Random Compressed Coding with Neurons

Simone Blanco Malerba

Mirko Pieropan*

Yoram Burak

Rava da Silveira[1]

[1]Laboratoire de Physique de l'Ecole Normale Supérieure, ENS, Université PSL, CNRS, Sorbonne Université, Université de Paris, Paris
[2]Racah Institute of Physics, Hebrew University of Jerusalem, Jerusalem
[3]Edmond and Lily Safra Center for Brain Sciences, Hebrew University of Jerusalem, Jerusalem
[4]Institute of Molecular and Clinical Ophthalmology Basel, Basel
[5]Faculty of Science, University of Basel, Basel

May 6, 2021

Abstract

The brain encodes the information about sensory world through the joint activity of neural populations. Classically, the mean response of neurons to parameters of sensory stimuli has been described through simple, unimodal or monotonic, smooth 'tuning curves'. Nevertheless, interesting coding properties emerge when considering complex response profiles. As an example, grid cells, with their spatially periodic responses, generate a precise combinatorial code, which allows them to represent a large range of locations with high accuracy, outperforming other spatial codes with unimodal tuning curves. Is periodicity necessary for enhanced coding, or similar properties emerge in other coding schemes? To address this question, we consider a simple circuit that produces complex but unstructured tuning curves, namely, a feedforward neural network with random connectivity, in which information is compressed from a first layer to a second one of smaller size. These irregular tuning curves represent richer 'sensors' of the stimulus (as compared to unimodal tuning curves), but may result in ambiguous coding which can yield catastrophic errors. Efficient coding implies an optimal point that specifies the spatial scale of tuning curve irregularities, as a function of the compression of the information between network layers and of the magnitude of noise affecting neural responses. By revisiting data from monkey motor cortex, we show how the tuning curves found in this area can be viewed as an instantiation of this 'compressed coding' scheme.

1 Introduction

Neurons convey information about the physical world by modulating their response as a function of parameters of sensory stimuli. Classically, the mean neural response to a stimulus (referred to as the neuron's 'tuning curve') is often described as a smooth function (of a stimulus parameter) with a simple monotonic or unimodal form [41, 38, 65, 54, 15, 23, 43]. The deviation from the mean response — the 'neural noise' — may lead to ambiguity in the identity or strength of the encoded stimulus, and the coding performance of a population of neurons as a whole is dictated by forms of the tuning curves and the joint neural noise. In the study of population codes, the efficient coding hypothesis has served as a theoretical organizing principle. It posits that tuning curves are arranged in such as way as to achieve the most accurate coding possible given a constraint

*Current affiliation: Department of Applied Science and Technology (DISAT), Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino on the neural resources engaged [9, 4, 51]. The latter is often interpreted as a metabolic constraint on the maximum firing rate of the single neuron or on the mean firing rate of the whole population [77, 11, 68].

In order to tackle this constrained optimization problem in practice, tuning curves are parametrized, and the corresponding parameters are optimized. Here is the point at which the simplicity of the form of tuning curves matters, as it generally results in a small set of parameters. A large body of literature addresses this constrained optimization problem, in particular in the perceptual domain. For example, many studies model tuning curves as Gaussian or other bell-shaped functions, and obtain the values of their means and variances that minimize the 'perceptual' error committed when information is decoded from the activity of a population of model neurons [77, 24, 73, 34, 31]. In the resulting optimal populations, and if noise among neurons is independent, the coding error typically scales like $1/\sqrt{N}$, where N is the number of model neurons [58]. (Tuning curves parameters govern the prefactor of this scaling. In some models where they can be further optimized, as is the case for the tuning width of neurons encoding one-dimensional stimuli, this scaling can be improved up to a factor of $\sqrt{N}$ [10, 46]). This behavior can be intuited simply based on the observation that the 'signal' in the neural population grows like N while the noise grows like $\sqrt{N}$, yielding a signal-to-noise ratio that increases in proportion to the square root of the population size.

