Low Error Discrimination using a Correlated Population Code

Greg Schwartz*1, Jakob Macke*3,+, Dario Amodei2, Hanlin Tang2, Michael J Berry II1

*authors contributed equally

1Department of Molecular Biology and the Princeton Neuroscience Institute
2Department of Physics
Princeton University, Princeton, NJ 08544
3Max Planck Institute for Biological Cybernetics, Tübingen, Germany

current address: Department of Physiology and Biophysics, University of Washington, Seattle
+current address: Gatsby Computational Neuroscience Unit, University College London,
London, UK

current address: Biophysics Program, Harvard University, Cambridge, MA 02138

Abstract

We explored the manner in which spatial information is encoded by retinal ganglion cell populations. We flashed a set of 36 shape stimuli onto the tiger salamander retina and used different decoding algorithms to read out information from a population of 162 ganglion cells. We compared the discrimination performance of linear decoders, which ignore correlation induced by common stimulation, against nonlinear decoders, which can accurately model these correlations. Similar to previous studies, decoders that ignored correlation suffered only a modest drop in discrimination performance for groups of up to ~30 cells. However, for more realistic groups of 100+ cells, we found order-of-magnitude differences in the error rate. We also compared decoders that used only the presence of a single spike from each cell against more complex decoders that included information from multiple spike counts and multiple time bins. More complex decoders substantially outperformed simpler decoders, showing the importance of spike timing information. Particularly effective was the first spike latency representation, which allowed zero discrimination errors for the majority of shape stimuli. Furthermore, the performance of nonlinear decoders showed even greater enhancement compared to linear decoders for these complex representations. Finally, decoders that approximated the correlation structure in the population by matching all pairwise correlations with a maximum entropy model fit to all 162 neurons were quite successful, especially for the spike latency representation. Together, these results suggest a picture in which linear decoders allow a coarse categorization of shape stimuli, while nonlinear decoders, which take advantage of both correlation and spike timing, are needed to achieve high-
INTRODUCTION
Throughout the brain, neural circuits use large populations of neurons to encode and transmit information. Because much of what we know about neural codes comes from experiments and analysis of single neurons, a key issue in understanding population codes is the pattern and strength of correlation among neurons (Averbeck et al. 2006). In a wide range of neural systems, correlations between pairs of neurons have modest values, often changing the joint probability of two cells firing by < 10% compared to chance (Usrey and Reid 1999). Similarly, the information encoded by pairs of neurons is close to that for independent neurons in the retina (Puchalla et al. 2005), V1 (Gawne et al. 1996), S1 (Petersen et al. 2001), and motor cortex (Narayanan et al. 2005). However, even weak pairwise correlation can introduce more dramatic departures from independence for groups of 7 or more neurons (Gauthier et al. 2009; Schaub and Schultz 2012; Schneidman et al. 2006; Yu et al. 2011), and the quadratic growth in the number of pairs as the number of neurons increases suggests that correlation could have a more substantial effect on reading out information from larger populations (Pillow et al. 2008; Schaub and Schultz 2012; Tkachik et al. 2006).

As the direct estimation of mutual information is typically not tractable for large neural populations, we instead employed a decoding approach that involved building different classes of decoders to read out the information encoded by the large population. The discrimination performance of any decoder always establishes a lower bound on the information encoded by the neural population, and by comparing different, successively more complex and complete decoders, one can gain insight into the population neural code. In many cases, the simplest of all possible algorithms – the linear decoder – achieves quite high performance, both for continuous estimation (Bialek et al. 1991; Stanley et al. 1999; Warland et al. 1997) and for discrete categorization (Hung et al. 2005). However, linear decoders are not guaranteed to be able to extract all the available information from a correlated neural population (Schaub and Schultz 2012).
these patterns of correlations are important for the population code, then nonlinear decoders will significantly outperform linear decoders.

Here, we used this approach to study how spatial information was represented by a population of 162 ganglion cells in the salamander retina. We flashed 1 of 36 shapes onto the retina and asked how the neural population allowed discrimination between one of the shapes and all of the rest. Different classes of decoders were constructed that either included or ignored the patterns of correlation induced by common stimulation (signal correlation). In addition, we compared decoders that represented the response of each ganglion cell as spike/no-spike in a single time bin versus decoders that used multiple spikes and spike timing information. We found that this ganglion cell population supported high-fidelity discriminations that reached the limit of zero errors during the entire experiment. This performance required large populations (100+ cells) as well as nonlinear decoders that included the effects of signal correlation. Decoders that included spike timing information performed dramatically better than simpler decoders, but only when correlation was also included. Furthermore, we constructed decoders based only on the pattern of pairwise correlation among neurons rather than requiring knowledge of the stimulus tuning properties of all neurons. These maximum entropy decoders had high performance, reaching zero discrimination errors for many shape stimuli.

METHODS

Recording. Retinal ganglion cells from tiger salamander retina were recorded extracellularly using a multi-electrode array. In brief, animals were sacrificed in accordance with institutional guidelines, eyes were removed, and then retinas were dissected out of the eye, trimmed to ~2x2 mm pieces, and placed on a planar multi-electrode array. The details of the recording procedure are described elsewhere (Segev et al. 2004). Cells were selected if no spike-sorting errors were evident looking at template fits to the raw spike waveforms by hand. No firing rate criteria were used for cell selection.
Stimuli. We presented 38 different shapes, one-at-a-time, for 500 ms followed by 1 s grey screen (Fig. 1A). The 38 shapes were each presented 150 times, and trials were randomly interleaved. In order to avoid non-stationarity in the data, we analyzed only trials 26-125 for each shape. Most of the shapes (36 out of 38) contained 400 black pixels covering a total area of 48,400 μm². The other two shapes consisted of a small square, covering 12,100 μm² and a large square, covering 108,900 μm². For most analyses, we only considered the 36 shapes with equal area.

Receptive Field Analysis. For each ganglion cell, we measured the receptive field using reverse correlation to random flicker with 55 μm squares. We found the 1-sigma ellipse from a Gaussian fit to the receptive field center mechanism. Coverage was defined as the number of ganglion cells with their 1-sigma ellipse covering a single point in visual space (see Segev et al., J. Neurophysiology, 2006 for more detail).

Mutual Information about Shape Identity. If the number of spikes exceeded a threshold $n_{\text{max}}$, the extra spikes were discarded, and the neuron was said to have fired exactly $n_{\text{max}}$ spikes. We varied the value of $n_{\text{max}}$ from 1 to 5. We defined the response of each cell $i$ as $r_i = \left[0, n_{\text{max}} \right]$, which was the number of spikes fired in a single time bin, often the bin from 100 – 200 ms after shape onset. The mutual information that the responses of cell $i$ contained about the set of 36 stimuli was calculated by the formula:

$$I(r_i; S) = -\sum_{r_i=0}^{n_{\text{max}}} p(r_i) \log_2 p(r_i) + \frac{1}{L} \sum_{k=1}^{L} \sum_{r_i=0}^{n_{\text{max}}} p(r_i | s_k) \log_2 p(r_i | s_k),$$

where $p(r_i | s_k)$ is the probability that cell $i$ fired $r_i$ spikes in response to shape $k$, and $p(r_i)$ is the average of this quantity across all $L$ shapes. We used standard bias correction techniques to estimate entropies (Brivanlou et al. 1998; Puchalla et al. 2005).
More Complex Response Representations. We explored nine different choices of more complex response representations (Fig. 8A):

1. **Optimal threshold**: A binary response representation, where the threshold for the ‘1’ symbol was independently chosen for each cell in order to maximize the information that cell individually conveyed about shape identity. For cells that responded with many spikes, a threshold greater than 1 spike might convey more shape information.

2. **Burst**: A multi-symbol representation where the response symbols could be ‘0’, ‘1’, or ‘2’, corresponding to 0, 1, or 2+ spikes in the first time bin, which occurred 100-200 ms after stimulus onset.

3. **Multi-spike**: A multi-symbol representation where response symbols could be in the range [0,5] for up to $n_{max} = 5$ spikes in the first time bin.

4. **2 $\otimes$ 50 ms**: A multi-time bin representation where we separately measured spike or no-spike in each of the first two 50 ms time bins and concatenated the total response together into a 2-digit binary word. This description allowed a spiking response to be ‘absent’, ‘early’, ‘late’, or ‘both’.

5. **4 $\otimes$ 25 ms**: A multi-time bin representation where we measured spike or no-spike in each of the first four 25 ms time bins.

6. **Second bin**: A binary representation for the second 100 ms time bin, which occurred 200-300 ms after the onset of the shape stimulus and included the beginning of the sustained response.

7. **First 200 ms**: A binary representation where we measured spike or no-spike in a time bin 100-300 ms after stimulus onset.

8. **2 $\otimes$ 100 ms**: A multi-time bin representation where we measured spike or no-spike in each of the first two 100 ms time bins.

Spike Latency Representation. The spike latency function for each cell, $p(t_i | r_i = 1, s_k)$, was estimated by taking the first spike in all trials with one or more spikes from cell $i$,
convolving this with a Gaussian kernel with width 10 ms and area one, and summing across all spikes, and dividing by the probability of getting a spike, \( p(r_i | s_k) \). To ensure that the estimated spike-density was nonzero in our time interval of interest, we added a constant function of area one to the histogram and normalized by dividing by its total area, namely \( N_{\text{spikes}} + 1 \), where \( N_{\text{spikes}} \) is the total number of non-silent binary events ('spikes') sampled during the entire recording. Thus, cells that produced few spikes had a nearly constant spike latency function. Noise and uncertainty in the time of stimulus onset was modeled by adding a random time jitter to all the spike times, chosen from a Gaussian distribution with width \( \Delta \). All spikes had the same jitter on a given trial; this jitter varied from trial-to-trial. Jittered times were used both to train and test the decoder.

**Connection between Independent and Linear Decoders.** Equation 13 describes the probability distributions underlying the independent decoder. For the binary representation, we can simplify these expressions: \( p_i \equiv p(r_i = 1 | T) \) for the target shape and \( q_i \equiv p(r_i = 1 | D) \) for the distracter ensemble, which in turn imply \( p(r_i = 0 | T) = 1 - p_i \) and \( p(r_i = 0 | D) = 1 - q_i \). Given these definitions, the probability distributions can be written as:

\[
P(R | T) = \prod_{i}^{\text{cells}} (p_i)^r_i (1 - p_i)^{1-r_i} = \prod_{i}^{\text{cells}} (1 - p_i) \prod_{i}^{\text{cells}} \left( \frac{p_i}{1 - p_i} \right)^{r_i},
\]

with an analogous expression for \( P(R | D) \). Recall that the decoding rule requires \( P(T | R) / P(D | R) \geq \theta \) in order to classify the population response \( R \) as coming from the target stimulus. We can take the logarithm of both sides of the equation and use Bayes’ Rule (Eqn. 12):

\[
\log \left[ \frac{P(T | R)}{P(D | R)} \right] = \sum_{i}^{\text{cells}} r_i \log \left( \frac{p_i}{1 - p_i} \right) - \sum_{i}^{\text{cells}} r_i \log \left( \frac{q_i}{1 - q_i} \right) + \Omega,
\]
where the constant $\Omega = \log P(T) - \log P(D) + \sum_i \log (1 - p_i) - \sum_i \log (1 - q_i)$ does not depend on any of the neural responses $\{r_i\}$. The decoding rule can now be rewritten as:

$$
\sum_i^{\text{cells}} w_i r_i \geq \Theta, \quad \text{where} \quad w_i = \log[p_i(1 - q_i)] - \log[q_i(1 - p_i)]
$$

and $\Theta$ is a new constant. From Eqn. 4 it can be readily be seen that this is a linear decoder with a particular choice of the weights, $\{w_i\}$, that reflect the firing rate of each cell in response to the target stimulus, $\{p_i\}$, and the distracter ensemble, $\{q_i\}$.

