and T are the threshold values in the case of responding to central and typical patterns, respectively. They are obtained from the condition of maintaining sparseness f in the cortical layer:

$$f = (1-f)H\left(\frac{T + f(1-\Delta S)}{\sigma} \right) + fH\left(\frac{T - (1-f)(1-\Delta S)}{\sigma} \right). \quad \text{(Equation 14)}$$

Note that T_0 is obtained by setting $\Delta S = 0$ in the aforementioned condition. The exact form of ΔC is illustrated as solid curves in Figures 5A and 5B, while Equation 9 is its approximation in the limit of small f and finite ΔS.

Amplitude of Excess Overlap

Random Projections. We define the overlap between a pair of cortical clusters as $r^{m,n} = \sum_{j=1}^{N_C} (\overline{C}_j^m - f)(\overline{C}_j^n - f)/N_C, m \neq n$. Its average over J is zero. Its variance can be written as in Equation 2 with:

$$Q^2 = \frac{N_S}{(N_C P f(1-f))^2} \times \sum_{j=1}^{N_C} \sum_{j'=1}^{N_C} \sum_{j \neq j'} \sum_{m=1}^{P} \sum_{n=1}^{P} \sum_{n \neq m} \left\langle \left(\overline{C}_j^m - f\right) \times \left(\overline{C}_{j'}^n - f\right)\left(\overline{C}_j^m - f\right)\left(\overline{C}_{j'}^n - f\right)\right\rangle. \quad \text{(Equation 15)}$$

The averaging is over stimulus patterns $(\overline{S}^m, \overline{S}^n)$ and also the feed-forward synaptic weights J, from which the cortical patterns $(\overline{C}^m, \overline{C}^n)$ originate. Note that the terms $j = j'$ contribute as $1/N_C$, which is the random overlap contribution to Equation 2; hence, they are not included in Equation 15. Thus, if the patterns $\overline{C}^m$ were independent random states, $Q = 0$. Nonzero Q reflects the correlations induced to the cortical patterns by the filtering of the random input states through the same projection matrix J. The averaging yields the following expression for amplitude of excess overlap:

$$Q = \frac{\exp(-T^2)}{2\pi f(1-f)}. \quad \text{(Equation 16)}$$

This exact form of Q is illustrated as solid curves in Figure 3C, while Equation 3 is its approximation in the limit of small f.

Structured Projections. In the case of the structured synapses and $\Delta S > 0$, the definition of the overlap between clusters must take into account the different statistics of cortical patterns C^m induced by noisy inputs S^m, in addition to cluster centers $\overline{C}^m$, which appear in the expression of the readout synapses W (Equation 4). Thus, the relevant definition of overlap between a pair of cortical clusters is $r^{m,n} = \sum_{j=1}^{N_C} (\overline{C}_j^m - f)(C_j^n - f)/N_C, m \neq n$, yielding

$$Q^2 = \frac{N_S}{(N_C P f(1-f))^2} \times \sum_{j=1}^{N_C} \sum_{j'=1}^{N_C} \sum_{j \neq j'} \sum_{m=1}^{P} \sum_{n=1}^{P} \sum_{n \neq m} \left\langle \left(\overline{C}_j^m - f\right) \times \left(C_{j'}^n - f\right)\left(\overline{C}_j^m - f\right)\left(C_{j'}^n - f\right)\right\rangle. \quad \text{(Equation 17)}$$

Note that here, even in the case of $\Delta S = 0$, Q^2 is nonzero. The reason is that, although the cortical states R^m that appear in J (Equation 8) are uncorrelated, the actual cluster centers $\overline{C}^m$ are correlated as they are affected by all random states R^m, through filtering of the random stimulus patterns through the same projection matrix J. The averaging in Equation 17 is performed over stimulus patterns $(\overline{S}^m, \overline{S}^n)$, the random states (R^m, R^n) incorporated in the feed-forward synaptic weights J, and stimulus patterns (S^m, S^n) from which the cortical patterns (C^m, C^n) originate. In the case of $\Delta S = 0$, the averaging yields:

$$Q = A\sqrt{\alpha_S A^2 + (\alpha_S A + 2B)^2}, \quad \text{(Equation 18)}$$

where A and B are given as:

$$A = \frac{f \exp\left(-\frac{(T_0 - (1-f))^2}{2\alpha_S f(1-f)}\right) + (1-f)\exp\left(-\frac{(T_0+f)^2}{2\alpha_S f(1-f)}\right)}{\sqrt{2\pi\alpha_S f(1-f)}}, \quad \text{(Equation 19)}$$

$$B = H\left(\frac{T_0 - (1-f)}{\sqrt{\alpha_S f(1-f)}}\right) - H\left(\frac{T_0 + f}{\sqrt{\alpha_S f(1-f)}}\right)$$

The exact form of Q (Equation 18) is illustrated as solid curves in Figure 6, while Equation 10 is its approximation in the limit of small f.

Readout Error

The net input to the readout neuron in response to a pattern belonging to cluster m is $g^m = \sum_{j=1}^{N_C} W_j (C_j^m - f)$. If $g^m > 0$, the readout classifies the pattern as belonging to class 1, and otherwise belonging to class -1. Conditioned on the class label, the net input g^m has a mean:

$$\langle g^m \rangle = \left\langle \sum_{j=1}^{N_C} W_j \left(C_j^m - f\right) \right\rangle = \left\langle \sum_{j=1}^{N_C} \sum_{n=1}^{P} \left(\overline{C}_j^n - f\right)\left(C_j^m - f\right) L^n \right\rangle = N_C f(1-f)(1 - \Delta C) L^m. \quad \text{(Equation 20)}$$

The averaging is over all P cluster center/label pairs $(\overline{C}^n, L^n)$. With a similar averaging, the variance of the synaptic input to the readout is:

$$\sigma_g^2 = \left\langle (g^m)^2 \right\rangle - \langle g^m \rangle^2 \approx N_C P f^2 (1-f)^2 + N_C^2 P f^2 (1-f)^2 Q^2 \Big/ N_S. \quad \text{(Equation 21)}$$

The first term in the aforementioned equation arises from random overlaps between cortical patterns, and the second term arises from excess overlaps; hence, it contains Q^2. Assuming that the synaptic input to the readout neuron obeys a normal distribution, the probability of misclassification of pattern S^m is $\varepsilon = H(L^m \langle g^m \rangle / \sigma_g)$. The error ε calculated in this way appears as solid curves in Figures 4 and 7 and the panels of Figure 8. In analogy with ideal observer theory, the square of the argument of H represents the *SNR* of the system.

SUPPLEMENTAL INFORMATION

Supplemental Information includes Supplemental Experimental Procedures and eight figures and can be found with this article online at http://dx.doi.org/10.1016/j.neuron.2014.07.035.

ACKNOWLEDGMENTS

We acknowledge grants from the Gatsby Charitable Foundation, the Max Planck Hebrew University Center, the NIH, and the Human Frontier Science Program (Project RGP0015/2013). We thank an anonymous reviewer for his helpful comments.

Accepted: July 25, 2014
Published: August 21, 2014

REFERENCES

Albus, J.S. (1971). A theory of cerebellar function. Math. Biosci. *10*, 25–61.

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

Bell, A.J., and Sejnowski, T.J. (1997). The "independent components" of natural scenes are edge filters. Vision Res. *37*, 3327–3338.

Brecht, M., and Sakmann, B. (2002). Dynamic representation of whisker deflection by synaptic potentials in spiny stellate and pyramidal cells in the barrels and septa of layer 4 rat somatosensory cortex. J. Physiol. *543*, 49–70.

Caron, S.J.C., Ruta, V., Abbott, L.F., and Axel, R. (2013). Random convergence of olfactory inputs in the Drosophila mushroom body. Nature *497*, 113–117.