Real neurons, however, can come with much more complex tuning curves than simple Gaussian or bell-shaped ones. The most salient example today is offered by grid cells in the enthorinal cortex [39, 25, 75, 45], which respond as a periodic function of spatial coordinates and hence display multi-modal tuning curves, but a number of other examples have also been noted in other cortical regions across species [42, 61, 50, 37, 28]. It was noted early on that such richer tuning curves can give rise to greatly enhanced codes. Given the periodicity of their tuning curves, and provided that the neural population includes several modules made up of cells with different periodicities [29, 69], grid cells can represent spatial location with an accuracy which scales exponentially (rather than algebraically, as above) in the number of neurons [62, 53, 17]. Thus, the richer structure of individual tuning curves can be traded for a strong boost in the efficiency of the population code.

Here, we ask whether highly efficient codes must rely on finely-tuned properties such as the tuning curves' periodicity or the arrangement of different modules in the population, or, by contrast, can arise more generically and robustly in populations of neurons with complex tuning curves. We approach the question by studying the benchmark case of a random code; specifically, a population code that relies on irregular tuning curves that emerge from a simple, feedforward, shallow network with random synaptic weights. The input layer in the network is made up of a large array of 'sensory' neurons with classical, bell-shaped tuning curves; these neurons project to a small array of 'representation' neurons with complex tuning curves. We show that, in the resulting population code, the coding error is suppressed exponentially with the number of neurons in the population, obtains robustly and is efficient even in the presence of high-amplitude noise. Here it is not sufficient to consider a 'typical error': efficiency results from the compression of the stimulus space in a layer of neurons of comparatively small size; the price to pay for this compression is the emergence of two qualitatively distinct types of error—'local errors', in which the encoding of nearby stimuli is ambiguous, and 'global (or catastrophic) errors', in which the identity of the true stimulus is lost altogether. The efficient coding problem then translates into a trade-off between these two types of errors. In turn, this trade-off yields an optimal width of the tuning curves in the 'sensory layer': when stimulus information is compressed into a 'representation layer', tuning curves in the sensory layers have to be sufficiently wide as to prevent a prohibitive rate of global errors.

We first develop the theory for a one-dimensional input (e.g., spatial location along a line or angle), then generalize it to higher-dimensional inputs. The latter case is more subtle because the sensory layer itself can be arranged in a number of ways (while still operating with simple, classical tuning curves). This allows us to apply our model to data from monkey motor cortex, where cells display complex tuning curves. We fit our model to the data and discuss the merit of a complex 'representation code'. Overall, our approach can be viewed as an application of the efficient coding principle to downstream ('representation') processing, as opposed to the more common applications to peripheral (sensory) processing. Our study extends earlier theoretical work on grid cells and other 'finely designed' codes by proposing that efficient compression of information can occur robustly even in the case of a random network. Our analysis is based on considering the geometric properties of neural activity in a downstream layer and how these vary with network parameters.

2 Results

We organize the discussion of our results as follows. First, we present, in geometric terms, the qualitative difference between a code that uses simple, bell-shaped tuning curves and one that uses more complex forms. Second, we introduce a simple model of a shallow, feedforward network of neurons that can interpolate between simple and complex tuning curves depending on the values of its parameters. Third, we characterize the accuracy of the neural code in the limiting case of maximally irregular tuning curves. Fourth, we extend the discussion to the more general case in which an optimal code is obtained from a trade-off between local and global errors. All the above is done for the case of a one-dimensional input space. In a fifth subsection, we generalize our approach to the case of a multi-dimensional stimulus. This then allows us to apply our model to recordings of motor neurons in monkey, and to analyze the nature of population coding in that system. Finally, we extend our model to include an additional source of noise—'input noise' in the sensory layer, in addition to the 'output nosie' present in the representation layer; input noise gives rise to correlated noise downstream, and we analyze its impact on the population code.