**Optimal Linear Decoder.** We were interested in finding the best possible weights $\{w_i\}$ for a linear decision rule of the form $\sum_i^{\text{cells}} w_i r_i \geq \Theta$, where $\{r_1, r_2, \ldots, r_N\}$ is the vector of neural responses and $\Theta$ is an optimized threshold. Specifically, we denote by $\tilde{R}^{(k)} = \left(r_1^{(k)}, r_2^{(k)}, \ldots, r_N^{(k)}\right)$ the population response to the $k^{\text{th}}$ presentation of the stimulus, where $r_i^{(k)} = 1$ if the $i^{\text{th}}$ neuron in the population fired one or more spikes, and $r_i^{(k)} = 0$ otherwise. The trial indices $k$ run from 1 to $M$, where $M$ is the total number of trials in the experiment. We are interested in finding a linear decoder, i.e. a decoder of the form $f(\tilde{R}) = \sum_i^{\text{cells}} w_i r_i - \Theta$. A popular method for finding the weights $\{w_i\}$ is logistic regression (Hastie et al. 2005), which finds weights by optimizing the log-likelihood of the data under a logistic model. If the population size $N$ is big, it is advisable to use a regularized variant, such as L2-regularized logistic regression. In this case, the weight-vector $\tilde{w} = (w_1, w_2, \ldots, w_N)$ is found by minimizing the loss function:

$$
L(\tilde{w}) = \sum_{k=1}^{M} F\left(f(\tilde{R}^{(k)}), y^{(k)}\right) + C \sum_{i=1}^{N} w_i^2,
$$

where $F\left(f(\tilde{R}^{(k)}), y^{(k)}\right) = \log\left[1 + \exp\left(-y^{(k)} f(\tilde{R}^{(k)})\right)\right]$. (5)
Here, the label \( y^{(k)} = 1 \) if trial \( k \) was a target trial, and \( y^{(k)} = -1 \) otherwise. The first term ensures that the predicted labels \( f(\tilde{R}^{(k)}) \) are close to the actual labels \( y^{(k)} \). The second term is a regularization term, which is used to protect against over-fitting when the dimensionality of \( \tilde{w} \) is high. Regularization is necessary to achieve good generalization performance for large population sizes \( N \).

To ensure that the solution of logistic regression is at least as good as the independent model (which is also a linear decoder), we used a modified regularization term of the form \( C \sum_{i=1}^{N} (w_i - \omega_i)^2 \), where \( \{\omega_i\} \) are the weights of a linear decoder constructed from the independent model, i.e. \( \omega_i = \log[p_i(1-q_i)] - \log[q_i(1-p_i)] \) (Eqn. 4). Here, \( p_i \) is the probability of a spike in cell \( i \) in response to the target, and \( q_i \) the probability of a spike in response to the distracter ensemble.

As the loss function \( L(\tilde{w}) \) is convex, its global minimum can be found by gradient descent. We optimized the weights \( \{w_i\} \) over different choices of the parameter \( C \), and used the weights that resulted in best performance. After fitting, the offset \( \Theta \) was chosen such that the decoder had a hit rate of 99%.

**Tempotron.** The tempotron is a decoder that uses spike timing combined linearly at the readout stage. Implementation of the tempotron was based on Gutig and Sompolinsky (2006). Input patterns consisted of either zero spikes or the first spike from 100 ms to 300 ms after shape onset. Parameters were as follows: \( \tau \) (time constant of synaptic input) = 20 ms; \( T \) (duration of signal) = 500 ms; \( \lambda \) (learning increment) = \( 1 \times 10^{-4} \); \( \mu \) (momentum term) = 0.99; \( N = 162 \) synapses. Because our distracter set (100 trials for each of 35 shapes) was much larger than our test set (100 trials for one shape), the learning rule was dominated by inhibition. Therefore, we adjusted the \( \lambda \) parameter to \( \lambda/35 \) for distracter stimuli, while it remained at \( \lambda \) for target stimuli. For each shape, the
Tempotron was trained on the entire data set for 500 passes and tested on the same data set. Error rates reported are the lowest false alarm rate found in the 500 passes given a hit rate threshold of 99%. Runs on a subset of shapes with larger or smaller values of $\tau$ (10 ms – 200 ms) and $\lambda$ (1 x 10^{-3} to 1 x 10^{-5}) yielded similar or inferior results. Longer runs (2000 passes) also showed no improvement for the subset of shapes tested.

**Maximum Entropy Model.** We fit a probabilistic model $P(\vec{R})$ to experimentally measured population responses. In our case, the responses were given by a data set $U_{\text{emp}} = \{\vec{R}_{1}, \vec{R}_{2}, \ldots, \vec{R}_{M}\}$, where $\vec{R}^{(k)}$ is an $N$-dimensional vector of binary responses from $N$ neurons. We require $P(\vec{R})$ to respect the empirically estimated first and second order moments, *i.e.* mean firing rates and pairwise correlations. In addition, we want $P(\vec{R})$ to be as unstructured as possible, in the sense that its higher-order correlations are minimal. This can be achieved by searching for the model $P(\vec{R})$, which has maximum entropy (Jaynes 1957; 1978).

The maximum entropy distribution $P(\vec{R})$ is given by the solution to a convex, constrained optimization problem (Jaynes 1978; Kunkin and Frisch 1969; Schneidman et al. 2003). This optimization problem has a unique solution, and the solution is in the exponential family. In our specific case, it is of the form:

$$P_{\lambda}(\vec{R}) = \frac{1}{Z} \exp \left( \sum_{i} h_{i} r_{i} + \frac{1}{2} \sum_{i,j} J_{ij} r_{i} r_{j} \right) \quad (6)$$

This implies that the maximum entropy model is mathematically identical to the Ising model from statistical physics at a fixed temperature (Schneidman et al. 2006; Schneidman et al. 2003).
For notational convenience, we replace the biases \( \{ h_i \} \) and the couplings \( \{ J_{ij} \} \) by a parameter-vector \( \vec{\lambda} = \{ h_1, h_2, \ldots, h_N, J_{12}, J_{13}, \ldots, J_{N-1,N} \} \) and define a feature map \( \vec{F}(\vec{R}) = \{ r_1, r_2, \ldots, r_N, r_{12}, r_{13}, \ldots, r_{N-1,N} \} \). With this new notation, \( P_\lambda(\vec{R}) \) is given by

\[
P_\lambda(\vec{R}) = \frac{1}{Z} \exp \left( \sum \lambda_i f_i(\vec{R}) \right) = \frac{1}{Z} \exp \left( \vec{\lambda} \cdot \vec{F}(\vec{R}) \right),
\]
and the log-likelihood of the data under the model by

\[
L(\vec{\lambda}, U_{\text{emp}}) = \sum_{\vec{R} \in U_{\text{emp}}} \log P(\vec{R}).
\]

Maximizing the entropy subject to constraints is the equivalent to a form of maximum-likelihood fitting. Thus, the optimal parameters can be identified by searching for the parameters that maximize the log-likelihood, \( L(\vec{\lambda}, U_{\text{emp}}) \). It is known that this optimization problem is convex, and we are therefore guaranteed that there is a unique solution, and that there are no local minima in which the search algorithm could get stuck.

The computational challenges largely arise from the fact that, in principle, computation of the partition function \( Z \) requires summation across the entire state-space \( \{0,1\}^N \). If \( N \) is small enough for this summation to be feasible, an algorithm called iterative scaling can be used to find the optimal parameters. We perform a gradient descent on the cost function:

\[
C = -\frac{1}{M} L(\vec{\lambda}, U_{\text{emp}}) = \log(Z) - \frac{1}{M} \sum_{\vec{R} \in U_{\text{emp}}} \vec{\lambda} \cdot \vec{F}(\vec{R}) = \log(Z) - \vec{\lambda} \cdot \langle \vec{F} \rangle_{\text{emp}},
\]
where we have defined the vector of empirical feature means \( \langle \vec{F} \rangle_{\text{emp}} = \frac{1}{M} \sum_{\vec{R} \in U_{\text{emp}}} \vec{F}(\vec{R}) \).

The gradient of the loss function with respect to the \( i^{th} \) coordinate is given by (Broderick et al. 2007):

\[
\frac{\partial C}{\partial \lambda_i} = \frac{1}{Z} \frac{\partial Z}{\partial \lambda_i} - \frac{1}{M} \sum_{\vec{R} \in U_{\text{emp}}} F_i(\vec{R}) = \langle F_i(\vec{R}) \rangle_{\lambda_i} - \langle F_i(\vec{R}) \rangle_{\text{emp}},
\]
i.e. the difference between the desired mean \( \langle F_i(\vec{R}) \rangle_{\text{emp}} \) and the mean of the model with our current parameters \( \vec{\lambda} \).
Changing the parameter $\lambda$ from its current value $\lambda$ to a new value $\lambda' = \lambda + \delta$ leads to a reduction $\Delta L = L_{\lambda'} - L_{\lambda} = \log \left( \frac{Z_{\lambda'}}{Z_{\lambda}} \right) + \delta \circ \langle \bar{F}(\bar{R}) \rangle_{emp}$. It is convenient to only update one dimension of $\lambda$ at a time, i.e. to choose $\delta$ to only have one non-zero entry $\delta_i$. It can be shown (Broderick et al. 2007) that the optimal step-length is:

$$\delta_i = \log \left( \frac{\langle F_i(\bar{R}) \rangle_{emp} \left(1 - \langle F_i(\bar{R}) \rangle_{\lambda} \right)}{\langle F_i(\bar{R}) \rangle_{\lambda} \left(1 - \langle F_i(\bar{R}) \rangle_{emp} \right)} \right).$$  \hspace{1cm} (9)

As the update-step is computationally expensive, we do not select the update-dimension $i$ at random, but pick the one that will lead to the largest reduction $\Delta L$.