Chacron, M.J., Longtin, A., and Maler, L. (2011). Efficient computation via sparse coding in electrosensory neural networks. Curr. Opin. Neurobiol. *21*, 752–760.

Chadderton, P., Margrie, T.W., and Häusser, M. (2004). Integration of quanta in cerebellar granule cells during sensory processing. Nature *428*, 856–860.

Cortes, C., and Vapnik, V. (1995). Support-vector networks. Mach. Learn. *20*, 273–297.

Cover, T.M. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. IEEE Trans. Electron. Comput. *14*, 326–334.

Demmer, H., and Kloppenburg, P. (2009). Intrinsic membrane properties and inhibitory synaptic input of kenyon cells as mechanisms for sparse coding? J. Neurophysiol. *102*, 1538–1550.

DeWeese, M.R., Wehr, M., and Zador, A.M. (2003). Binary spiking in auditory cortex. J. Neurosci. *23*, 7940–7949.

Field, D.J. (1987). Relations between the statistics of natural images and the response properties of cortical cells. J. Opt. Soc. Am. A *4*, 2379–2394.

Field, D.J. (1994). What is the goal of sensory coding? Neural Comput. *6*, 559–601.

Galliano, E., Gao, Z., Schonewille, M., Todorov, B., Simons, E., Pop, A.S., D'Angelo, E., van den Maagdenberg, A.M.J.M., Hoebeek, F.E., and De

Zeeuw, C.I. (2013). Silencing the majority of cerebellar granule cells uncovers their essential role in motor learning and consolidation. Cell Reports *3*, 1239–1251.

Ganguli, S., and Sompolinsky, H. (2012). Compressed sensing, sparsity, and dimensionality in neuronal information processing and data analysis. Annu. Rev. Neurosci. *35*, 485–508.

Gardner, E. (1988). The space of interactions in neural network models. J. Phys. A: Math. Gen. *21*, 257–270.

Gütig, R., and Sompolinsky, H. (2006). The tempotron: a neuron that learns spike timing-based decisions. Nat. Neurosci. *9*, 420–428.

Haddad, R., Weiss, T., Khan, R., Nadler, B., Mandairon, N., Bensafi, M., Schneidman, E., and Sobel, N. (2010). Global features of neural activity in the olfactory system form a parallel code that predicts olfactory behavior and perception. J. Neurosci. *30*, 9017–9026.

Koulakov, A.A., and Rinberg, D. (2011). Sparse incomplete representations: a potential role of olfactory granule cells. Neuron *72*, 124–136.

Koulakov, A.A., Kolterman, B.E., Enikolopov, A.G., and Rinberg, D. (2011). In search of the structure of human olfactory space. Front. Syst. Neurosci. *5*, 65, http://dx.doi.org/10.3389/fnsys.2011.00065.

Marr, D. (1969). A theory of cerebellar cortex. J. Physiol. *202*, 437–470.

Miller, E.K., and Cohen, J.D. (2001). An integrative theory of prefrontal cortex function. Annu. Rev. Neurosci. *24*, 167–202.

Mombaerts, P., Wang, F., Dulac, C., Chao, S.K., Nemes, A., Mendelsohn, M., Edmondson, J., and Axel, R. (1996). Visualizing an olfactory sensory map. Cell *87*, 675–686.

Olshausen, B.A., and Field, D.J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature *381*, 607–609.

Olshausen, B.A., and Field, D.J. (2004). Sparse coding of sensory inputs. Curr. Opin. Neurobiol. *14*, 481–487.

Ölveczky, B.P., Andalman, A.S., and Fee, M.S. (2005). Vocal experimentation in the juvenile songbird requires a basal ganglia circuit. PLoS Biol. *3*, e153.

Poo, C., and Isaacson, J.S. (2009). Odor representations in olfactory cortex: "sparse" coding, global inhibition, and oscillations. Neuron *62*, 850–861.

Rozell, C.J., Johnson, D.H., Baraniuk, R.G., and Olshausen, B.A. (2008). Sparse coding via thresholding and local competition in neural circuits. Neural Comput. *20*, 2526–2563.

Sachdev, R.N.S., Krause, M.R., and Mazer, J.A. (2012). Surround suppression and sparse coding in visual and barrel cortices. Front. Neural Circuits *6*, 43.

Saxe, A., Bhand, M., Mudur, R., Suresh, B., and Ng, A.Y. (2011). Unsupervised learning models of primary cortical receptive fields and receptive field plasticity. In Advances in Neural Information Processing Systems 24, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K. Weinberger, eds. (La Jolla, CA: Neural Information Processing Systems), pp. 1971–1979.

Sheth, S.A., Abuelem, T., Gale, J.T., and Eskandar, E.N. (2011). Basal ganglia neurons dynamically facilitate exploration during associative learning. J. Neurosci. *31*, 4878–4885.

Stettler, D.D., and Axel, R. (2009). Representations of odor in the piriform cortex. Neuron *63*, 854–864.

Tsodyks, M.V., and Feigelman, M.V. (1988). The enhanced storage capacity in neural networks with low activity level. Europhys. Lett. *6*, 101–105.

Turner, G.C., Bazhenov, M., and Laurent, G. (2008). Olfactory representations by Drosophila mushroom body neurons. J. Neurophysiol. *99*, 734–746.

Valiant, L.G. (2012). The hippocampus as a stable memory allocator for cortex. Neural Comput. *24*, 2873–2899.

Vincis, R., Gschwend, O., Bhaukaurally, K., Beroud, J., and Carleton, A. (2012). Dense representation of natural odorants in the mouse olfactory bulb. Nat. Neurosci. *15*, 537–539.