The geometry of neural coding with simple vs. complex tuning curves

A neural code is a mapping that associates given stimuli to a probability distribution of spiking patterns; in particular, the code maps any given stimulus to a mean population activity. In the case of a continuous, one-dimensional stimulus space, the latter is mapped into a curve in the N-dimensional space of the population activity, whose shape is dictated by the form of the tuning curves of individual neurons. As an illustration, we compare the cases of three neurons with bell-shaped (here, Gaussian) tuning curves vs. three neurons with periodic (grid-cells-like) tuning curves with three different periods (Fig. 1A). Simple tuning curves generate a smooth curve, implying that similar stimuli are mapped to nearby responses; by contrast, more complex tuning curves give rise to a serpentine shape. The latter makes better use of the space of possible population responses than the former, and hence can be expected to yield higher-resolution coding. Indeed, when the population response is corrupted by noise of a given magnitude, it will elicit a smaller *local* error in the case of complex tuning than in the case of simple tuning: by 'stretching' the mean response curve over a longer trajectory within the space of possible population activities, complex tuning affords the code with higher resolution relative to the range of the encoded variable. However, this argumentation does not capture in full the influence of noise on the nature of coding errors. In the case of a winding and twisting mean response curve, two distant stimuli are sometimes mapped to nearby activity patterns. In the presence of noise, this geometry gives rise to *global* (or catastrophic) errors. This enhanced resolution of the neural code associated with the occurrence of global errors was also noted in the context of grid cell coding [71, 62]. Because of this trade-off, whether a simple or complex coding scheme is preferable becomes a quantitative question, which depends upon the details of the structure of the encoding.

Shallow feedforward neural network as a benchmark for efficient coding

In order to address the problem mathematically, we examine the simplest possible model that generates complex tuning curves, namely a two-layer feedforward model. An important aspect of the model is that it does not rely on any finely-tuned architecture or parameter tuning: complex tuning curves emerge solely because of the variability in synaptic weights; thus, the model can be thought of as a benchmark for the analysis of population coding in the presence of complex tuning curves. The architecture of the model network and the symbols associated with its various parts are illustrated in Fig. 1B. In the first layer, a large population of L *sensory* neurons encodes a one-dimensional stimulus, x, into a high-dimensional representation. Throughout, we assume that x takes values between zero and one, without loss of generality. (If the input covered an arbitrary range, say r, then the coding error would be expressed in proportion to r. In other words, one cannot talk independently of the range and of the resolution of a code. We set the range to unity in order to avoid any ambiguity.) Sensory neurons come with classical tuning curves: the mean firing rate of neuron j in response to stimulus x is given by a Gaussian with center c_j (the preferred stimulus of that neurons) and width σ:

$$u_j\left(x\right) = A\exp\left(-\frac{\left(x-c_j\right)^2}{2\sigma^2}\right). \tag{1}$$

Following a long line of models, we assume that the preferred stimuli in the population are evenly spaced, so that $c_j = j/L$. As a result, the response vector for a stimulus x_0, $\mathbf{u}\left(x_0\right)$, can be represented as a Gaussian 'bump' of activity centered at x_0.

Complex tuning curves appear in the second layer containing N *representation* neurons; we shall be interested in instances with $N \ll L$, in which efficient coding results in compression of the stimulus information from a high-dimensional to a low-dimensional representation. Each representation neuron receives random synapses from each of the sensory neurons; specifically, the elements of the all-to-all synaptic matrix, $\mathbf{W}$, are i.i.d. Gaussian random weights with vanishing mean and variance equal to $1/L$ ($W_{ij} \sim \mathcal{N}\left(0, 1/L\right)$). In the simple, linear case that we consider, the mean response of neuron i in the second layer is this given by

$$v_i\left(x\right)=\sum_{j=1}^{L}W_{ij}u_j\left(x\right). \tag{2}$$

Since the weights W_{ij} correspond to a given realization of a random process, they generate tuning curves, $v_i\left(x\right)$, with irregular profiles. The parameter σ is important in that it controls the smoothness of the tuning curves in the second layer: it defines the width of u_j, which in turns dictates the correlation between the values of the tuning curve v_i for two different stimuli. By the same token, the amplitude of the variations of v_i with x depends upon the value of σ. For a legitimate comparison of population coding for different networks, we fix this amplitude to a constant,