In this algorithm, we need to evaluate the model-probabilities for each possible binary pattern, i.e. $2^N$ patterns. For the large populations here, performing these calculations is computationally prohibitive, and we therefore used Markov-Chain Monte Carlo Sampling techniques (Metropolis and Ulam 1949; Robert and Casella 2004; Tierney 1994) as described in (Broderick et al. 2007) to approximate the relevant quantities: If we generate $n_{MC}$ samples from our model, we can use the sample-average across those samples to approximate averages across the whole probability distribution:

$$\frac{1}{n_{MC}} \sum_{R \in U_{\lambda}} F_i(\bar{R}) \approx \langle F_i(\bar{R}) \rangle.$$

To generate samples from our model, we use the fact that the conditional probabilities of our model can be evaluated in closed form:

$$P_{\lambda}(r_i = 1| r_{ji}) = \frac{P_{\lambda}(r_i = 1, r_{ji})}{P_{\lambda}(r_i = 1, r_{ji}) + P_{\lambda}(r_i = 0, r_{ji})} = \frac{1}{1 + \exp \left[ -\lambda \circ (\bar{F}(\bar{R}_{r_i=1}) - \bar{F}(\bar{R}_{r_i=0})) \right]}.$$

(10)
Here, $P_\lambda(r_i = 1 | r_{\neq i})$ denotes the conditional probability of neuron $i$ being active given the states of all other $N-1$ neurons, and $\bar{R}_{r=0}$ the vector $\bar{R}$ with its $i^{th}$ element replaced by 0. Most of the terms in the difference in Equation 10 do not depend on $r_i$, and will therefore cancel out. Using the conditional probabilities, we can use a variant of MCMC-sampling called Gibbs sampling (Robert and Casella 2004) to generate approximate samples from the model: One starts at some random binary vector $\bar{R}$. To obtain the next sample, we select a component $i$ at random, set it to 1 with the probability specified above, and to 0 otherwise. This produces the next random binary vector, and the process is iterated for subsequent binary vectors. It can be shown that, under fairly general assumptions, a sequence of binary vectors generated by this procedure will eventually converge and constitute samples from the probability distribution $P_\lambda(\bar{R})$ (Robert and Casella 2004). This procedure allows us to make the approximation:

$$\log \left(\frac{Z_{\lambda}}{Z_{\lambda}}\right) = \log \left(\sum_{\bar{R}} \frac{\exp(\bar{\lambda} \cdot \bar{F}(\bar{R}))}{Z_{\lambda}} \exp(\bar{\delta} \cdot \bar{F}(\bar{R}))\right)$$

$$= \log \left(\exp(\bar{\delta} \cdot \bar{F}(\bar{R}))\right)$$

$$= \log \left(\frac{1}{n_{MC}} \sum_{\bar{R} \in U_\lambda} \exp(\bar{\delta} \cdot \bar{F}(\bar{R}))\right)$$

(11)

To avoid dependence of this procedure on the choice of the starting point of the chain, one often throws out the initial segment of the chain (“burn-in”). In our application, however, varying the burn-in length did not seem to have a major impact on the performance on the algorithm, so we used all samples. The error in the MCMC approximation constitutes a lower bound on the possible accuracy of the algorithm.

If the generated samples were uncorrelated, the error in the MCMC approximation of the empirical moments would be on the order $\sqrt{\text{Var}(\bar{F}(\bar{R})) / n_{MC}} \leq 1 / \sqrt{4n_{MC}}$. As samples from an MCMC chain are, by construction, not uncorrelated, the variance of
the empirical feature means is larger, and would by larger by some constant factor. We made no attempt at estimating the additional variance resulting from correlations in the chain, and simply chose $n_{MC}$ such that $1/\sqrt{n_{MC}}$ was at least twice as small as the desired level of accuracy.

In theory, one should re-calculate $\langle \hat{F}(\hat{R}) \rangle_{\hat{z}}$ after every update step. As the generation of the MCMC chain is the most time-consuming component of the algorithm, it is much more efficient to perform a number of update steps before a new set is sampled (Broderick et al. 2007). At each step, the algorithm updates the parameter $i$ which leads to the biggest performance gain. Thus, some parameters $i$ will be updated multiple times, whereas other might not ever be updated and stay at zero. As a consequence, the resulting parameters can be very sparse, with many entries being exactly zero.

If the data is sparse, the MCMC sample can be stored in a memory-efficient manner by only storing a list of indices of those bins that contain a spike (Broderick et al. 2007). For our data set, this was not the case, and in particular, some neurons had spike probabilities which were bigger than 0.5. Therefore, we 'flipped' the values of every neuron that had a spike-probability bigger than 0.5 to obtain a surrogate data set that was more sparse than the original one, fitted a maximum entropy model to the surrogate data set, and subsequently transformed the parameters accordingly to obtain the optimal set of parameters for the original data set.

We used the algorithm described above to fit an Ising model to neural populations of up to 162 neurons. As the computational difficulties increase with population size, we only report the detailed results of the fit for the biggest population size, 162 neurons. We used a Monte Carlo sample size of $n_{MC} = 200,000$, and performed 25 gradient steps on each sample. We assessed the convergence of the algorithm by tracking the difference between the empirical moments and those of the Ising model under the current estimate of parameters. We calculated three error measures: i) the average absolute deviation of
the Ising model means from the data, \( E_{\text{mean}} = \frac{1}{N} \sum_i \left( \langle r_i \rangle_{\text{emp}} - \langle r_i \rangle_{\lambda} \right) \); ii) the average
absolute deviation in the covariances, \( E_{\text{covar}} = \frac{1}{N(N-1)} \sum_{ij} \left( \langle r_i r_j \rangle_{\text{emp}} - \langle r_i \rangle_{\lambda} \langle r_j \rangle_{\lambda} \right) \); iii) the deviation
in pairwise correlation coefficients, \( E_{\text{corr}} = \frac{1}{N(N-1)} \sum_{ij} \left( \langle c_{ij} \rangle_{\text{emp}} - \langle c_{ij} \rangle_{\lambda} \right) \), with correlation
coefficients defined as \( c_{ij} = \frac{\text{Cov}(r_i, r_j)}{\sqrt{\text{Var}(r_i) \text{Var}(r_j)}} \). The search algorithm was said
to have converged once all three performance metrics had reached the desired level of
accuracy. We defined the finishing condition as \( F_{\text{means}} = \frac{2}{N\sqrt{n_{MC}}} \sum_i \text{Var}(r_i) = 0.001 \), as
well as \( F_{\text{moments}} = 0.0009 \) and \( F_{\text{corrs}} = 0.005 \), analogously defined.

When fitted to the entire data set, the finishing conditions were reached after 4876
iterations of the algorithm (Fig. 1A). Once the algorithm had converged, both the means
and the correlation coefficients of the data were in very close agreement with those
measured experimentally (Fig. 1B). The log-probabilities of the resulting model \( P_{\lambda}(\bar{R}) \)
were highly correlated with those sampled directly from the data (correlation 0.96; Fig.
1C). Here, we used the assumption of conditional independence to sample the empirical
probabilities, \( P_{\text{mix}}(\bar{R}) \) (see below for definition, Eqn. 14). In contrast, the probabilities of
the independent model were substantially different (correlation 0.767). This indicates
that taking second-order correlations into account indeed leads to a much better
approximation of the full response probability distribution. The bias terms \( \{h_i\} \) found by
the algorithm were scattered around a mean of -2.87, with a standard deviation of 3.36
(Fig. 1E). As described above, the algorithm only changed some of the coupling terms
away from zero. At convergence, 78.6% of the interaction terms \( \{J_{ij}\} \) still had a value of
zero. Of those interaction terms that were updated, the mean value was 0.16 and the
standard deviation was 0.80 (Fig. 1E). Fig. 1D shows the resulting matrix of interactions,
with neurons sorted according to their information about shapes. Non-zero interaction
weights were mainly between neurons that were informative about the stimulus. This is
a consequence of the fact that many of the non-informative neurons had very low firing rates, and did not contribute much to the likelihood of the model.

Impact of Noise Correlation. For quantifying the impact of signal correlation, we compared performance of a decoder that used the true model of signal correlation with a decoder that used an approximate model that assumed statistical independence among all cells. These decoders were both tested on the same, shuffled spike trains. The analogous test for noise correlation would involve comparing the performance of a decoder that included all correlation versus a decoder that only included signal correlation, with both acting on the real, simultaneous spike trains (Nirenberg and Latham 2003). However, this test of the impact of noise correlation is not possible with our data set, because it requires that we measure the fully correlated response distribution for all $N$ cells, which involves sampling at least $2^N$ different probabilities, in order to build the decoder that included all correlation (signal plus noise). Therefore, we instead compared the performance of the same decoder, which here included only signal correlation, acting on either shuffled spike trains or simultaneously recorded spike trains. Our measure of the importance of noise correlation is thus not directly comparable to the previously proposed measure (Nirenberg and Latham 2003), nor does it provide a bound on performance.

RESULTS

We studied the role of correlations in the population code among retinal ganglion cells using a shape discrimination task. The task consisted of static, spatial patterns that were flashed onto the retina for half a second separated by a grey screen for one second. Each pattern or shape was a black region on a grey background; shapes included geometric objects, such as circles and squares, as well as letters (Fig. 2A). The shapes were roughly the same size as an individual ganglion cell’s receptive field center (Fig. 2B). They all had the same mean light level, contrast, and centroid, so they could not be distinguished by such low-level cues. We therefore expected that the
responses of individual retinal ganglion cells would be highly ambiguous for the identity of these shapes.

We recorded from many ganglion cells in the salamander retina using a multi-electrode array. Previous analysis has shown that even the smallest region in visual space is encoded by 100-200 ganglion cells (Puchalla et al. 2005), almost all of which project to the optic tectum (Wiggers 1999). In order to achieve a dataset of this scope, we recorded from twelve retinas, obtaining 162 cells. The pieces of retina were aligned across experiments such that the receptive fields were highly overlapping with a total coverage similar to that of the entire ganglion cell population (Fig. 2C). Some cells fired selectively for certain shapes (Fig. 2D first row, where the firing rate is largest for the vertical bar and nearly zero for the triangle), but most cells responded to multiple shapes (Fig. 2D third row, where the firing rate is small for the triangle, but nearly identical for the inverted T and the vertical bar). Of course, the brain does not have the luxury of calculating a firing rate histogram during real behavior, as this would require integrating over many identical trials. Instead, behavior must be based on the spikes from a population of ganglion cells on a single trial (Fig. 2E).

Our task was to distinguish one shape, called the target (T), from all of the other stimuli, called the distracter ensemble (D). We then repeated our analysis, treating each shape as the target. We chose this task because in our typical everyday visual experience we must identify specific stimuli against a background of a huge range of other possibilities. Because we pooled cells across multiple retinas, most correlation among cells was induced by common responses to the stimuli (known as ‘signal correlation’). For groups of cells recorded from the same retina, we shuffled their spike trains to remove all noise correlation, thus leaving only signal correlation. Recall that when the discrimination task involves a stimulus class (in this case the distracter ensemble), signal correlation can affect decoding performance. A major goal of this study is thus to quantify how signal correlation affects the structure of encoded visual information and to determine what kinds of decoders are necessary to read out such information.
Central to the study of a neural code is the question of what details of spike trains matter: Does one need simply to count spikes in a single time window? How finely does one need to distinguish between spike counts? Does more detailed spike timing also matter? Similarly, at the level of populations of neurons: Can one simply aggregate spike counts across many cells? Or is cell identity critical? And how significant are correlations among cells?