Neuron, Volume *83*

Supplemental Information

Sparseness and Expansion in Sensory Representations

Baktash Babadi and Haim Sompolinsky

Supplemental Information

Supplemental figures

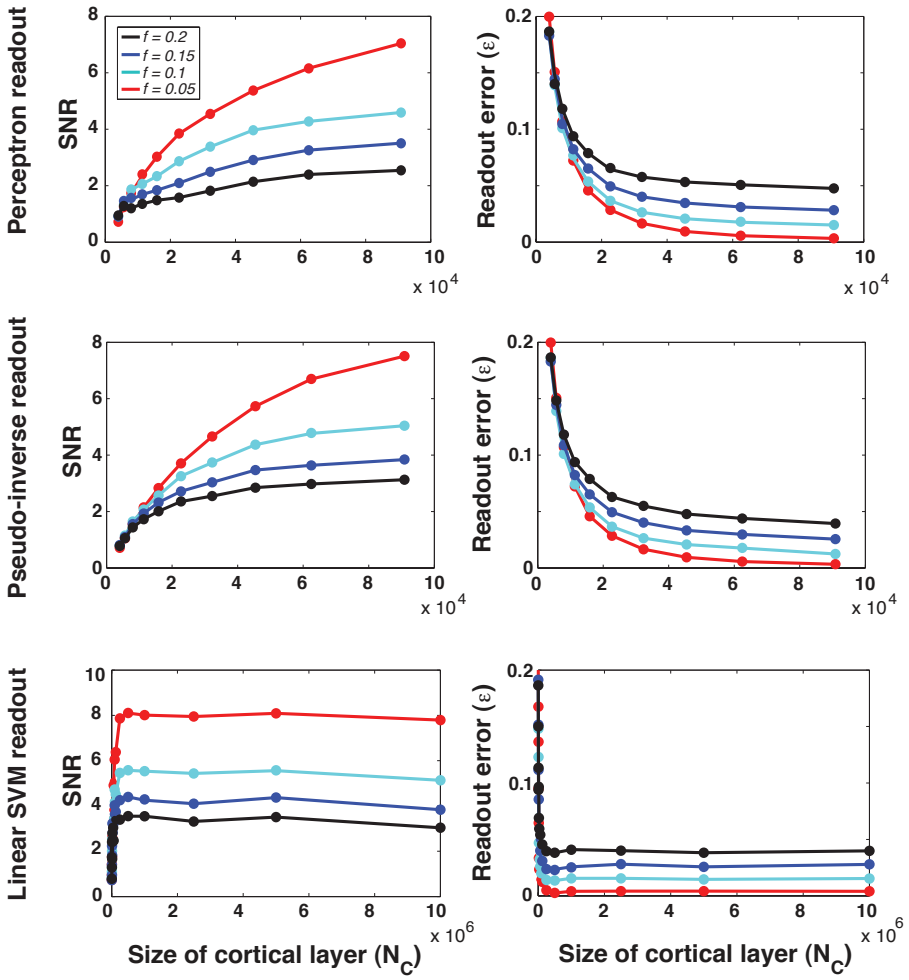

Figure S1

The effect of expansion on the performance of alternative readouts in the case of random feed-forward projections (compare to **Figure 4A-B**). The panels on the left show the SNR and the panels on the right show readout error, ε, as functions of the size of cortical layer, N_C. The top panels correspond to perceptron, the middle panels to pseudo-inverse and the bottom panels to linear SVM. Different curves correspond to different fractions of active cortical neurons, f. The stimulus cluster size is fixed at $\Delta S = 0.1$. Other parameters are $N_S = 1000$, $P = 1000$ and $N_C = 10000$. All curves and markers are obtained by numerical simulations. Note that the maximum N_C in the case of perceptron and pseudo-inverse (top and middle panels) is to 10^5, while in the case of SVM (bottom panels) it is 10^7. We checked the performance of SVM up to much larger values of N_C to ascertain that the readout performance indeed reaches a plateau in this elaborate readout scheme.

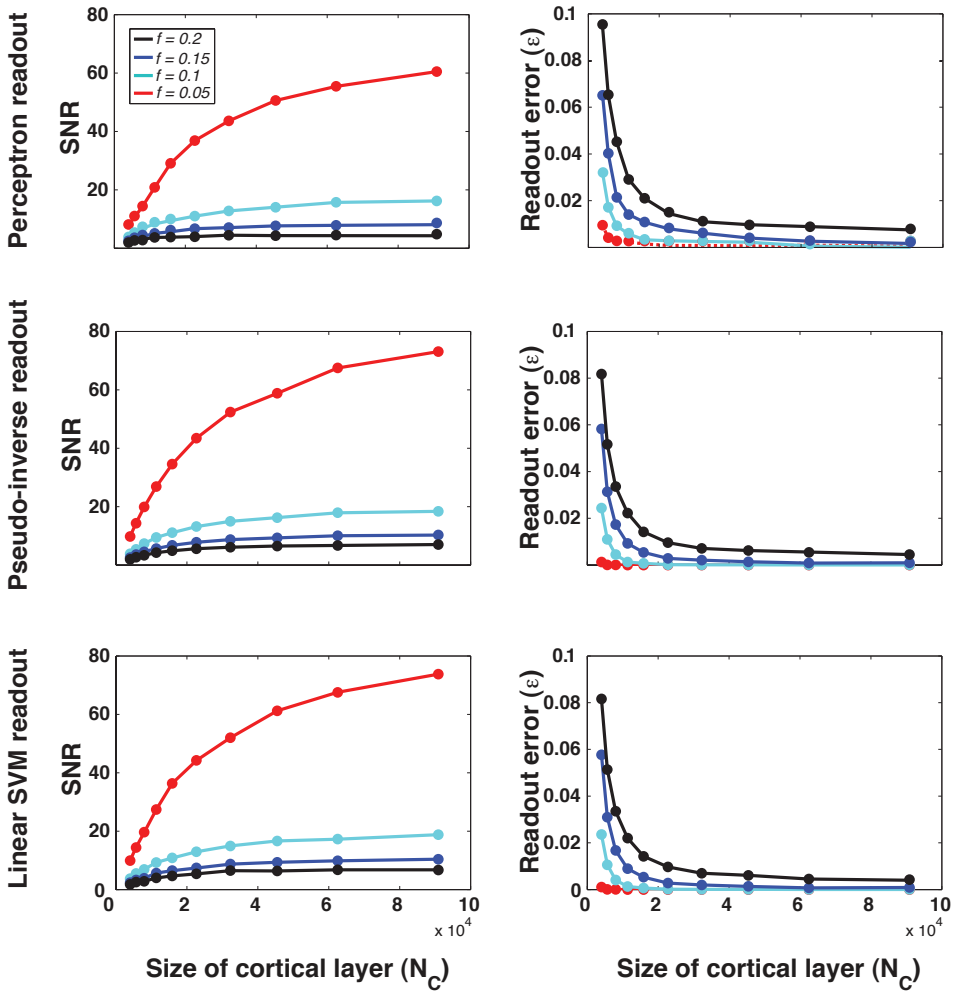

Figure S2

The effect of expansion on the performance of alternative readouts in the case of structured projections (compare to **Figure 7A-B**). The panels on the left show the SNR and the panels on the right show readout error, ε, as functions of the size of cortical layer, N_C. The top panels correspond to perceptron, the middle panels to pseudo-inverse and the bottom panels to linear SVM. Different curves correspond to different fractions of active cortical neurons, f. The stimulus cluster size is fixed at $\Delta S = 0.1$. Other parameters are $N_S = 1000$, $P = 1000$ and $N_C = 10000$. All curves and markers are obtained by numerical simulations.