$$\left\langle\int_0^1 dx\left[v_i\left(x\right)-\int_0^1 dx' v_i\left(x'\right)\right]^2\right\rangle_W=R, \tag{3}$$

by choosing the value of the prefactor in Eq. (1), A. The average over the synaptic weights, indicated by the brackets $\langle\cdot\rangle_W$, makes A independent from the specific realization of the weights. This constraint corresponds to the usual constraint of ‘resource limitation’ in efficient coding models; it amounts to setting a maximum to the variance of the output over the stimulus space, as is commonly assumed in analyses of efficient coding in sensory systems [4, 66, 26, 78].

Returning to our geometric picture, we observe that, by changing the value of σ, we can interpolate between smooth and irregular tuning curves in the second layer (Fig. 1C). In the limiting case of large σ, representation neurons come with smooth tuning curves akin to classical ones; in the other limiting case of small σ, the mean response curve becomes infinitely tangled. Thus, as the value of σ is decreased, the mean responses curve ‘stretches out’ and winds in such a way as to fit within the allowed space of population responses defined by Eq. (3). A longer mean response curve fills the space of population responses more efficiently and represents the stimulus at a higher resolution, but its twists and turns may result in greater susceptibility to noise.

To complete the definition of the model, we specify the nature of the noise in the neural response. We assume that neuron i in the second layer is affected by i.i.d. noise, which we denote z_i, such that its response at each trial is given by $r_i=v_i\left(x\right)+z_i$. For the sake of simplicity, we use Gaussian noise with vanishing mean and variance equal to η^2. In most of our analyses, we suppose that responses in the first layer are noiseless and that the noise in the second layer is uncorrelated among neurons; in the last subsection, however, we relax these assumptions, and discuss the implications of noisy sensory neurons and correlated noise in representation neurons. (Our motivation for considering noiseless sensory neurons is that we are primarily interested in analyzing the compression of the representation of information between the first and the second layer of neurons. By contrast, noise in sensory neurons affects the fidelity of encoding in the first layer.) We quantify the performance of the code in the second layer through the mean squared error (MSE) in the stimulus estimate as obtained from an ideal decoder. The use of an ideal decoder is an abstract device that allows us to focus in the uncertainty inherent to *encoding* (rather than to imperfections in *decoding*); it is nevertheless possible to obtain a close approximation to an ideal decoder in a simple neural network with biologically plausible operations, as we show in Methods.

Compressed coding in the limiting case of narrow sensory tuning

It is instructive to study coding in our model in the limiting case of narrow tuning in the sensory layer, with $\sigma\ll 1$ ($\sigma\to 0$), because this limit yields the most irregular tuning curves in the representation layer of our network (Fig. 1C). As we will see, this limiting case also corresponds to that of a completely uncorrelated, random code, for which the mathematical analysis simplifies. When the value of σ is much smaller than $1/L$, each sensory neuron responds only if the stimulus coincides with its preferred stimulus; stimulus values that lie in between the preferred stimuli of successive sensory neurons in the first layer do not elicit any activity in the system. We can thus consider that any stimulus of interest is effectively chosen in a discrete set of L stimuli with values $x_j=j/L$, with $j=1,\ldots,L$.

Each of these stimuli elicits a mean response

$$v_i(x_j)=\tilde{A}W_{ij}\sim\mathcal{N}(0,R) \tag{4}$$

in neuron i of the second layer. Here, $\tilde{A}$ is chosen by fixing the amplitude of the variations of v_i, obtained from Eq. (3) in case of discrete stimuli. Geometrically, this corresponds to mapping L stimulus values to a set of uncorrelated, random locations in the space of population activity vectors that correspond to the mean responses (as illustrated in Fig. 2A for a two-neuron population). In any given trial, the response of the representation layer is corrupted by noise that takes it away from the corresponding mean response (Fig. 2A).