To formalize our approach to decoding or “reading out” the information conveyed by the optic nerve, we chose simpler representations of the response than the full spike raster across the population. Any simplification of the spiking response runs the risk of losing information important to the decoding task. However, an overly complex representation causes difficulties both in experimental sampling and decoding complexity. If the response representation contains too many parameters, they cannot be adequately measured from a finite data set, so the decoder tends to over-fit noise in the data and generalize poorly. Additionally, the complexity of the decoder required to read out the response could exceed the biologically plausible capability of circuits in higher visual centers. With the tradeoff between completeness and simplicity in mind, we tested a number of different response representations (Brenner et al. 2000; Jacobs et al. 2009; Krahe and Gabbiani 2004; Petersen et al. 2002; Reinagel et al. 1999; Victor and Purpura 1998). We examined the role of correlations in decoding performance in each case and found that the importance of these correlations in the population code did not critically depend on the specific choice of a representation.

Across the population, there was a strong response to the appearance of the shape that reached its peak 140 ms after stimulus onset at this light level (Fig. 3A). This was followed by sustained activity during the presentation, and a response to the offset of the stimulus. We measured the amount of information that each cell $i$ individually conveyed about shape identity, $I(r_i; S)$, with $S$ defined as the set of shapes, $S = \{s_1, s_2, ..., s_L\}$ (see Methods). This calculation was carried out using binary neural
responses from each 100 ms bin following stimulus onset (Fig. 3B). The bins beginning at 100 ms and 200 ms after shape presentation had higher information rates per cell than the later bins.

In these calculations, we also considered the importance of multiple spike events. To determine the effect of ignoring multiple spikes, we counted the number of spikes that each cell fired in a given time interval. If the number of spikes exceeded a threshold $n_{\text{max}}$, those extra spikes were discarded, and the neuron was said to have fired exactly $n_{\text{max}}$ spikes. We varied the value of $n_{\text{max}}$ from 1 to 5, and calculated the mutual information that the responses of each neuron contained about the set of 36 stimuli (see Methods). By comparing the mutual information obtained for different values of $n_{\text{max}}$, we could determine the loss in mutual information resulting from ignoring multiple spikes (Fig. 3B). This analysis revealed that characterizing the response only by “no spike” and “at least one spike” retained 73% of the mutual information obtained from using up to 5 spikes in the first time bin and even larger fractions in subsequent time bins (Fig. 3B).

Based on these results, we first considered the simple case of using binary responses for each cell (“no spike” vs. “at least one spike”) within a time bin 100-200 ms after stimulus onset. This time bin captured the transient response to the stimulus upon which the brain would base its fastest visual judgments (Thorpe et al. 1996). After describing the results for this simple case, we will return to more complex response representations, including: optimizing the threshold for binary responses, no spike/spike/burst, full multiple spike count, different time bins, multiple time bins, and first spike latency.

**The Binary Representation**

The binary response across cells was represented as a vector of zeros and ones for each shape presentation. Individual neural responses, \( r_i = \{0, 1\} \) for no spike or spike in cell \( i \), were combined into population response vectors, \( \mathbf{R} = (r_1, r_2, \ldots, r_N) \) for \( N \) cells. For
each cell we calculated a response probability for each $k$ out of $L$ shapes, $p(r | s_k)$ (Fig. 4A). For the first 36 shapes, which contained equal dark areas, the average response probability was quite similar ($0.277 \pm 0.027$; Fig. 4B). The small and large squares had very different response probabilities ($0.053$ and $0.489$, respectively). Some cells responded rarely to any of the shapes while others responded to nearly every shape. Naturally, cells with a response probability near 0.5 contained the most mutual information about shape identity (Fig. 4C). The information conveyed by any individual cell never exceeded 0.6 bits (compared against a total stimulus entropy of $\log_2(36) = 5.17$ bits), indicating that populations of neurons are necessary for high-fidelity discrimination among shapes.

In order to combine the information conveyed by many neurons, we used a decoding formalism. This approach is favored because a direct estimate of the mutual information conveyed by a large neural population can be prohibitively difficult to sample. In contrast, the performance of any decoder always sets a lower bound on the fidelity of information encoded by the neural population. Furthermore, a decoder that captures most of the structure of the spike trains can be expected to achieve a fairly tight lower bound. Since each discrimination task had just two alternatives (“target” vs. “distracter”), the optimal decision rule could be obtained from a maximum likelihood rule (Abbott and Dayan 2001; Green and Swets 1966). Given a response $\bar{R}$, we found the ratio, $P(T | \bar{R})/P(D | \bar{R})$. If this ratio was larger than a threshold, $\theta$, the decoder choose target $T$; otherwise it choose distracter $D$. The choice of $\theta$ determined the bias toward minimizing misses (false negatives) or false alarms (false positives). One often considers minimizing the total error, but this choice is not natural when the target is rare. With only 100 target trials, error rates less than 1/100 will tend to be dominated by the miss rate, which is not as well sampled as the false alarm rate. Furthermore, a total error of 1/36 can be achieved simply by having the decoder always answer “distracter”. Thus, in order to study the low error regime, we chose $\theta$ so that the hit rate ($1 – \text{miss rate}$) was at least 99% and measured decoding error as the false alarm rate at this criterion hit rate.
We used Bayes’ rule to convert the above probability ratio into quantities that could be estimated from our experimental data:

$$\frac{P(T | \bar{R})}{P(D | \bar{R})} = \frac{P(\bar{R} | T) P(T)}{P(\bar{R} | D) P(D)} \quad .$$  \hfill (12)

The prior probabilities $P(T)$ and $P(D)$ were fixed constants, and so without loss of generality we absorbed them into our definition of the threshold, $\theta$. Because the target was a single shape stimulus and because we had no noise correlation, we could always assume $P(\bar{R} | T = s_k) = \prod_i p(r_i | s_k)$. However, the distracter ensemble included multiple shape stimuli, so signal correlation affected the probability of activity patterns in the distracter ensemble, $P(\bar{R} | D)$. Our study involved two basic dimensions. First, we explored how signal correlation affected the population neural code by forming one probability distribution that included signal correlation, $P_{\text{mix}}(\bar{R} | D)$, and another that ignored it, $P_{\text{nd}}(\bar{R} | D)$. Second, we explored how different details of the spike trains affected coding fidelity through different choices of the response variables we included for each cell, $\bar{R} = \{r_i\}$. In all cases, our choice of $P(\bar{R} | T)$ and $P(\bar{R} | D)$ defined the decoding algorithm.

Probability distributions need to be estimated from experimental data, so we used 100-fold cross validation to ensure that we did not over-fit the data (Hastie et al. 2005). Specifically, we withheld the data from a set of 36 stimulus presentations (one for each shape), and constructed each decoder by using data from the remaining 3564 stimulus presentations to sample the above-mentioned probability distributions. Then, the decoders were used to predict the stimulus labels of each of the withheld 36 data-points. This procedure was repeated 100 times, such that each type of decoder was evaluated on each of the 3600 data-points exactly once.
Perhaps the most straightforward decoder, which we will call the “cell count” decoder, characterizes the response probability, $P\left(\bar{R} | s_k \right)$, by the number of cells responding, $K$: $P_{cc}\left(\bar{R} | s_k \right) \rightarrow P_{cc}\left( K | s_k \right) : K = \sum_i r_i$. For shapes 1 to 36, the cell count decoder performed very poorly (Fig. 5A; median false alarm rate = 62%). This is not surprising since the average response probability across the population was very similar for all these shapes (Fig. 4B). This reflects the ambiguity in the neural code: since all shapes had the same mean light level and contrast, they evoked similarly strong total spike counts in the ganglion cell population. However, performance was very good on the small and large squares (false alarm rate < $10^{-3}$), because these shapes elicited very different response probabilities in each neuron. Subsequent analyses were carried out on only the first 36 shapes, excluding the differently sized squares.

Since the number of responding cells proved to be an insufficient characterization of the response pattern, we constructed decoders that included information about which cell was firing (cell identity). Because we wanted to study the role of correlation in coding for shape identity, we compared an independent decoder that ignored correlation to a decoder that included the patterns of correlation among ganglion cells induced by common stimulation.

First, we created a decoder that ignored correlation, such that the population response distribution was assumed to be a simple product over each cell’s response probability:

$$P_{ind}\left(\bar{R} | D \right) = \prod_{i}^{\text{cells}} p\left(r_i | D \right) \quad \text{where} \quad p\left(r_i | D \right) = \frac{1}{L-1} \sum_{k}^{\text{distracter shapes}} p\left(r_i | s_k \right). \quad (13)$$

This assumption greatly simplifies the structure of the decoding algorithm, and if signal correlation does not have a major impact on the activity patterns of the neural population, then this decoder will achieve nearly full performance. Under this assumption, the resulting maximum likelihood decoding rule is linear, i.e. a stimulus is classified as the target shape if $\sum_{i}^{\text{cells}} w_i r_i$ is greater than a threshold, where the
weights, \( w_i \), can be calculated from the firing rates of the neurons (see Methods). We call this decoder the *independent decoder* to emphasize that this form assumes that the activity in the neural population is independent across all shape stimuli. This decoder is an example of a class of decoders known as linear decoders, linear classifiers, or perceptrons (Hastie et al. 2005). It is important to keep in mind that one can construct a different linear decoder with weights chosen to partially capture the effects of correlation. We call this the *optimal linear decoder* and explore its properties below.

With the independent decoder, performance was significantly enhanced compared to the cell count decoder (Fig. 5B; notice the change to a log scale for the error rate). This indicates that cell identity is a crucial component of the ganglion cell neural code, as has been observed in other systems (Panzeri et al. 2003; Reich et al. 2001a). At the same time, it was not possible to distinguish any shape from the others at very low error rate (Fig. 5B). The median false alarm rate across all the shapes was 5.2%. There was considerable variation of the error depending on the particular target shape, with values ranging from as high as 24% to as low as 0.14%.

The failure of the independent decoder to reach lower false alarm rates suggests that the task might require a model that captures the correlations between different cells’ activity that were induced by common stimulation. Signal correlations can be perfectly modeled by constructing a product distribution across the population given each shape and then averaging the population response probabilities over all \( L-1 \) distracter shapes:

\[
P_{\text{mix}}(\bar{R} | D) = \frac{1}{L-1} \sum_{k} \left( \prod_{i} p(r_i | s_k) \right) .
\]  

We called the decoder that used this distribution the *mixture decoder*, because it is an explicit mixture of independent terms coming from the response to each shape. We also note that this is one example of a nonlinear decoder, because it has a form that cannot be captured by the weighted sum rule characteristic of a linear decoder. Because this decoder embodies the true model of signal correlation among neurons, we expect that its performance will constitute a tight lower bound on the population’s possible
performance, thus capturing essentially all of the shape information encoded by the neural population.