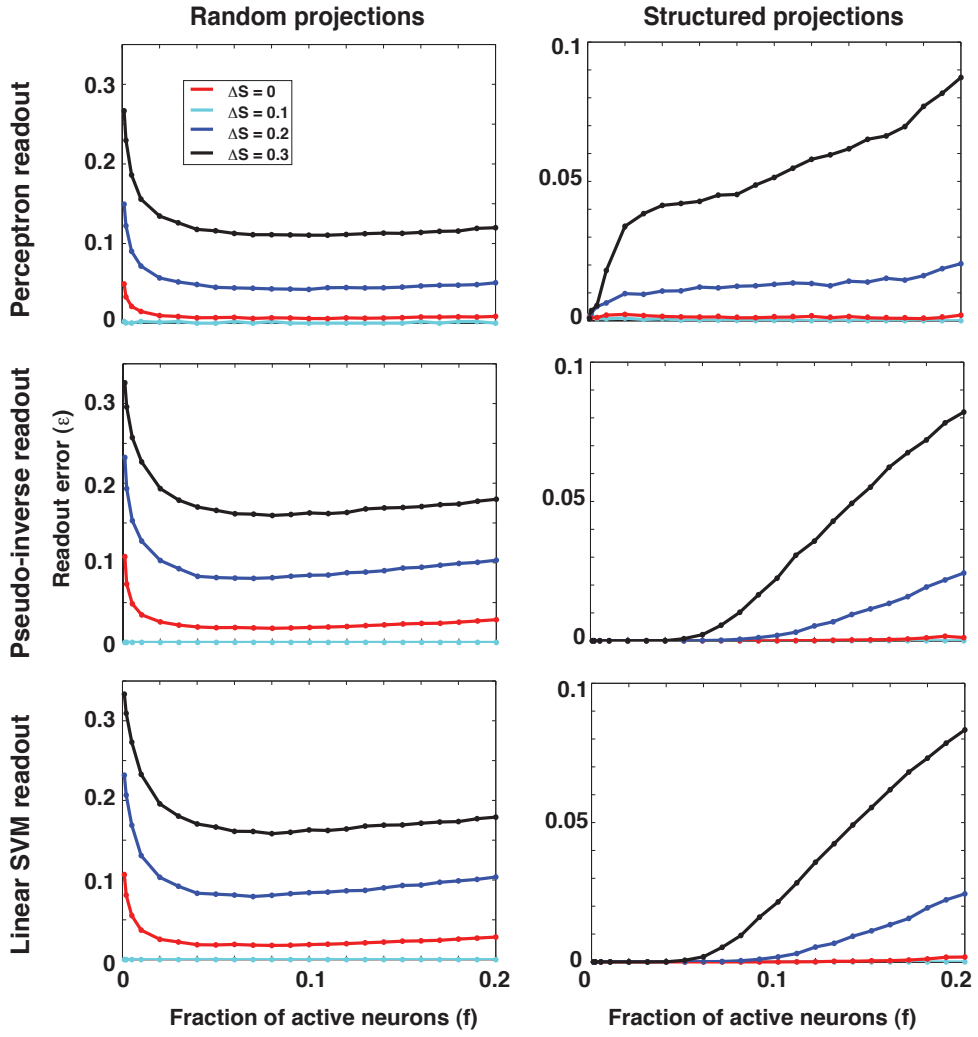

Figure S3

The effect of sparseness on the performance of alternative readouts. All panels show readout error, ε, as a function of the fraction of active cortical neurons, f. The panels on the left are for the case of random projections from stimulus to the cortical layer (compare to **Figure 4C**), and the panels on the right are for the case of structured projections from stimulus to the cortical layer (compare to **Figure 7C**). The top panels correspond to perceptron, the middle panels to pseudo-inverse and the bottom panels to linear SVM. Note that there is an optimal sparseness in the case of random projections for all readouts (left), while sparseness decreases the error monotonically in the case of structured projection for all of them. Different curves correspond to different stimulus cluster sizes, ΔS. Other parameters are $N_S = 1000$, $P = 1000$ and $N_C = 10000$. All curves and markers are obtained by numerical simulations.

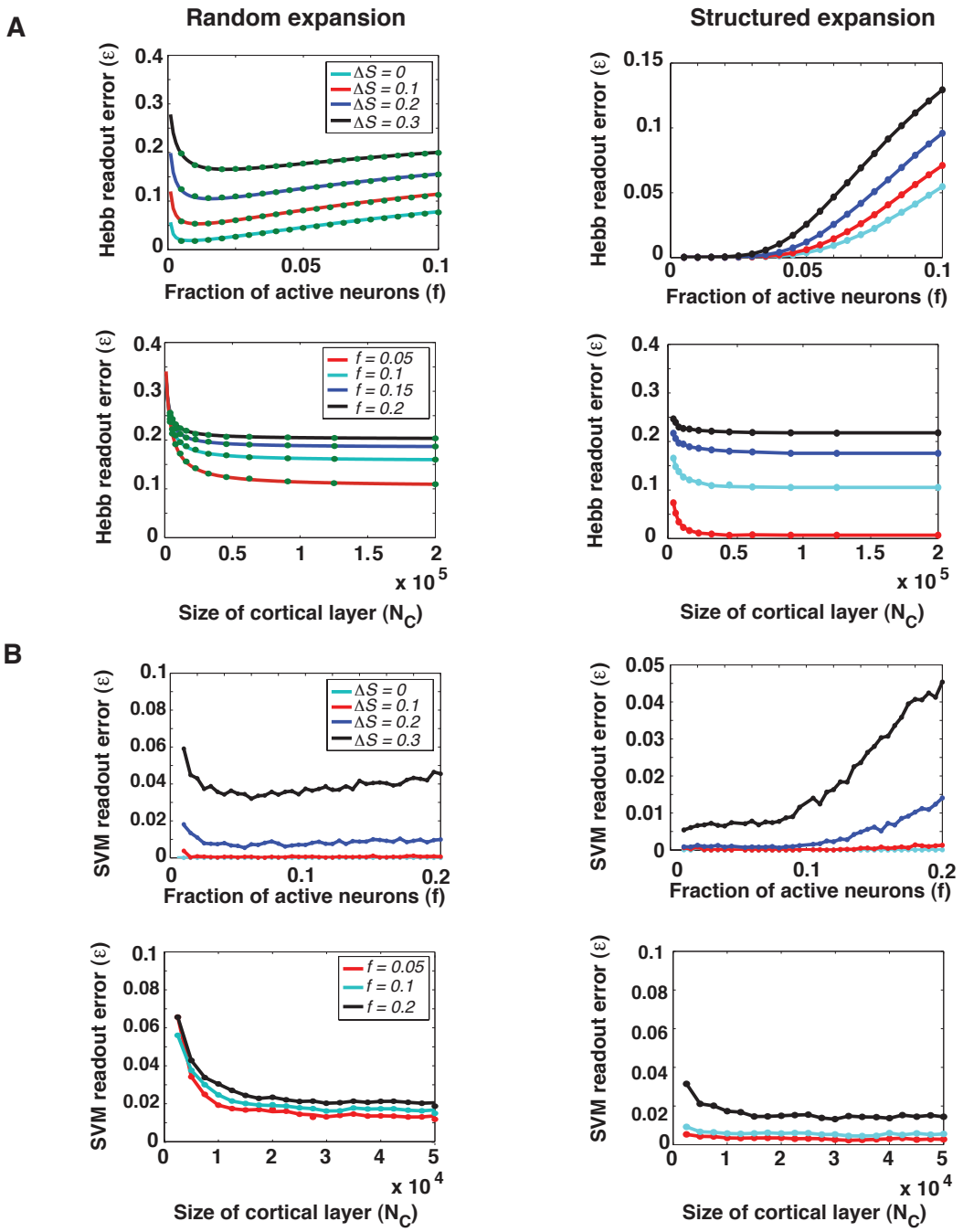

Figure S4
The effect of expansion and sparseness on the performance of the readouts trained with samples of typical patterns. Left and right columns correspond to random and structured expansion, respectively. **A)** The top row shows the error of the Hebb readout as a function of the fraction of active cortical neurons, f. Different curves correspond to different stimulus cluster sizes,

ΔS. The other parameters are $N_S = P = 1000$, $N_C = 10000$, $\Delta S_{learn} = 0.3$ and $N_{learn} = 100$. As in **Figure 4C**, there is an optimal activity level for which the readout error is minimized for random expansion (left), but not for structured expansion (right, **Figure 7C**). The bottom row shows the performance of the Hebb readout as a function of the size of cortical layer, N_C. Different curves correspond to different sparseness levels, f. The other parameters are $N_S = P = 1000$, $\Delta S = \Delta S_{learn} = 0.3$ and $N_{learn} = 100$. As in **Figures 4B & 7B**, the readout error reaches a plateau. **B)** The top row shows the error of the SVM readout as a function of the fraction of active cortical neurons, f. Different curves correspond to different stimulus cluster sizes, ΔS. The other parameters are $N_S = P = 50$, $N_C = 5000$, $\Delta S_{learn} = 0.3$ and $N_{learn} = 100$. As in **Figure S3**, there is an optimal activity level for which the readout error is minimized for random expansion (left), but not for structured expansion (right). The bottom row shows the performance of the linear SVM readout as a function of the size of cortical layer, N_C. Different curves correspond to different sparseness levels, f. The other parameters are $N_S = P = 50$, $\Delta S = \Delta S_{learn} = 0.3$ and $N_{learn} = 100$. As in **Figures S1 & S2**, the readout error reaches a plateau.