The ideal decoder (here, 'ideal' means that it minimises the mean error) interprets a single-trial response as being elicited by the stimulus associated to the nearest possible mean response (Fig. 2A). The outcome of this procedure can be twofold: either the correct or an incorrect stimulus is decoded; in the latter case, because the possible mean responses are arranged randomly in the space of population activity (Fig. 2A and Eq. (4)), errors of any magnitude are equiprobable. As a result, in a model with narrow sensory tuning curves which results in a second-layer representation that does not preserve distances among inputs, the decoding error is either vanishing or, typically, on the order of the input range (set to unity here). The mean error can then simply be equated to the probability with which the ideal decoder makes a mistake.

In Methods, we provide a derivation of this quantity in the case where it is much smaller than 1. We obtain the dependence of the probability of making a decoding error as a function of the various model parameters, as

$$P_{\text{error}} \approx \frac{L}{\sqrt{2\pi N}} \exp\left(-\log\left(1+\frac{R}{2\eta^2}\right)\frac{N}{2}\right). \tag{5}$$

The main dependence to note, here, is the exponentially strong suppression as a function of the number of neurons in the second layer (Fig. 2B). By contrast, the probability of error scales merely linearly with the size of the stimulus space, L, as is expected in a limit of small probability of error. This result implies that it is possible to compress information highly efficiently in a comparatively small representation layer ($N \ll L$) even though the code is completely random. The price to pay for this randomness is that any given error is 'catastrophic' (on the order of L), but these large errors happen prohibitively rarely. It is also worth noting that the rate of exponential suppression depends on the variance of the noise, η^2, or, more precisely, on the single-neuron signal-to-noise ratio, R/η^2 (where R is the variance of the signal, Eq. (3)). In numerical simulations, we set $R = 1$ and we vary η^2 to explore different noise regimes. Interestingly, even when this signal-to-noise ratio becomes small, i.e., when the noise in the activity of individual neurons is comparable to modulations of their mean responses, the exponential suppression of the probability of error remains valid, with a rate approximately equal to $R/4\eta^2$.

Compressed coding with broad tuning curves: trade-off between local and global errors

As we saw in the previous section, in the case of infinitely narrow tuning curves the coding of a stimulus in a given trial is either perfect or indeterminate; that is, any error is a global error, on the order of the entire stimulus range. In the more general case of sensory neurons with arbitrary tuning width, the picture is more complicated: in addition to *global* errors which result from the winding and twisting of the mean response curve, the population code is also susceptible to *local* errors (Fig. 3A). This is because broad tuning curves in the sensory layer partly preserve distances: locally, nearby stimuli are associated with nearby points on the mean response curve; as a result, the coding of any given stimulus is susceptible to local errors due to the response noise. As the tuning width in the sensory layer, σ, decreases, two changes occur in the mean response curves: it becomes longer (it 'stretches out') and it becomes more windy (Fig. 1C). Stretching increases the local resolution of the code (because it allows for two nearby stimuli to be mapped to two more distant points in the space of population activity), while windiness increases the probability of global errors. This trade-off is apparent when we plot the histogram of coding-error magnitudes as a function of σ: for larger values of σ, global errors are less frequent, but local errors are boosted (Fig. 3B). Also noticeable, here is that the large-error tails of the histograms are flat, consistent with the observation that global errors of all sizes are equiprobable. (Strictly speaking, this happens if the stimulus has periodic boundary conditions, such that, picking two random points, the probability that they are at a given distance is constant for all distances.)