The mixture decoder performed much better than the independent decoder (Fig. 5C). In fact, this decoder did not make a single error in the entire experiment for several of the shapes. Because false alarms could occur for any of 35 non-target shapes over 100 different trials, there were 3500 opportunities for a false alarm. We could not estimate the error very well for these shapes, so we chose to display a conservative value of 0.5 errors / 3500 trials = 1/7000. The mixture decoder outperformed the independent decoder on all of the shapes ($p < 10^{-10}$). To characterize the difference in performance, we calculated the ratio between the error of the independent and mixture decoders and took the geometric average over all shapes. This improvement factor was $6.4 \pm 0.9$ (s.e.m). The degree of improvement achieved by the mixture decoder depended systematically on the overall discriminability of the target stimulus, with greater improvement found when the independent decoder already had better performance (Fig. 5D). Thus, the mixture decoder tended to amplify the discriminability established by the independent decoder by increasingly large factors.

To explore how decoding performance depended on the population size, we applied both decoders to populations of size $N$ neurons, randomly selecting 100 subsets at each value of $N$ and averaging the error over all subsets (Fig. 6A,B). For the independent decoder, the error rate typically decreased exponentially for small $N$, but reached a constant for large $N$. Thus, there were diminishing returns from the independent decoder as more cells were added to the population. In contrast, the error rate for the mixture decoder showed steady improvement with population size. Interestingly, for population sizes of up to ~30 neurons, the performance of the independent and mixture decoders was nearly identical, differing by less than 10% (Fig. 6A,B). Thus, even for population sizes larger than sampled in many studies of neural coding, signal correlations made no appreciable difference to the error rate in this discrimination task, and a linear classifier was sufficient to read out the encoded information. However, for the entire population, the mixture decoder exceeded the
independent decoder by large factors (Fig. 6A: 169-fold; Fig. 6B: 6.7-fold) and reached zero errors during the entire experiment (3500 distracter trials) for 4 out of 36 shapes (Fig. 5C).

We next explored how decoding performance depended on the number of shape stimuli in the distracter ensemble. Twelve shapes were randomly selected as target stimuli, then for each target shape, we randomly made 20 selections of groups of \( L \) out of 35 other shapes for the distracter ensemble. We computed the error rate for independent and mixture decoders making another 25 random selections of either 50 or 100 cells. Results were averaged over all selections of shapes and cells. We found that the error rate was systematically lower for 2 or 5 shapes, but seemed to reach roughly its maximum at values of \( L \) of about 10 shapes for both decoders and population sizes (Fig. 6C,D). The dependence of the error rate versus number of shapes \( L \) was qualitatively the same for individual choices of shapes for target and/or distracter (data not shown). This indicates that, in our ensemble, the effect of using multiple distracters is evident at very low numbers of distracter shapes, and afterwards is only weakly dependent on the number of distracters.

Why can the mixture decoder outperform the independent decoder by such large factors? One way to gain insight is to visualize the decision boundaries of the two decoders in the space of neural responses. This cannot be done easily, because the response space has 162 dimensions. However, we can use standard techniques, like principal component analysis, to find directions of the largest variability in the space of neural responses and reduce its dimensionality. We chose to display the response of the entire population along three dimensions: the first two principal components and the dimension of the response space defined by the weights of the independent decoder, \( \{w_i\} \) (Fig. 7A). In this subspace, the decision boundary of the independent decoder (Eqn. 4) is simply a line at a fixed coordinate along the linear discriminator direction (Fig. 7B). Because this decision surface must be a hyperplane, it cannot fully separate the cloud of points corresponding to the target stimulus from those corresponding to the distracter ensemble. In contrast, the mixture decoder can implement a curved decision
boundary, which separates target from distracter very effectively (Fig. 7C,D). Note that, as the shape of the decision boundary of the mixture decoder is potentially sensitive to all 162 dimensions, its decision boundary cannot unambiguously be visualized in this two-dimensional representation. Therefore, the plotted line represents the decision boundary in an arbitrarily chosen slice through this 162-dimensional space, and points classified as target can appear on both sides of the plotted boundary.

**Other Response Representations**

In order to determine whether the results we have seen thus far were dependent on the choice of the simplified binary response representation, we studied nine different representations of increasing complexity (Fig. 8A; see Methods). These representations allowed more than one spike per time bin to include additional spike count information as well as multiple time bins to include additional spike timing information. For each response representation, we built maximum likelihood decoders and looked at how much the error rate was reduced by including signal correlations (ratio of error for independent vs. mixture decoders) and how much the error rate was reduced by including additional details of each cell’s spike train (ratio of error for the binary code in the first 100 ms time bin vs. more complex codes, in each case using the independent decoder). To summarize the results for a single coding scheme, we calculated the geometric mean of both ratios across all target shapes.

We found that for all these response representations, the mixture decoder showed greater improvement over the independent decoder than we found previously for the binary code in the first 100 ms (Fig. 8B). This implies that correlations had an even stronger effect for more complex coding variables that included additional timing and spike count information. In addition, the improvement in performance for the mixture decoder tended to increase as the improvement from having more complex response representations increased. Both results are consistent with the amplifying effect of signal correlations noted previously (Fig. 5D).
While both the burst and multi-spike representations improved discrimination performance, the multi-spike representation did much better. This suggests that large responses, which could be quite rare, conveyed significant shape information and might be important for the brain to recognize as distinct neural symbols. Response representations that formed binary words in the first time bin (2 \( \times \) 50 ms, 4 \( \times \) 25 ms), effectively resolving extra temporal details in the spike trains, both showed a larger improvement for the mixture decoder than for the independent decoder. This is consistent with an amplifying effect between timing information for single neurons and the effect of correlation at the population level.

The binary representation for the second time bin, which included the initial phase of the sustained response, had roughly the same performance for the independent decoder but considerably better performance for the mixture decoder. This is consistent with the fact that the pairwise correlations were stronger in the second time bin than the first (rms correlation coefficient = 0.102 vs. 0.075). Also, the firing rates for most cells were lower in this time period (Fig. 3A; 3.9 spikes/s/cell in the second time bin vs. 7.1 spikes/s/cell in the first time bin), implying that the population code was more efficient in the amount of shape information encoded per spike during the sustained response. Together, these results suggest that the manner in which ganglion cell populations represent visual information shifted somewhat from the transient portion of the response to the sustained (Mechler et al. 1998; Reich et al. 2001b). Along these lines, the response representation with the best performance was the one which assigned a different binary response to the first and second time bins (2 \( \times \) 100 ms), suggesting that integrating information across the transient and sustained responses allows a significant improvement in the fidelity of the population code.

**The Spike Latency Representation**

Finally, we explored a spike latency representation, in which each cell’s response included both whether the cell fired a spike as well as the time of the first spike, \( t_i \) (Gollisch and Meister 2008; Panzeri et al. 2001; VanRullen et al. 2005). This modified the response probability for cell \( i \) to a joint distribution of a binary variable (spike, no-
spike) and an analog variable (first spike latency):
\[ p(r_i | s_k) \rightarrow p(r_i, t_i | s_k) = p(r_i | s_k) p(t_i | r_i = 1, s_k) \] (see Methods). We then formed population response distributions for independent and mixture decoders, as before.

Using the mixture decoder, the spike latency representation was dramatically more effective than the binary representation, reaching zero error for 20 out of 36 shapes and having an average improvement factor of 8.5 ± 2.2 (Fig. 9A). Including signal correlation was crucial for the spike latency representation, as the mixture decoder outperformed the independent decoder by an average factor of 26 ± 6.9 and a maximum factor of 760 (Fig. 9B). The impact of correlation for the spike latency representation was the largest of all the representations we tested. In stark contrast, spike latency information offered almost no improvement over spike/no-spike information for the independent decoder (Fig. 9C). These comparisons indicate that there is a strong and beneficial interaction between spike latency information and signal correlation among ganglion cells.

In the spike latency representation, time was measured relative to the onset of the stimulus, but how does the brain know when the stimulus occurred? Our experimental design, which involved flashing a stimulus onto the retina, can be thought of as an approximation for how the visual world is sampled by saccadic eye movements. This suggests two possibilities for how the brain might know the time of stimulus onset. First, there will be a volley of spikes from many ganglion cells over the entire retina, giving a ‘time stamp’ that is much more accurate than can be derived from the cells we happened to record. Second, the brain, in principle, knows when it generated the saccade, so that it might send an efference copy of the motor command to the higher brain centers that interpret retinal spike trains (Sommer and Wurtz 2008). In either case, we can not measure exactly how accurately the brain knows the time of its own saccade, but we can assume that there is a timing jitter of standard deviation \( \Delta \) from trial-to-trial and see how robust the spike latency representation is to this uncertainty in the time of stimulus onset (see Methods). Small jitters of up to 5 or 10 ms had only a small effect on decoding performance; once the jitter was increased beyond 25 ms, performance began to deteriorate (Fig. 9D). Thus, this representation is robust to fairly
large uncertainty in the time of stimulus onset. Interestingly, the level of timing jitter that can be tolerated is similar to the timing precision of ganglion cells (Berry et al. 1997).

**Noise Correlation**

So far, our analysis has focused on the role of signal correlation, but the effects of noise correlation can be studied for groups of ganglion cells recorded simultaneously from the same retinal patch. To analyze the importance of noise correlation, we applied the mixture decoder, which correctly captures signal correlation, to simultaneously recorded spike trains (see Methods). To the extent that noise correlation changes the firing patterns of retinal ganglion cells, the decoder based only on signal correlation will make more errors on simultaneous spike trains than shuffled spike trains. The error rate was systematically higher when applied to simultaneous than shuffled spike trains (Fig. 10A). Averaged over all shapes, the error rate was a factor of $2.5 \pm 0.25$ (s.e.m.) larger for simultaneous spike trains in one retinal patch with 25 cells, an effect that was modest but significant. However, as noted above, even signal correlation does not affect the decoding performance greatly for such small populations.