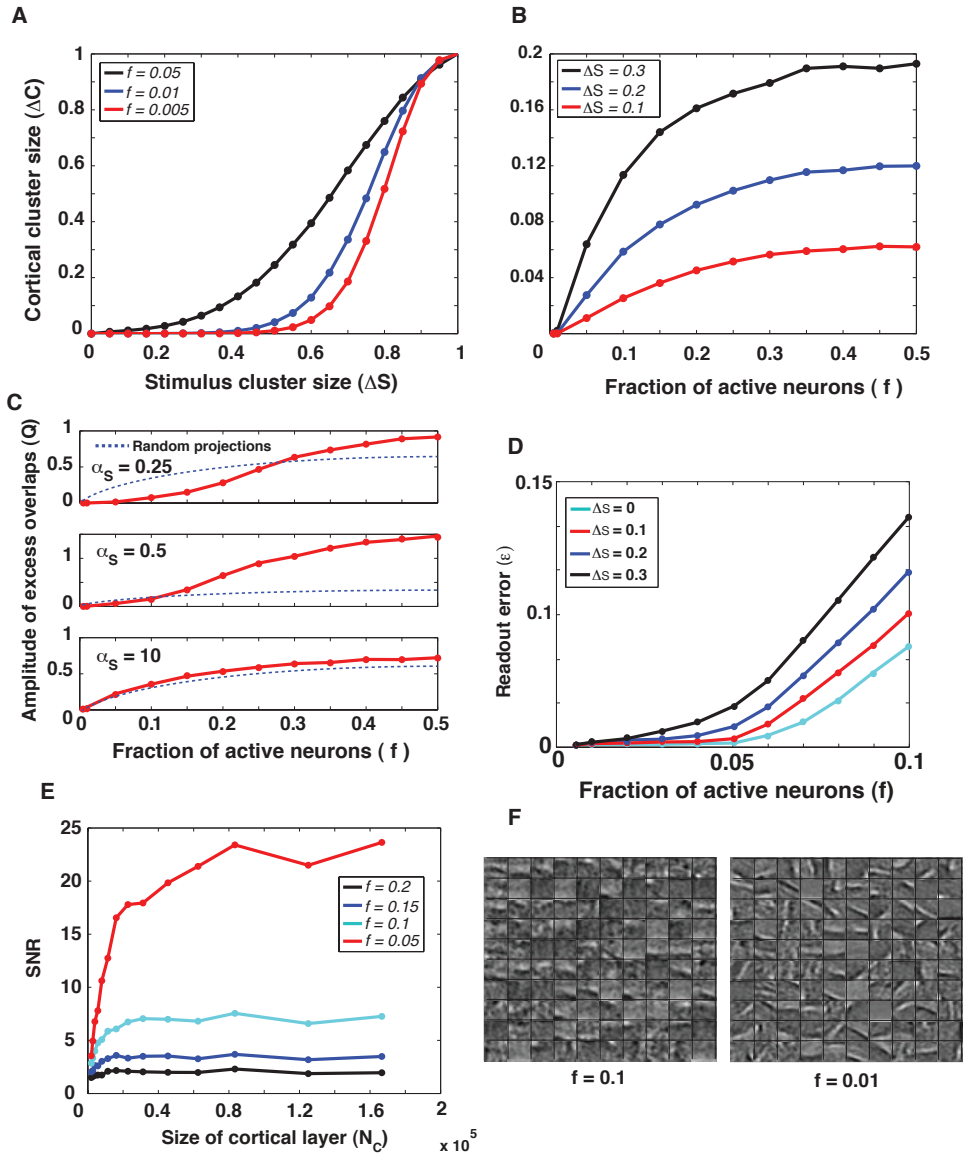

Figure S5

Online Hebbian learning of synaptic weights from stimulus to cortical layer. **A)** Shrinkage of the size of cortical clusters, ΔC, compared to that of the stimulus clusters, ΔS. Different curves correspond to different cortical sparseness levels, f. Other parameters are $N_S = 1000$, $N_C = 10000$ and $P = 1000$ (compare to **Figure 5A**). **B)** The size of cortical clusters, ΔC, increases as a function of the fraction f of active neurons in the cortical layer. Different curves correspond to different stimulus cluster sizes, ΔS. Other parameters are the same as in panel **A** (compare to **Figure 5B**). **C)** Red curves show the amplitude of excess overlaps in the case of online Hebbian learning as a function of the fraction of active cortical neurons, f (compare to **Figure 6**). Dotted blue curves show the amplitude of excess overlaps in the case of random projections for comparison. The number of cluster (P and hence α_S) increases from top to bottom. Other parameters are $N_S = 1000$ and $N_C = 10000$. **D)** The error of the Hebb readout is monotonically increasing as a function of the fraction of active cortical neurons, f (compare to **Figure 7C**). **E)** *SNR* of the readout increases as a function of the size of cortical layer, N_C, but in reaches a plateau (compare to **Figure 7A**). **F)** Application of the online Hebbian learning to patches of natural images. The patches, of size 16×16 pixels,

were taken from a Olshausen's data set of natural images. Each neuron in the stimulus layer corresponds to one pixel in the patch ($N_S = 256$). Shown are the resultant synaptic weights of 100 randomly chosen cortical neurons, in the same 16×16 configuration as the stimulus patches. The weights exhibit a Gabor-like structure, particularly when the cortical representation is highly sparse (right vs. left panel). There network is trained for 10240 patches, and the size of cortical layer is $N_C = 5000$. All curves, markers and plots are obtained by numerical simulations.