For a more quantitative understanding, we carried out an approximate calculation, in which (i) we approximated the mean response curve by a linear function locally and (ii) considered that the distance between two segments of the curve containing the mean response to two stimuli distant by more than σ is sampled randomly. Using these two assumptions, we obtained the MSE as a sum of two terms (see Methods for mathematical details), as

$$\varepsilon^2 = \left\langle E^2 \right\rangle_W \approx \left\langle E_l^2 \right\rangle_W + \left\langle E_g^2 \right\rangle_W \approx \frac{2\sigma^2\eta^2}{RN} + \frac{1}{\sigma\sqrt{2\pi N}}\bar{\varepsilon}_g \exp\left(-\log\left(1+\frac{R}{2\eta^2}\right)\frac{N}{2}\right), \tag{6}$$

where $\bar{\varepsilon}_g$ is a term of $\mathcal{O}(1)$ that depends upon the choice of stimulus boundary conditions (see Methods). This expression quantifies the MSE for a 'typical' network, obtained by averaging over possible choices of synaptic weights, as indicated by the brackets $\langle\cdot\rangle_W$. The first term on the right-hand-side of Eq. (6) represents the contribution of local errors, while the second term corresponds to global errors (Fig. 3C). Their form can be intuited as follows. The variance of local errors is proportional to σ^2 and inversely proportional to N, as in classical models of population coding with neurons with bell-shaped tuning curves (see, e.g., [23]). Furthermore, decreasing σ stretches out the mean response curve, which increases the local resolution of the code and explains the factor σ^2 in Eq. (6). (The form of this first term can also be understood as the inverse of the Fisher information [58, 16], which bounds the variance of the error.) The second term on the right-hand-side of Eq. (6) is obtained as an extension of Eq. (5): instead of considering the probability that two mean response points are placed nearby, we consider the probability that two segments of the mean response curve with size σ each fall nearby. There are $1/\sigma$ such segments (since we have set the stimulus range to unity), and this explains why the factor L in Eq. (5) is replaced by a factor $1/\sigma$ in Eq. (6). Importantly, the two terms in Eq. (6) are modulated differently by the two parameters N and σ. Depending upon their values, either local or global errors dominate (Fig. 3C). We tested the validity of Eq. (6): it agrees closely with results from numerical simulations, in which we computed the MSE using a Monte Carlo method and a network implementation of the ideal decoder (Fig. 3D, see Methods for details). The non-trivial dependence is illustrated by the observations that the MSE error may decrease or increase as a function of σ, around a given value of σ, depending upon the value of N (Fig. 3E). Furthermore, the strong (exponential) reduction in MSE with increasing N occurs only up to a crossover value depending on σ (Fig. 3F); beyond this value, global errors disappear, and the error suppression is shallower (hyperbolic in N, due to improved local resolution). For small values of σ, the crossover values of N are larger and occur at lower values of the MSE.

As is apparent from Figs. 3D and E, for any value of N there exists a specific value of $\sigma = \sigma^*(N)$ that balances the two contributions to the MSE such as to minimize it. This optimal width can be thought as the one that stretches out the mean response curve as much as possible to increase local accuracy but that stops short of inducing too many catastrophic errors. This optimum is obtained from Eq. (6). The MSE is asymmetric about the optimal width, σ^*: smaller values of σ cause a rapid increase of the error due to an increased probability of global errors, while larger values of σ mainly harm the code's local accuracy, resulting in a milder effect. From Eq. (6), we obtain the dependence of the optimal width upon the population size as

$$\sigma^* \approx \left(\frac{\bar{\varepsilon}_g}{4\eta^2}\sqrt{\frac{N}{2\pi}}\right)^{1/3} \exp\left(-\log\left(1+\frac{R}{2\eta^2}\right)\frac{N}{6}\right), \tag{7}$$

and the optimal MSE as a function of N,as

$$\varepsilon^{2*} = \langle E^2(\sigma^*)\rangle_W \approx \left(\frac{\eta\bar{\varepsilon}_g}{\sqrt{2\pi N}}\right)^{2/3} \exp\left(-\log\left(1+\frac{R}{2\eta^2}\right)\frac{N}{3}\right). \tag{8}$$

Both these analytical results agree closely with numerical simulations (Figs. 4A and B). Equation (8) and Fig. 4B show that the optimal MSE is suppressed exponentially with the number of representation neurons in the second layer. Thus, highly efficient compression of information and coding also occurs when tuning curves in the sensory layer are not infinitely narrow. The rate of this scaling depends upon the noise variance, η^2; in Figs. 4C and D, we illustrate the dependence of σ^* and $\left\langle E^2(\sigma^*)\right\rangle_W$ upon N and η^2.