Analyzing the same cells, we found that ignoring signal correlation (as in Fig. 9B) increased the error by a factor of $1.9 \pm 0.17$ (s.e.m.) for this group of cells – a roughly comparable effect. Calculating these error ratios for 13 different retinal patches, we found that noise and signal correlation consistently had a comparable impact (Fig. 10B). Similar results were found for the binary representation, but with a somewhat reduced magnitude of the effect for both signal and noise correlation (Figs. 10C,D). While these two measures of the impact of ignoring correlation have different interpretations (see Methods), the rough order-of-magnitude agreement suggests that noise correlation might also have a substantial effect on larger populations. However, the limited impact of correlations at population sizes up to 25 cells makes an extrapolation to larger populations unreliable.
Biologically Plausible Decoders

Most of our analysis has involved testing the fidelity of the population code by using a maximum likelihood decoder acting on the true distribution of responses, $P_{mix}(\bar{R})$. Such a scheme achieves optimal performance for our two-alternative, target vs. distracter discrimination, and is thus appropriate for assessing the amount of shape information encoded by the entire population. However, this decoder is not necessarily available to the animal. The problem is that the formation of $P_{mix}(\bar{R})$, while simple and straightforward, has a flaw. In order to sample the response of each cell for a given shape, $p(r_i | s_k)$, we need to know on any given trial which shape was presented. This is easy for us to do, because we designed the experiment and therefore always know what shape was presented, but this is precisely what the animal is trying to figure out from its neural activity patterns. Therefore, our method of learning $P_{mix}(\bar{R})$ from the data is circular and cannot be used by a behaving animal. In addition, we needed many presentations of each shape stimulus to form $p(r_i | s_k)$. This is practical when there are a small number of discrete stimuli, but not realistic when the full range of natural stimuli is possible. In fact, during natural behavior, it is unlikely that exactly the same pattern ever appears twice on the sensory periphery.

For these reasons, we sought to find a biologically plausible strategy for extracting the shape information present in this correlated population of retinal ganglion cells. We wanted this strategy to operate without the requirement that the distracter set consists of a small number of discrete, labeled stimuli repeated many times. The idea is that what an animal does plausibly have available is some form of reinforcement learning signal that serves to distinguish target from distracter, but not detailed and exact information about which member of the distracter ensemble was present at each moment.

We considered two kinds of biologically plausible decoders that fulfilled these criteria. The first was an optimal linear classifier, and the second was a maximum entropy
decoder (Hastie 2001; Hastie et al. 2005; Schneidman et al. 2006). In each case, we trained the decoder on part of the data and tested it on the remaining data. We used standard numerical techniques to learn the parameters of each decoder and did not consider here an explicit, biologically inspired learning algorithm. The goal was to determine the best possible performance for decoding algorithms that could be implemented with information that is plausibly available to a behaving animal.

**Optimal linear classifier:** As shown in Methods, our independent decoder is an example of a linear decoder, because its action can be summarized by a linear decision rule:

$$\sum_{i}^{\text{cells}} w_i r_i > \Theta.$$  

If the neural population actually has independent activity, then our independent decoder is the best possible choice of a linear decoder, in the sense that we are using the best weights, $\{w_i\}$ (Rieke et al. 1997; Warland et al. 1997). However, our results clearly demonstrate that correlations induced by common stimulation have a dramatic effect on the ability of decoding algorithms to read out information encoded by the ganglion cell population. By choosing a different set of weights, it may be possible to capture, at least to some degree, the effects of correlation and thereby find a better linear decoder. In choosing optimal weights, one need only distinguish *target* from *distracter*, so this decoder does not require detailed information about the identity of every stimulus in the distracter ensemble. We used numeric optimization to find the best set of weights for making shape discriminations with our ganglion cell spike trains (see Methods) and formed what we called the optimal linear classifier.

Indeed, the performance of the optimal linear classifier exceeded that of the independent decoder (Fig. 11A; error ratio = 2.6 ± 0.3). However, the optimal linear classifier still did not match the performance of the mixture decoder (Fig. 11B; error ratio = 4.3 ± 0.8). These comparisons indicate that while the linear decoding algorithm can capture a significant portion of the information encoded by ganglion cell populations, there is still substantial potential for nonlinear decoders to capture more information and to reach very low discrimination error rates.
**Tempotron decoder:** We can ask a similar question for the case of the spike latency representation: although the independent decoder we formulated did not have very good performance, is there a better choice of a linear decoder? The literature regarding timing-based decoders is not nearly as well developed as the study of linear classifiers, but an important recent development is the tempotron decoder (Gütig and Sompolinsky 2006). This decoder uses spike timing information to implement curved decision boundaries in the space of population responses and achieves similar but greater capacity than the optimal linear classifier. The tempotron has a single weight for each cell in the population, and combines all weighted spike times linearly. We trained it by optimizing these weights for the best possible decoding performance (see Methods). However, we found that the tempotron provided almost no advantage over the independent decoder (Fig. 11C): the average error ratio (independent latency decoder versus the tempotron) was $0.67 \pm 0.2$, and the tempotron did not achieve zero error for any of the shapes. In contrast, the mixture decoder reached zero error for 20 shapes and had error lower than the tempotron by an average factor of $32 \pm 11$ (Fig. 11D). Thus, nonlinear decoders are likely to be necessary to read out high-fidelity timing information from correlated neural populations.

**Maximum entropy decoder:** Another approach to finding a biologically plausible decoder relies on the idea that simple statistics of the firing patterns of the neural population, such as the firing rates and pairwise correlations of all the neurons, are available to a behaving animal. These simple statistics can be accumulated over the entire distracter ensemble without having any knowledge of the identity of the stimulus at every moment. Then, these simple statistics can be used to form an approximate probability distribution for the correlated responses of the neural population that can then serve as the basis for a maximum likelihood decoder.

How can we find this approximate probability distribution? A promising approach is to use the maximum entropy principle to form a probability distribution that matches the simple statistics – firing rates and pairwise correlations – while making no other assumptions. This notion of making as few assumptions as possible can be formalized...
by requiring that the approximate probability distribution remain as random as possible
given the constraints, which is the same as requiring that it have maximum entropy
(Amari 2001; Jaynes 1957; Schneidman et al. 2003). This model was shown to be
highly accurate for groups of 7-10 retinal ganglion cells (Schneidman et al. 2006; Shlens
et al. 2006), and more recently for populations of 40-100 ganglion cells (Shlens et al.
2009; Tkachik et al. 2006).

The maximum entropy model has a probability distribution with separate weights for
each cell and for interactions among each pair of cells, written here as \( \{h_i\} \) and \( \{J_{ij}\} \),
respectively:

\[
P_{\text{maxent}}(\vec{R}) = \frac{1}{Z} \exp \left( \sum_i h_i r_i + \sum_{i,j} J_{ij} r_i r_j \right), \quad (15)
\]

where \( Z \) is a normalization constant. We used standard Monte Carlo numerical
methods to learn the parameters of the maximum entropy distribution (see Methods).
Importantly, the parameters of the model were derived from statistics of the full
ensemble of firing patterns with no explicit knowledge of the different stimulus conditions.

For any given target, the distracter ensemble consists of the other 35 shapes. For
convenience, we made the approximation \( P_{\text{maxent}}(\vec{R}|D) \approx P_{\text{maxent}}(\vec{R}) \), which includes
responses to all 36 shapes. This approximation had the advantage that it did not require
us to determine separate parameters for each target shape. It also has the conceptual
appeal that with this approach the animal could compile over its entire lifetime one
single model of the probability of neural responses given the natural stimulus ensemble.
We note that when the target shape is present, there are no correlations, and so the
response distribution \( P(\vec{R}|T) \) is simply the independent distribution summarized by the
weights, \( \{h_i^{(T)}\} \).

We first tested the performance of the maximum entropy decoder using the binary
representation. The maximum entropy decoder performed significantly better than the
independent decoder ($p < 10^{-5}$ for difference in mean; Fig. 12A), indicating that it was successful in capturing at least part of the correlation structure in the neural population. The average improvement factor was $4.0 \pm 0.7$, and the maximum improvement for a single shape was a factor of 56. Another measure of the success of the maximum entropy model is to compare its performance with the full model of correlations (Fig. 12B). While the error rate for the maximum entropy decoder was always higher than the full model, its performance was close for many shapes. For 24 of 36 shapes, the difference in error rate was less than factor of three, and the overall ratio of error rates was $2.8 \pm 0.3$. These results show that it is possible to achieve order-of-magnitude reductions in the rate of discrimination errors by modeling correlations among neurons with the pairwise maximum entropy model. The error rate for the best shape was less than one per experiment, indicating that this form of decoder can reach the low error regime. Furthermore, its performance came close to that for the complete model of signal correlation in the neural population.

Because the spike latency representation was so successful, we wanted to test the maximum entropy decoder for this response representation. However, the latency of the first spike is an analog variable, which is not readily incorporated in the maximum entropy formalism. Instead, our approach was to separate information about whether there was a spike or not (binary representation) from information about the spike latency if there was a spike. We then used the maximum entropy probability distribution for the binary representation and assumed independence for the spike latencies: $P_{\text{maxent},T}(\vec{R},\vec{T}) = P_{\text{maxent}}(\vec{R}) P_{\text{ind}}(\vec{T})$, where $\vec{T}$ denotes a vector of first spike latencies compiled across $N$ neurons. Such a decoder will only capture information about correlations in whether cells spike or not but not correlations in when their spikes occur.

Despite this limitation, the maximum entropy decoder performed very well, with 8 shapes reaching zero error, an average improvement factor of $7.7 \pm 1.9$, and a maximum of 370 compared to the independent latency code (Fig. 12C). When compared against the full mixture decoder applied to both spike occurrence and latency, we found that the maximum entropy decoder had larger error by an average factor of
For the easiest shapes to discriminate, both decoders reached zero error, while the maximum entropy decoder underperformed for shapes with intermediate discriminability (Fig. 12D). Another way to assess how severe was the assumption of modeling correlations in whether cells spike but not when they spike is to form the analogous decoder using the full model of signal correlations:

\[ P_{\text{mix},T}(\bar{R},\bar{T}) = P_{\text{mix}}(\bar{R})P_{\text{ind}}(\bar{T}) \]

The performance of this hybrid mixture model was similar to the maximum entropy decoder (Figs. 12C,D), indicating that much of the drop in performance came from assuming independence for spike latencies, rather than in a failure of the maximum entropy model to capture correlations in whether cells spike together. Together, these results show that the maximum entropy model offered substantial improvements over the independent decoder, especially for the spike latency representation.

Another advantage of the maximum entropy approach is that the corresponding decoder can be written as a simple rule in terms of the parameters of the maximum entropy distribution:

\[ h_i(T) - h_i(0) \sum_{\text{cell}} \sum_{\text{cell pairs}} J_{ij} r_i r_j > \Theta' \]

Notice that this rule is similar to the rule for a linear classifier, except with the extra term added that depends on the interaction terms \( J_{ij} \). An elegant feature of the linear classifier is that one can readily imagine how a single readout neuron could implement this decoder: Each input spike train makes a synapse onto the dendrites of the readout neuron with an appropriate weight, \( w_i \), all such inputs are summed linearly, and the readout neuron must have an appropriate threshold for generating its output, \( \Theta \) (Fig. 12E).

For the maximum entropy decoder, nonlinear properties of dendritic integration could allow a single readout neuron to implement its decision rule (Fig. 12F). As before, individual synapses have weights given by \( w_i = h_i(T) - h_i(0) \). If two synapses are made within proximity on a single dendritic branch, then the effect of the soma can be super-linear, which effectively implements \( J_{ij} > 0 \) when both synapses have a spiking input. The degree of the superlinearity depends on the spacing between the synapses, allowing different values of \( J_{ij} \). Such behavior has been shown to result from NMDA-
dependent plateau potentials (Poirazi et al. 2003; Polsky et al. 2004), but other biophysical mechanisms are possible. If instead, two synapses are farther apart on the same dendritic branch, then the net effect on the soma can be sub-linear due to the shunting effect of synaptic conductances (Agmon-Snir et al. 1998; Hausser et al. 2001), which effectively implements $J_{ij} < 0$. Finally, synapses on different dendritic branches often add linearly, implementing $J_{ij} = 0$ (Cash and Yuste 1999).