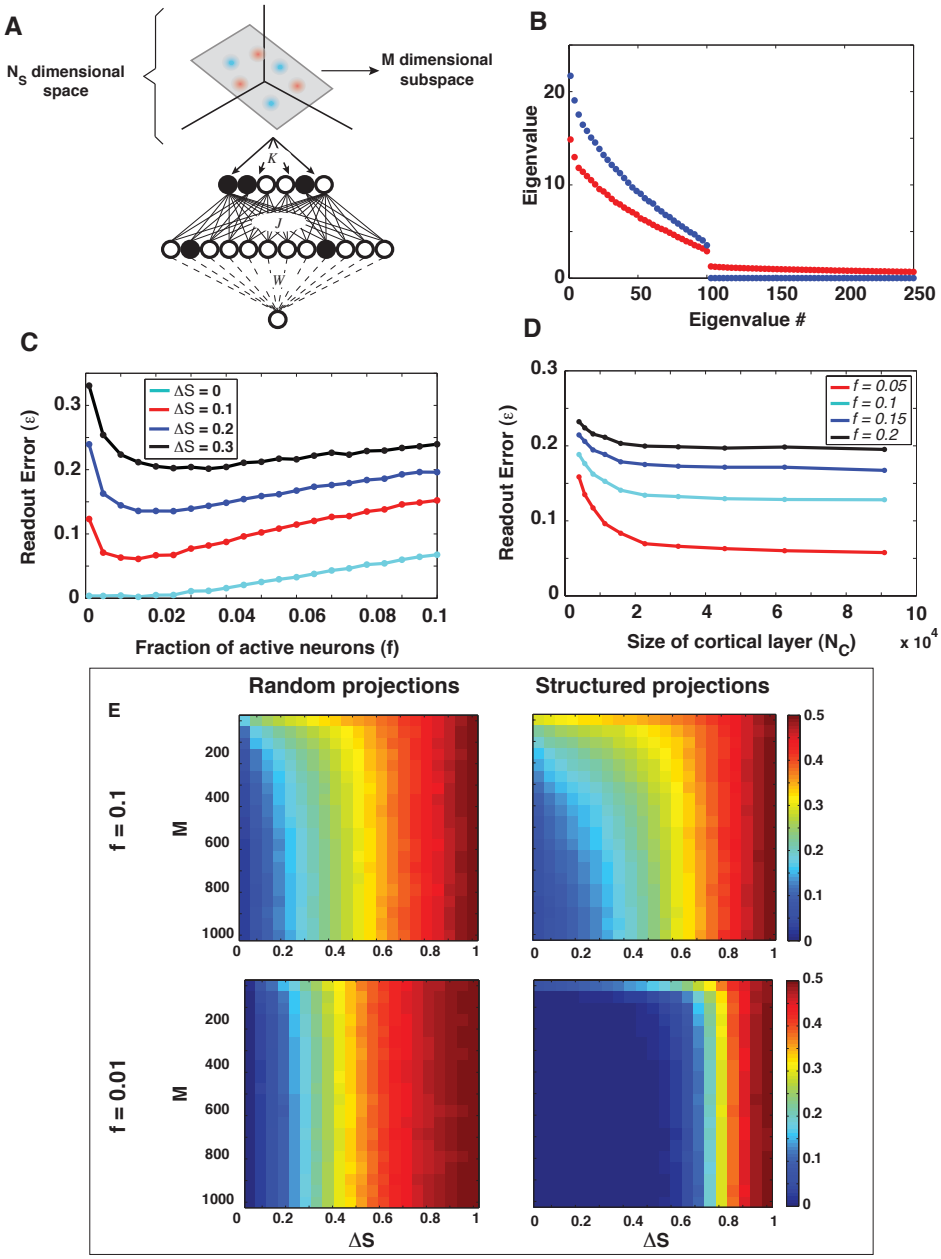

Figure S6
Sparse and expansive representation of stimuli with intrinsic low dimensionality. **A)** Schematic description of the stimulus dimensionality. The stimuli lie in a low (M) dimensional linear subspace of the N_S dimensional space spanned by the activity of neurons in stimulus layer. They are projected onto neurons of stimulus layer via random matrix K, and then are binarized. Through feed-forward synaptic projections J, patterns in the stimulus layer are mapped onto patterns in the cortical layer, and a downstream readout neuron learns the binary classification of the stimulus clusters through the synaptic weights W. **B)** Blue dots show the principal components of the projected stimuli to the neurons in stimulus layer before binarization, and red dots show the principal components of the binarized projected patterns in the stimulus layer. The parameters are $N_S = 1000$,

$M = 100$ and $P = 1000$. Binarization induces the residual components to the right of the gap after M-th component (red dots). **C)** Error of the Hebb readout for stimuli with low intrinsic dimension, as a function of the activity level of cortical layer, f. The projections from stimulus to cortical layer are random, resulting in an optimal sparseness (compare to **Figure 4C**). The parameters are $N_S = 1000$, $M = 200$ and $P = 1000$. **D)** Error of the Hebb readout for stimuli with low intrinsic dimension, as a function of the size of cortical layer, N_C . The projections from stimulus to cortical layer are random. Readout error reaches a plateau (compare to **Figure 4B**). **E)** Comparison of readout performance between random (left) and structured projections (right), when the fraction of active neurons is $f = 0.1$ (top row) and $f = 0.01$ (bottom row). Horizontal and vertical axes indicate the stimulus cluster size, ΔS, and the dimension of the subspace, M, respectively. The color scale shows the readout error. The other parameters are $N_S = P = 1000$ and $N_C = 500,000$. Note that as in **Figure 8**, the difference between the two modes of projections are more manifest when the cortical activity is highly sparse (bottom).

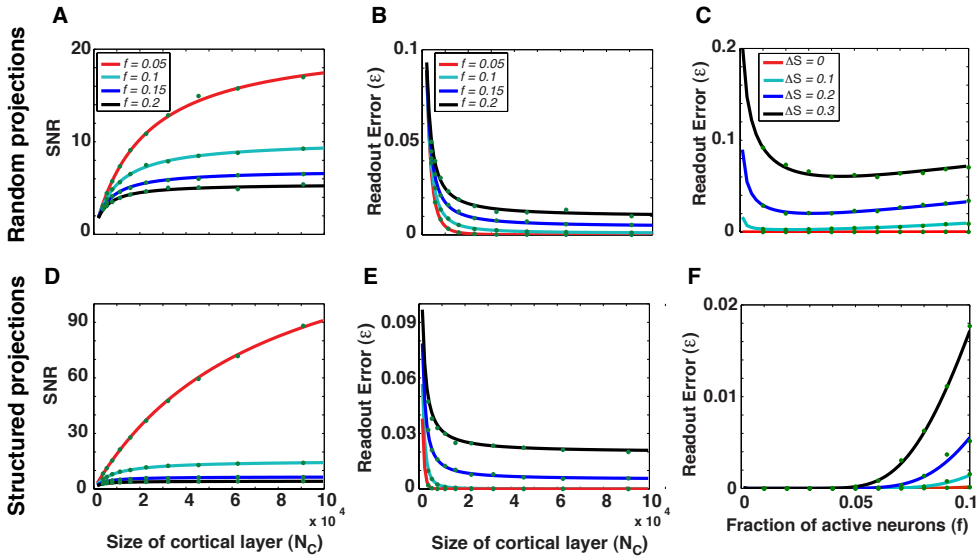

Figure S7

The effect of expansion and sparseness on readout performance with unequal probability of labels. The top row shows the results in the case of random projections (compare to **Figure 4**), and the bottom row shows the results for structured projections (compare to **Figure 7**). The probability l of the label being 1 is 0.1 in this case. All other parameters are as in **Figures 4 and 7**. The effect of sparseness and expansion are the same as the case of equal probability of labels (**Figures 4 and 7**). However, the sparseness of the labels increases the SNR and decreases the readout error. The curves show analytical results, and the markers show numerical simulations.

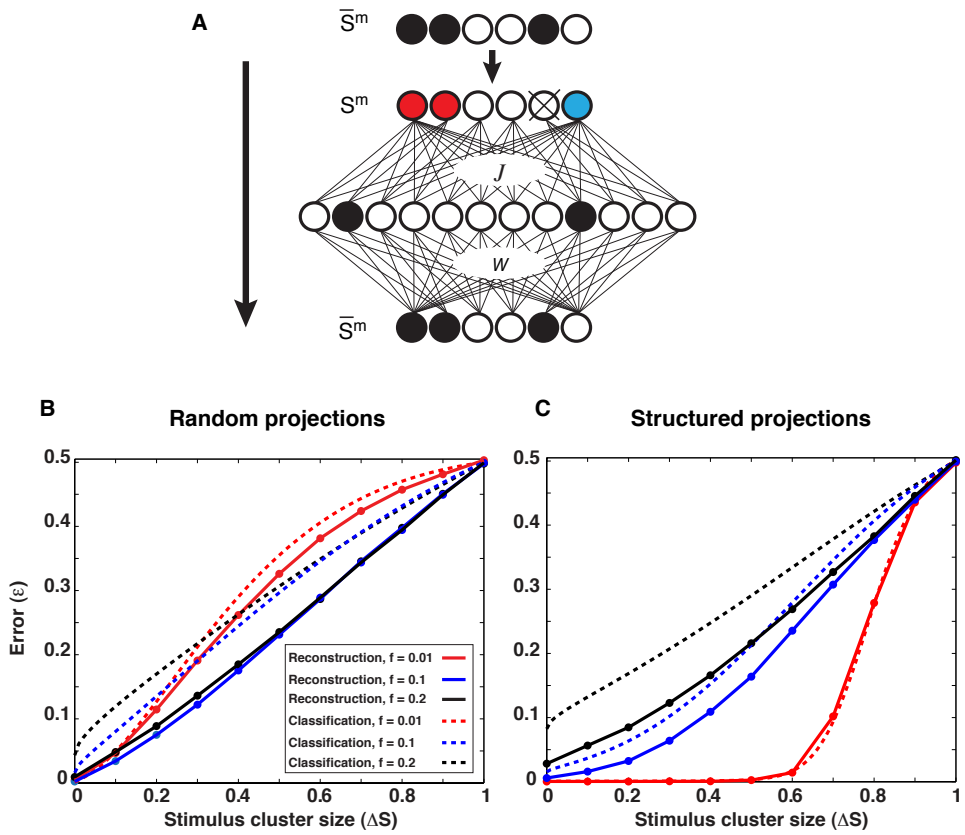

Figure S8
Performance of the readout in stimulus reconstruction task. **A)** Schematic description of the reconstruction task. The readout layer contains N_S neurons which are trained to retrieve cluster centers $\bar{S}^m$, from typical patterns S^m. **B)** Readout error in the case of random projections. Solid curves show the performance of the readout layer in stimulus reconstruction, and dotted curves show the performance of the readout in random binary classification task for comparison. Different colors correspond to different activity levels of cortical layer, f. Other parameters of the network are $N_S = 1000$, $N_C = 10000$, and $P = 1000$. **C)** Readout error in the case of structured projections. The parameters and color code are the same as in panel **A**.