Compressed coding of multi-dimensional stimuli

Real-world stimuli are multi-dimensional. Our model can be extended to the case of stimuli of dimensions higher than one, but particular attention should be given to the nature of encoding in the first layer—because sensory neurons can be sensitive to one or several dimensions of the stimuli. In one limiting case, a sensory neuron is sensitive to all dimensions of the stimulus; for example, place cells respond as a function of the two- or three-dimensional spatial location. Visual cells constitute another example of multi-dimensional sensitivity, as they respond to several features of the visual world; for example, retinal direction-selective cells are sensitive to the direction of motion, but also to speed and contrast. In the other limiting case, sensory neurons are tuned to a single stimulus dimension, and insensitive to others. We will refer to these two coding schemes as *conjunctive* and *pure*, following Ref. [30] (where they are explored in the context of head-direction neurons in bat). The authors conclude that the relative advantage of a pure coding scheme—with neurons that encode a single head-direction angle—with respect to a conjunctive coding scheme—with neurons that encode two head-direction angles—depends on specific contingencies, such as the decoding time window or the population size. Indeed, in a conjunctive coding scheme individual neurons carry more information, but the population as a whole needs to include sufficiently many neurons to cover the (multi-dimensional) stimulus space—a constraint which becomes stronger as the number of dimensions increases.

We generalized our model to include the possibility of K-dimensional stimuli. For the sake of simplicity, we consider only the two limiting cases of *conjunctive* and *pure* coding in the *sensory* layer of our model (i.e., we do not discuss intermediate cases, in which a given sensory neuron is sensitive to several but not all stimulus dimensions, see Methods). In our model, furthermore, neurons in the *representation* layer receive random inputs from all sensory neurons; as such, the representation layer embodies a conjunctive coding scheme. By using our geometric picture, we illustrate the differences between the two coding schemes in the case of a two-dimensional stimulus space (Fig. 5A). This can be visualized as a planar 'sheet', which is mapped by the network, into a two-dimensional surface in the N-dimensional space of the population activity. In the pure case, the activity of a single sensory neuron is maximally modulated when the stimulus changes across a certain direction, which corresponds to its preferred dimension. Stimulus variations in the perpendicular directions will have no effect on the neural activity. As a result, in the representation layer, which randomly sum the neural responses of the sensory neurons, there will be many directions in which stimulus variations weakly modulate the neural activity. In the N-dimensional space, the surface will look 'folded' in these directions of low sensitivity. By contrast, in the conjunctive case the activity of sensory neurons is modulated by stimulus changes across all directions. When we consider the joint activity of neurons in the second layer, the resulting surface will be randomly curved in all directions, and will look more similar to a 'crumpled' sheet.

We can qualitative explain the different behavior of the two types of errors through this geometrical picture. (See Methods for detailed analytical calculations.) The local error is determined by the curvature of the surface, measuring the variation as a function of local stimulus changes. In the pure case, the folding directions are characterized by a low curvature, implying a lower resolution. As for comparison, the conjunctive scheme produces a surface curved in all directions, and this leads to a lower local error (Eq. (54)-Eq. (58)). Moreover, in the pure scheme, due to the folds of the surface, global errors are most likely to affect a single stimulus dimension each time. Since we fixed the amplitude of variations of neural responses across all the stimulus space, the variation across a single dimension is $1/K$ of the total one. This leads to a signal-to-noise ratio, governing the rate of exponential scaling of global errors as a function of the number of neurons, which is reduced by a factor of Kin the pure case. This reduction is absent in the conjunctive case, where global errors can happen across all stimulus directions, and the scaling is governed by the variations across the whole stimulus space (Eq. (55)-Eq. (59)).