**DISCUSSION**

Our results suggest a rough hierarchy for representation of visual information by populations of retinal ganglion cells. For coarse stimulus properties, like the overall size or contrast of an object flashed onto the retina, successful discrimination was achieved using simple properties of the neural activity, like the overall spike count compiled over many cells. However, decoding mechanisms that were only sensitive to the population spike count could not resolve many finer details about the stimulus, such as the shape of objects with the same size and contrast. For these finer details, a rough estimate could be achieved using a linear decoder that had different weights for each ganglion cell. Such decoders effectively took into account information about the identity of which cells fired and which cells were silent. For the stimuli in our study, linear decoders acting on binary neural responses typically had false alarm error rates of ~2-10%. However, ganglion cell populations actually encoded high-fidelity visual information about the shape of objects, which allowed discrimination among differently shaped objects with error rates so low that we often could not measure them in our ~3 hour experiment. This level of performance has not previously been demonstrated for a complex spatial task. Reading out this information required nonlinear decoders that took into account the pattern of correlations among cells induced by the stimulus.

High fidelity coding also required large neural populations. For populations of up to ~30 ganglion cells, the error rates for shape discrimination were often quite high, of the order of 10%, and the performance of linear and nonlinear decoders was roughly the same.
For the independent decoder, larger neural populations experienced diminishing returns, such that the error rate often did not drop below 5% using all 162 neurons. Given these trends, it seems unlikely that substantial discrimination improvement could be achieved even with much larger neural populations. In contrast, the mixture decoder showed continual improvement as more neurons were added, for many shapes reaching an error rate two orders of magnitude lower than was possible with the independent decoder. Many previous studies of the effects of correlation involved populations smaller than the scale at which nonlinear decoders substantially outperformed linear decoders, suggesting that conclusions about the importance of correlations for population neural codes may need to be re-evaluated for larger populations. Of course, our study primarily tested the importance of signal correlation, and noise correlation has different strength and potentially different impact on the neural code of large populations.

We tested for the importance of spike count and spike timing information by constructing decoders for 10 different representations of single neuron spike trains. These representations included multiple spike counts, multiple time bins, and first spike latencies. In general, more complex representations allowed significantly lower error rates for the independent decoder. In addition, the error rate for the mixture decoder decreased by a substantially larger amount. Of all the representations considered, by far the most successful was the one that used first spike latency measured on a 10 ms time scale, in addition to the presence or absence of a spike. For the spike latency representation, it was particularly important to use a nonlinear decoder that took into account the effects of signal correlation among ganglion cells: Performance for the independent decoder was hardly different than for the binary response representation, but performance for the mixture decoder was lower by an average factor of 8.6. While some previous studies have examined temporal coding (Gollisch and Meister 2008), this strong interaction between temporal coding and signal correlation has not previously been explored.
We showed that optimizing the weights of a linear classifier could partially take into account the effects of correlation, but that such decoders only achieved modest improvements over the independent decoder for the binary response representation. Similarly for the spike latency representation, tuning the weights of a tempotron was not effective at all. One explanation for the poor performance of the tempotron is that the number of stimuli we were attempting to classify (3600 stimuli) was larger than the capacity of the tempotron (~500 stimuli for 162 cells). This raises the possibility that a multiple-stage tempotron decoder with nonlinear hidden units might be more effective. Together, these results argue strongly for the use of nonlinear decoders to read out high-fidelity information from a correlated neural population. Finally, we showed that the effects of signal correlation could be well captured by a maximum entropy decoder that only required knowledge of the firing rates and pairwise correlations among neurons, statistics of the neural population activity that are available to a behaving animal (Fig. 12). To study the performance of this decoder in a regime where correlations have a substantial impact on decoding performance, we had to be able to fit maximum entropy models to populations of over 100 neurons (Fig. 1).

**Experimental Design**

Several elements of our experimental design were essential to our conclusions. First, we probed a regime in which the responses of individual ganglion cells were ambiguous by selecting shapes matched in their total size (dark area). Thus, for the population to achieve low error, it not only had to average over the noise in single neurons but also overcome their coding ambiguity – namely, the fact that multiple stimuli could produce a similar response in any individual ganglion cell. Because we deal with an enormous number of possible stimuli in our everyday vision, ambiguity is a crucial issue for the population code during real behavior. In contrast, for stimuli that evoked different firing rates in most ganglion cells (e.g., the small vs. large squares in our study), accurate discrimination could be achieved with a very simple decoder that merely counted the total number of spikes in the population. This example suggests that while there are many discrimination tasks for which simple decoders are effective, there also exist finer
discriminations among stimuli for which the structure of correlation in the neural population is critical.

Second, the task we considered was to discriminate one target stimulus from all of the other stimuli in our experiment. This design is contrasted with many classic studies of discrimination thresholds, which attempt to distinguish between two very similar stimuli, often distinguished by variations in a single stimulus parameter, such as contrast or orientation. While these studies are clearly important for understanding the limits of a neural code, they do not tackle the more general problem that neurons tune for multiple stimulus properties so that their firing rates cannot be unambiguously associated with the value of a single stimulus parameter. Furthermore, our design should be viewed as the typical circumstance in which the visual system must perform, as we very rarely know ahead of time that the environment only contains one of two possible objects in a small region of visual space. An important corollary of this design is that because the distracter condition included many possible stimuli, correlation induced by common stimulation (signal correlation) was present in the distribution of population responses. Even more realistic and behaviorally important is the visual task of identifying a class of stimuli that represent different patterns of activation on the sensory epithelium corresponding to the same cause in the world, such as arising from the same object under different viewing conditions. In this case, the distribution of population responses for the target would also be affected by signal correlation.

Comparison to other studies of population neural codes

A number of studies of population neural codes have concluded that correlation has only a small effect (reviewed in (Averbeck et al. 2006); also see (Oizumi et al. 2010)). One common feature of most of these studies is that they involved smaller neural populations, often not more than ten cells. Because we also saw almost no effect of correlation for population sizes up to ~30 cells, our results should be viewed as broadly consistent with such previous studies. Another important difference is that most previous studies have focused on the effects of noise correlation rather than signal correlation. Noise correlation is the mysterious ingredient in the population neural code,
because simultaneous recording is required to measure this form of correlation, and
such methods are very challenging on the scale of hundreds of neurons. We also
studied the impact of noise correlation on the performance of our nonlinear decoder
(Fig. 10). Unfortunately, we could only perform these tests on populations of up to 25
ganglion cells, and at this population size, all effects of correlation were modest.

Pillow et al. (Pillow et al. 2008) reported a 20% increase in information rate in decoding
the light intensities of a randomly flickering checkerboard when they incorporated a
successful model of noise correlation among parasol ganglion cells in the macaque
retina. Their study involved a group of 27 cells, similar to the largest population we
recorded from a single retinal patch. The numerical impact of noise correlation in this
study is difficult to directly compare to our results, because Pillow et al. used an
information theoretic measure applied to a large ensemble of stimuli, while we used a
measure of discrimination errors with a discrete set of stimuli. However, this study
provides an additional case where incorporating models of the noise correlation can be
useful for decoding.

Several other studies are worth noting in more detail. Jacobs et al. (Jacobs et al. 2009)
studied decoding of spatial shapes (in this case gratings of different spatial frequency)
using a large population (~300 ganglion cells) in the mouse retina. This study had
several results in common with ours. First, good performance required large
populations, as we reported (Fig. 5). Second, a code taking spike timing into account
gave substantially better results than the spike count representation, similar to our
analysis of more complex response representations (Fig. 8), in particular the spike
latency representation (Fig. 9). Third, including correlations between spikes for the
same cell improved performance. We did not directly study the case of correlations
between spikes from the same neuron, although the response representations that
involved multiple spike counts and multiple time bins were sensitive to such correlations
in an indirect way. Our study, however, involved a more challenging task in which we
attempted to discriminate one spatial shape from an ensemble of other possible shapes,
a task in which signal correlation among neurons could affect the decoding outcome.
Jacobs et al. attempted to discriminate among single pairs of stimuli, so that signal correlation did not come into play, and consequently they did not study the effect of correlation among neurons.

Gollisch and Meister studied spike latency codes in the salamander retina (Gollisch and Meister 2008). They found that for certain stimulus discriminations, spike latencies encoded more information than spike counts. Furthermore, the relative spike latency between pairs of ganglion cells encoded additional information that was not available in single neuron responses. This study used different classes of stimuli than ours (gratings vs. shapes) but reached consistent conclusions, namely that a spike latency code was very effective and that correlations between neurons were important for this code. This study also involved smaller numbers of ganglion cells, so it did not reveal the strong interaction between spike timing and correlation for large neural populations.

Greschner et al. used populations of up to 20 ganglion cells in the turtle retina to discriminate among 56 different steps in a spatially uniform light intensity (Greschner et al. 2006). For each cell, combination of firing rate and spike latency variables were included in a linear discriminant analysis (LDA), which is a form of linear classifier. While the overall level of discrimination performance in this study was low compared to ours, the authors found several complementary results: 1) the discrimination performance began to saturate at the largest ganglion cell population sizes; 2) a combination of spike count and spike latency information significantly improved discrimination performance; 3) inclusion of early and late responses from the ganglion cells also improved performance.

Finally, Schaub and Schultz constructed maximum entropy (Ising) decoders and applied them to a model of orientation-selective neurons matched to properties of neurons recorded in cortical area V1 (Schaub and Schultz 2012). They found that the performance of the Ising decoder exceeded that of an independent decoder and an optimal linear classifier (OLC) over a wide range of conditions. Furthermore, the
improvement in performance increased with stronger pairwise correlation among neurons, underscoring the importance of accounting for correlation when decoding.

Acknowledgments

We acknowledge helpful conversations with R.A. da Silveira and W. Bialek. This work was supported by grants from the NIH (EY014196) and the NSF (IIS-0613435). J.M. was supported by the German Ministry of Education, Science, Research and Technology through the Bernstein award to Matthias Bethge (BMBF; FKZ: 01GQ0601) as well as grants from the German Academic Exchange Service (DAAD) and the Boehringer Ingelheim Fonds (BIF).
Figure Captions

Figure 1. Maximum entropy model. A. Performance metrics for the means $E_{\text{mean}}$ (blue), covariances $E_{\text{covar}}$ (green), and pairwise correlation coefficients $E_{\text{corr}}$ (red) plotted versus the number of iterations of the MCMC algorithm. B. First- and second-order moments derived from the Ising model, $\{\langle r_i \rangle, \langle r_i r_j \rangle \}$, plotted versus first- and second-order moments sampled from the experimental data, $\{\langle r_i \rangle_{\text{emp}}, \langle r_i r_j \rangle_{\text{emp}} \}$ (blue: first-order; red: second-order). C. Log probability of response patterns given the maximum entropy model $P_{\lambda}(\bar{R})$ plotted versus log probability of response patterns sampled from the data $P_{\text{mix}}(\bar{R})$ (red: first-order (independent) model; blue: second-order (Ising) model). D. Matrix of pairwise interaction terms $\{J_{ij}\}$ between neurons $i$ and $j$ (color scale). E. Frequency of values of the bias terms $\{h_i\}$ (left) and interaction terms $\{J_{ij}\}$ (right).