Supplemental Experimental Procedures

Different linear readouts

For Perceptron readout, the synaptic weights are tuned by the simple Perceptron rule (Engel and Van den Broeck, 2001). However, if the Perceptron is trained using the central pattern of each cluster, it drastically fails to classify other members of the clusters correctly. In order to improve its performance, we train the Perceptron with "noisy" versions of central patterns of each cluster, i.e., during learning the stimulus patterns are drawn from clusters of the size $\Delta S_{learn} = 0.1$. Briefly, the readout weights W_j are initialized randomly. Stimulus patterns S^m are presented to the network sequentially. Each pattern S^m induces a corresponding pattern C^m in the cortical layer. If $\mathrm{sign}(\sum_{j=1}^{N_C} W_j\, C_j^m) = L^m$ the weights remain intact; otherwise, they are updated as $W_j \to W_j + \eta\, C_j^m\, L^m$, where $\eta = 0.001$ is the learning rate. An epoch of training comprises the presentation of one pattern form each of the P clusters. At the end of every epoch, readout error in classifying the central patterns of all clusters is evaluated (see below). Learning is continued until the average readout error stops decreasing in two consecutive blocks of 100 epochs.

The Pseudo-inverse readout weights (Engel and Van den Broeck, 2001) are defined as

$$W_j = \sum_{m,n=1}^{P} \left(U^{-1}\right)_{mn} (\bar{C}_j^n - f) L^m$$

where $U_{mn} = \sum_{j=1}^{N_C} (\bar{C}_j^m - f)(\bar{C}_j^n - f)$. The weights of the linear SVM readout are tuned according to the standard SVM training optimization problem without slack variables (Vapnik, 1998) using *svmtrain* function from Matlab Statistics Toolbox. In **Figures S1-S3** the cortical pattern/label pairs $((\bar{C}^m - f), L^m)$ are used as the training data. In **Figure S4B**, N_{learn} samples of typical stimulus patterns $S^{m,l}$ with a given cluster size $\Delta S_{learn} = 0.3$ are generated from each cluster m ($l = 1, \ldots, N_{learn}$). The $P \times N_{learn}$ pairs of corresponding cortical patterns/labels $(C^{m,l}, L^m)$ are used as the training data (see below).

Training the readout with samples of typical patterns

Instead of using the central pattern of each cluster to train the readout, it can be trained by N_{learn} samples from each cluster, drawn from typical patterns with a given cluster size ΔS_{learn}. We denote the l-th sample cortical pattern ($l = 1, \ldots, N_{learn}$) from cluster m, which is used in training of the readout, as $C^{m,l}$. In the Hebb scheme, the synaptic weight W_j from neuron j in the cortical layer to the readout neuron is given by:

$$W_j = \sum_{m=1}^{P} \sum_{l=1}^{N_{learn}} (C_j^{m,l} - f)\, L^m$$

In order to calculate the readout error in this case, we should first generalize our measure of cortical cluster size, such that it encompasses the distance between two arbitrary typical cortical patterns. Assume $S^{m,1}$ and $S^{m,2}$ to be two stimulus patterns from cluster m, with distances ΔS_1 and ΔS_2 from the central pattern $\bar{S}^m$, respectively. The distance between their corresponding cortical patterns $C^{m,1}$ and $C^{m,2}$ can be formulated as $\Delta C(\Delta S_1, \Delta S_2) = \langle \sum_{j=1}^{N_C} |C_j^{m,1} - C_j^{m,2}| \rangle / \left(2\, N_C\, f\, (1-f)\right)$. In the case of random projection, this distance is:

$$\Delta C(\Delta S_1, \Delta S_2) = \frac{1}{f(1-f)} \int_T^\infty dh\, \frac{\exp\left(-\frac{h^2}{2}\right)}{\sqrt{2\pi}}\, H\left(\frac{(1-\Delta S_1)(1-\Delta S_2)\, h - T}{\sqrt{1-(1-\Delta S_1)(1-\Delta S_2)}}\right)$$

The net input to the readout neuron in response to a pattern belonging to cluster m is $g^m = \sum_{j=1}^{N_C} W_j \left(C_j^m - f\right)$. If $g^m > 0$ the readout classifies the pattern as belonging to class 1, and otherwise belonging to class -1. Conditioned on the class label, the net input g^m has a mean:

$$\begin{aligned}\langle g^m \rangle &= \left\langle \sum_{j=1}^{N_C} W_j (C_j^m - f) \right\rangle = \left\langle \sum_{j=1}^{N_C} \sum_{n=1}^{P} \sum_{l=1}^{N_{learn}} (C_j^{n,l} - f)(C_j^m - f) L^n \right\rangle \\ &= N_C N_{learn} f(1-f) \left(1 - \Delta C(\Delta S, \Delta S_{learn})\right) L^m\end{aligned}$$

The averaging is over all P cluster center/label pairs $(C^{n,l}, L^n)$. With a similar averaging, the variance of the synaptic input to the readout is:

$$\begin{aligned}\sigma_g^2 &= \left\langle (g^m)^2 \right\rangle - \langle g^m \rangle^2 \approx N_C \left(N_{learn}^2 - N_{learn}\right) P\, f^2 (1-f)^2 \left(1 - \Delta C(\Delta S_{learn}, \Delta S_{learn})\right) \\ &\quad + N_C^2 \left(N_{learn}^2 - N_{learn}\right) P\, f^2 (1-f)^2 (1 - \Delta S_{learn})^2\, Q^2 / N_S \\ &\quad + N_C^2 N_{learn} P f^2 (1-f)^2 \\ &\quad + N_C^2 N_{learn} f^2 (1-f)^2 Q^2 / N_S\end{aligned}$$

Assuming that the synaptic input to the readout neuron obeys a normal distribution, the probability of miss-classification of pattern S^m is $\varepsilon = H(L^m \langle g^m \rangle / \sigma_g)$. The error ε calculated in this way appears as solid curves in **Figures S4A-left**. In analogy with ideal observer theory, the square of the argument of H represents the *SNR* of the system. In the limit of large number of samples ($N_{learn} \gg 1$), the *SNR* can be expressed as:

$$SNR = \frac{\left(1 - \Delta C(\Delta S, \Delta S_{learn})\right)^2}{\left(\alpha_C \left(1 - \Delta C(\Delta S_{learn,} \Delta S_{learn})\right) + \alpha_S (1 - \Delta S_{learn})^2\, Q^2\right)}$$

Note that in the case of $\Delta S_{learn} = 0$, the *SNR* of the readout trained with the cluster centers (equation 5) is retrieved, as expected. Training the readout with samples reduces both the signal (numerator) and the noise (denominator). Nevertheless, the qualitative behavior of the readout remains similar to the case of training with the central patterns only; namely, there is an optimal sparseness level that minimizes the readout error for random expansion (**Figure S4A-top left**), but not for structured expansion (**Figure 4SA-top right**), and the error saturates as the number of cortical neurons, N_C, increases (**Figure S4A-bottom**). The same qualitative behaviors are observed for linear SVM readout for a wide range of parameters (**Figure S4B**).