Figure 2. Shape decoding experiment. A. The 38 shape stimuli used in the experiment. Shapes 1-36 have equal black area. Shapes 37 and 38 are smaller and larger squares, respectively. B. The receptive fields from the 162 cells recorded in the experiment. Ovals represent the 1-sigma contour of the two-dimensional Gaussian receptive field fits. A shape (the letter “E”) is overlaid at the scale and position at which it was presented. All shapes were centered at the same location. C. Total receptive field coverage of each point in visual space by the 162 ganglion cells in this study shown by color scale (right). D. Firing rates for 3 example cells responding to 3 different shapes. Time axis is time since shape onset. E. Raster plots across the whole recorded population for single presentations of three different shapes.

Figure 3. Population response and shape information. A. The population firing rate across all cells and all shapes. B. Average individual cell mutual information between
the binary response and shape identity in 100 ms bins (brown). Stacked bars (hotter
colors) are the extra information gained by using a set of spike count symbols for each
cell’s neural response that allow increasingly more spikes ($n_{max} > 1$). Percentages are
the information conveyed by binary neural responses (spike/no-spike) divided by the
information conveyed by multi-spike responses up to five spikes. Error bars are s.e.m.
for the binary representation, $n_{max} = 1$.

Figure 4. Population response matrix. A. Matrix of binary response probability in the
first time bin (100-200 ms after shape onset): vertical axis is for 38 shapes and
horizontal axis is for 162 retinal ganglion cells. Firing probability shown in color scale
(right). B. Response probability in the first time bin averaged over all 162 ganglion cells
and plotted versus shape identity. Notice that shapes 1-36 elicited roughly the same
number of spikes in the ganglion cell population, while shape 37 (small square)
triggered fewer and shape 38 (large square) triggered more spikes. C. Information
about shape identity for individual ganglion cells, $I(r;S)$, plotted against the cell’s firing
probability; each dot is for a different cell.

Figure 5. Performance of the cell count, independent, and mixture decoders. A,B.
Error rate, defined as false alarm rate at 99% hit rate criterion, for each shape using the
cell count decoder (A) and the independent decoder (B). Notice the different scales for
error rate; dotted line in (B) indicates mean performance of the cell count decoder. C.
Ratio of false alarm rate for the mixture decoder plotted against the false alarm rate of
the independent decoder. Each dot represents the result for a single target shape. Open
symbols indicate zero errors in the entire experiment. The error rate for these points is
set to 0.5 false alarms / number of distracter trials. D. Ratio of error rate for the
independent versus mixture decoders plotted against the error rate for the independent
decoder. Open symbols indicate zero measured false alarms for the mixture decoder.

Figure 6. Performance trends. A,B. Error rate for different population sizes for a
shape that reached zero error (A) and a shape having the median error (B) using both
independent and mixture decoders (purple vs. green). Each point represents the mean
error rate averaged over 100 different selections of a particular population size, $N$. C. Error rate plotted versus the number of shape stimuli for populations of 50 cells (thin lines) or 100 cells (thick lines) and for independent (purple) versus mixture (green) decoders. Results are averaged over 12 choices of the target shape each with 12 choices of distracter shapes as well as 25 random choices of cells. D. Error rate when using 35 shapes in the distracter ensemble plotted against the error rate when using only 10 shapes; results are for all 162 cells with either the independent (purple) or mixture (green) decoders.

**Figure 7. Visualizing the decoder's decision boundaries.** A. Plot of neural responses to the distracter ensemble (blue) and to the target stimulus (red) in a space reduced to three dimensions, corresponding to the projection of neural responses onto the first and second principal components (PC1 and PC2, respectively) as well as onto the vector defined by the independent decoder (linear discriminator). B, C. Plot of neural responses in a two-dimensional space with the decision boundary of the (independent, mixture) decoders (grey line). Some responses were correctly classified by the decoder as arising from either the target (red) or distracter (blue), while other responses were incorrectly classified (green). Notice that the independent decoder made many errors, while the mixture decoder only made one single error. D. Visualization of the decision surface for the mixture decoder on a color scale (blue-to-red indicates increasing likelihood of the target stimulus); notice the curved decision boundary (grey line).

**Figure 8. Comparing different response representations.** A. Illustration of how the nine different response representations are formed (described in Methods). B. Performance of response representations plotted as the geometric mean across shapes for two different error ratios: (1) on the y-axis, the error for the independent decoder divided by error for the mixture decoder for each representation; (2) on the x-axis, the error for the single time bin binary representation divided by the error for the given representation, both using the independent decoder. Bars are standard error of the mean.
Figure 9. The spike latency code. A. Error rate for the spike latency representation versus error rate for the binary representation, both using mixture decoders; each point is for a single target shape. Open symbols are for shapes with zero sampled errors. B. Error rate for mixture versus independent decoders, both using the spike latency representation. C. Error rate for the spike latency representation versus the binary representation, both using the independent decoder. D. Error rate averaged over all shapes as a function of the jitter added to the reference time on each trial.

Figure 10. Impact of noise correlation. A. Error rate for the mixture decoder applied to simultaneous spike trains vs. the error rate of the same decoder applied to shuffled spike trains using the spike latency representation for a single retinal patch (N = 25 cells); each symbol is for a single shape treated as the target stimulus. B. Effect of noise correlation vs. the effect of signal correlation for the spike latency representation; each symbol is for a single retinal patch (n = 13 patches). See text for definitions. C. Error rate for the mixture decoder applied to simultaneous spike trains vs. shuffled spike trains using the binary representation for a single retinal patch (N = 25 cells); each symbol is for a single shape treated as the target stimulus. D. Effect of noise correlation vs. the effect of signal correlation for the binary representation; each symbol is for a single retinal patch (n = 13 patches).

Figure 11. Optimal linear decoders. A. Error rate for the optimal linear classifier versus the independent decoder, both using the binary representation; each point is for a single target shape. B. Error rate for the mixture decoder versus the optimal linear classifier, both using the binary representation. C. Error rate for the tempotron decoder versus the independent decoder, both using the spike latency representation. D. Error rate for the mixture decoder versus the tempotron, both using the spike latency representation. Open symbols are for shapes that had zero sampled errors.

Figure 12. Maximum entropy decoder. A. Error rate for the maximum entropy decoder plotted against error rate for the independent decoder, both evaluated for the
binary response representation. B. Ratio of error for the maximum entropy decoder divided by the error for the mixture decoder plotted against error rate for the independent decoder, both evaluated for the binary response representation. C. Error rate for the maximum entropy decoder (red diamonds) and the hybrid mixture decoder (blue circles) plotted against error rate for the independent decoder, both evaluated for the spike latency response representation. D. Ratio of error for the maximum entropy decoder vs. the mixture decoder (red diamonds) and for the hybrid mixture decoder vs. the standard mixture decoder (blue circles) plotted against error rate for the independent decoder, both evaluated for the spike latency response representation. E. Schematic of population inputs integrated by a readout neuron that implements a linear decoder. F. Schematic of population inputs integrated by a readout neuron that implements a maximum entropy decoder using dendritic nonlinearities to capture pairwise interaction terms, $J_{ij}$.
REFERENCES


Figure 2 - Schwartz et al.
Figure 3 - Schwartz et al.

A

Shape ON

Shape OFF

B

Spikes Counted:
- $n_{max} = 1$
- $n_{max} = 2$
- $n_{max} = 3$
- $n_{max} = 4$
- $n_{max} >= 5$

Average Information (bits)

Time (s)
Figure 4 - Schwartz et al.
Figure 5 - Schwartz et al.

(A) Cell Count Decoder

(B) Independent Decoder

(C) Error Rate (Mixture) vs. Error Rate (Independent)

(D) Error Ratio (Ind. / Mix.) vs. Error Rate (Independent)
Figure 7 - Schwartz et al.

A. Population Data Projected into 3D

B. Independent Decoder in 2D

C. Mixture Decoder in 2D

D. Decision Surface for Mixture Decoder
Figure 8 - Schwartz et al.

A

spike train

<table>
<thead>
<tr>
<th>binary (1 if n&gt;0)</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>optimal threshold (1 if n&gt;2)</td>
<td>1</td>
</tr>
<tr>
<td>burst (0, 1, or 2+ spikes)</td>
<td>2</td>
</tr>
<tr>
<td>multi-spike</td>
<td>4</td>
</tr>
<tr>
<td>2 ⊗ 50 ms</td>
<td>3</td>
</tr>
<tr>
<td>4 ⊗ 25 ms</td>
<td>1</td>
</tr>
<tr>
<td>second time bin</td>
<td>1</td>
</tr>
<tr>
<td>first 200 ms</td>
<td>1</td>
</tr>
<tr>
<td>2 ⊗ 100 ms</td>
<td>1</td>
</tr>
</tbody>
</table>

Time from stimulus onset (s)

B

Error Ratio (Independent / Mixture)

<table>
<thead>
<tr>
<th>Error Ratio (Bin #1 Binary / Complex Representation)</th>
<th>1.0</th>
<th>1.5</th>
<th>2.0</th>
<th>2.5</th>
<th>3.0</th>
<th>3.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>bin #1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>opt. thres.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>burst</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>multi</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 ⊗ 50 ms</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 ⊗ 25 ms</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bin #2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 ⊗ 200 ms</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 ⊗ 100 ms</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 9 - Schwartz et al.

A. Mixture Decoder

B. Spike Latency

C. Independent Decoder

D. Average Error (Geometric Mean)
Figure 10 - Schwartz et al.
Figure 11 - Schwartz et al.

A

$Error\ Rate\ (Optimal\ Linear\ Classifier)$

$Binary$

B

$Error\ Rate\ (Mixture\ Decoder)$

Binary

C

$Error\ Rate\ (Tempotron\ Decoder)$

$Spike\ Latency$

D

$Error\ Rate\ (Mixture\ Decoder)$

$Spike\ Latency$
Figure 12 - Schwartz et al.

A) Error Ratio (Max. Ent. / Mixture)

B) Error Ratio (Approx. / Mixture)

C) Error Rate (Maximum Entropy)

D) Error Rate (Independent)

E) Linear Decoder

F) Maximum Entropy Decoder

\[ \sum_i w_i r_i > \theta \]

\[ \sum_i w_i r_i + \sum_{i,j} J_{ij} r_i r_j > \theta \]