Stimuli with Intrinsic Low Dimensionality

In order to study the effect of expansion and sparseness on stimulus representations with intrinsic low dimensional structure, we generate N_S dimensional stimuli with intrinsic dimensionality M, with $M < N_S$, as follows. We first generate P randomly sampled M-dimensional real-valued vectors Y^m, $m = 1, ..., P$ and then construct $X^m = K^T Y^m$, where K is a fixed random $M \times N_S$ dimensional matrix. Finally, these real valued vectors are thresholded to yield the binary cluster centers $\bar{S}_i^m = \Theta(X_i^m)$ for all i and m **(Figure S6A)**. The resulting PCA spectrum of these vectors is shown in **Figure S6B**. As expected, most of the power is

concentrated in the first M components. The residual $N_S - M$ components result from the nonlinear, thresholding operation which slightly distorts the M-dimensional hyperplane. Noisy binary vectors are generated by flipping each component of $\bar{S}^m$ at random with a probability $\Delta S/2$. Alternatively, noise can be modeled by additive gaussian components in Y^m (not shown).

Evaluating the performance of the cortical layer on these stimuli, we find that the readout error increases when M is small, but the qualitative behavior is similar to the case of random stimuli described before. The degraded performance is due to an increase in the excess overlaps of the cortical patterns (equation 5), which now has an extra excess overlap contribution to equation 2, which scales as $1/M$. Thus for these stimuli, even when both N_S and N_C are large, the performance will asymptote to a nonzero error with a saturation size N_C proportional to M (**Figure S6D**, compare with equation 6). Nevertheless, expansion and sparseness considerably improve the performance, particularly with the structured projections (equation 8), where error decreases monotonically with decreasing f **(Figure S6E**-right). As before, random projections yield performance that is optimal at a finite value of f (example shown in **Figure S6C**).

Readout labels with unequal probability

In the Experimental Procedures, we analyzed the case where the probability of assigning either of the two labels to each cluster was equal. Here, we generalize the analysis to the case of unequal probability of the labels. We assume that the label L^m assigned to cluster m is 1 with probability l, and 0 with probability $(1-l)$. The synaptic weight W_j from neuron j in the cortical layer to the readout neuron is given by

$$W_j = \sum_{m=1}^{P} (\bar{C}_j^m - f)(L^m - l)$$

The net input to the readout neuron in response to a pattern belonging to cluster m is $g^m = \sum_{j=1}^{N_C} W_j \, (C_j^m - f)$. Conditioned on the class label, the net input g^m has a mean:

$$\begin{aligned} \langle g^m \rangle &= \left\langle \sum_{j=1}^{N_R} W_j (C_j^m - f) \right\rangle = \left\langle \sum_{j=1}^{N_R} \sum_{n=1}^{P} (\bar{C}_j^m - f)(C_j^m - f)(L^n - l) \right\rangle \\ &= N_C f(1-f)(1-\Delta C)(L^m - l) \end{aligned} \tag{1}$$

The averaging is over all P cluster center/label pairs $(\bar{C}^n, L^n)$. With a similar averaging, the variance of the synaptic input to the readout is:

$$\sigma_g^2 = \left\langle (g^m)^2 \right\rangle - \langle g^m \rangle^2 \approx l(1-1)\left(N_C \, P \, f^2 (1-f)^2 + N_C^2 \, P \, f^2 \, (1-f)^2 \, Q^2 / N_S\right)$$

We assume that the midpoint between $\langle g^m | m=1 \rangle$ and $\langle g^m | m=0 \rangle$ serves as a threshold T_r, such that if $g^m > T_r$ the readout classified the pattern as belonging to class 1, and otherwise as belonging to class 0. The midpoint can be calculated from equation 1 as:

$$T_r = \frac{\langle g^m | m=1 \rangle + \langle g^m | m=0 \rangle}{2} = \frac{N_C f(1-f)(1-\Delta C)(1-2l)}{2}$$

Assuming that the synaptic input to the readout neuron obeys a normal distribution, the probability of misclassification of pattern S^m belonging to class 1 is $\varepsilon = H\left(\left(\langle g^m | m=1 \rangle - T_r\right)/\sigma_g\right)$. The error ε calculated in this way appears as solid curves in **Figure S7.** The square of the argument of H represents the *SNR* of the system:

$$SNR = \frac{(1-\Delta C)^2}{4l(1-l)\left(\alpha_C + \alpha_S\, Q^2\right)}$$

Note that the sparseness of the label, l, decreases the noise (denominator) by a factor of $4l(1-l)$. Note also that in the case of the labels with equal probability, i.e, $l = 0.5$, the SNR in the Results section (equation 5) is retrieved.

Online Hebbian learning

The unsupervised Hebbian learning that we used as a model for structured synapse does not specify the mechanism of assigning random cortical patterns R^m to the corresponding stimuli. We have explored numerically a hybrid model in which J undergoes online Hebb plasticity with random initial values. The weight matrix J is calculated as follows. First, they are initialized randomly as $J_{ji} \sim \mathcal{N}(0, 2/\sqrt{N_S})$. At each subsequent step, one cluster m is chosen randomly, and a stimulus pattern pattern S^m is drawn from the cluster with size $\Delta S_{learn} = 0.1$. The activity of neuron j in the cortical layer in response to pattern S^m is $C_j^m = \Theta(\sum_{i=1}^{N_S} J_{ji}^t\, S_i^m - T)$, where J_{ji}^t is the current value of synaptic weights from neuron i in stimulus layer to neuron j in the cortical layer. The threshold T is tuned to set the fraction of active neurons in the cortical layer to be f. The synaptic weights are updated according to the following rule:

$$\Delta J_{ji} = \eta(S_i^m - \frac{1}{2})(C_j^m - f)\ , J_{ji}^{t+1} = \frac{J_{ji}^t + \Delta J_{ji}}{||J_{ji}^t + \Delta J_{ji}||} \tag{2}$$

where $\eta = 0.25/P$ is the learning rate. The error of the Hebbian readout in classifying the central patterns of clusters is evaluated after every 10000 steps. Learning is continued until the error remains unchanged. In numerical simulations this happens after roughly $1500 \times P$ steps for $N_S = 1000$, $N_C = 10000$ and $\Delta S_{learn} = 0.1$.

The performance of this model, shown in **Figure S5**, is intermediate between the two models analyzed in the Results section. The Hebb contribution to J generates a bimodal distributions of synaptic inputs to cortical neurons, hence the behavior of cluster sizes is similar to the structured weights. On the other hand, the excess overlap of the cortical patterns $\bar{C}^m$ is as large as in the random weights or even larger (**Figure S5C**). Thus, although the dependence of the readout error on f (compare **Figure S5D** with **Figure 4C** and **Figure 7C**) is similar to the structured model, the actual readout errors are in-between the two models.

Online Hebbian learning of natural images

We applied the above online Hebbian learning to patches of natural images from Olshausen's natural image database (Olshausen and Field, 1996) as the stimulus patterns (**Figure S5F**). Each patch (16×16 pixels, $N_S = 256$) was pre-whitened and the luminance values of pixels were normalized to lie between zero and one. These normalized luminances are used as S_i^m in Equation 2 above. There were 10240 patches, the size of cortical layer was $N_C = 5000$ and different activity levels (f) were used. The online learning was continued for 10^7 steps.

Supplemental References

Engel, A. and Van den Broeck, C. (2001). *Statistical Mechanics of Learning*. Cambridge University Press, 1 edition.

Olshausen, B. A. and Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609.

Vapnik, V. N. (1998). *Statistical learning theory*. Wiley, 1 edition.