We discuss numerical results for the comparison of the two schemes in the specific case where $K = 3$, which, in case of conjunctive sensory neurons, will allow us to model a real biological circuit in the next section. The behavior of the MSE as a function of the tuning width is similar to the one-dimensional case. In both schemes, there exists an optimal σ which achieves a balance between the two contributions to the error, and it decreases as N increases (Fig. 5B). In order to quantify the relative advantage of the pure sensory scheme with respect to the conjunctive sensory scheme as a function of the two parameters N and σ, we plotted the ratio between the Root-MSE in the two cases (Fig. 5C). It is possible to distinguish different regimes of relative advantage, depending on the joint value of the two parameters. If the population size is small, the pure scheme slightly outperforms the conjunctive one. It has to be noticed that this is a regime where the errors are typically global, and therefore very large. As soon as N increases, the contribution of the local errors becomes more relevant. If the tuning curves are broad (large σ), and N is large, such that global errors are strongly suppressed, the errors are mainly local. In this case, the conjunctive scheme outperforms the pure one, with the ratio between Root-MSE approaching the theoretical value of $1/\sqrt{3}$. When the tuning width is narrowed, and we are in a regime of population size where the two types of errors have a comparable weight, the advantage of the conjunctive scheme is further increased. As predicted before, this is due to a stronger suppression of global errors as a function of N in the conjunctive case. Finally, when the tuning width is further narrowed below a certain value, which depends on the size of the first layer, the conjunctive sensory neurons are no more able to cover the stimulus space. This leads to large errors, independently from the size of the second layer, and a better performance of the pure scheme, which tiles the space more efficiently.

The different scaling of the optimal width and the relative error is determined by this relative advantage of one scheme with respect to the other (Fig. 5D and E). Both quantities decrease more rapidly in the conjunctive scheme. In this case the suppression of global errors as a function of the number of neurons is stronger, and therefore a narrower σ, yielding a lower local error, is preferable. Nevertheless, in the conjunctive case the lower bound to the optimal σ imposed by the size of the first layer is also higher. This limits the regime of exponential scaling of the optimal error to a smaller interval with respect to the pure case.

Compressed coding in monkey motor cortex

Neurons in the primary motor cortex (M1) of monkey are sensitive to space- and movement-related parameters. We consider here spatial tuning observed in recordings carried out during a 'static task' [44]. In this task, a monkey is cued to remain motionless during a given delay while having placed its hand at one of a number of preselected positions on a three-dimensional grid, defined by the vector $\mathbf{x} = \{x_1, x_2, x_3\}$. Recordings show that

M1 neurons exhibit tuning curves as functions of hand location [44, 67]. It has been customary to model these tuning curves as varying linearly with a combination of the spatial coordinates of the hand,

$$v_i(\mathbf{x}) = a_i + b_{1,i}x_1 + c_{2,i}x_2 + d_{3,i}x_3 = v_i(\mathbf{x}) = a_i + \mathbf{P}_i \cdot \mathbf{x}, \tag{9}$$

where i indexes the M1 neuron and $\mathbf{P}_i$, sometimes called 'preferred vector' or 'positional gradient', is a vector pointing along the direction of maximal sensitivity [67]. A recent study [50] observed, however, that a model of tuning curves that includes a form of irregularity yields an appreciably superior fit to the simple linear behavior of Eq. (9). This more elaborate model [50] bears similarity with our model of irregular tuning curves, and this naturally led us to ask about potential coding advantages that a complex coding scheme may afford in M1.

To be more specific, one can interpret here the first layer in our network, featured with neurons with three-dimensional Gaussian tuning curves, as representing neurons in the parietal reach area (or premotor area), which are known to display spatially localized tuning properties [2]. This population of neurons projects to a smaller population of M1 neurons which display spatially extended and irregular tuning profiles. In fitting our model to the M1 recordings [50], we considered the arrangement of stimuli used in the experiment, namely 27 spatial locations arranged in a $3\times3\times3$ grid in a 40 cm-high cube. We then followed a previous approach [50, 3] : given the diversity of the irregular tuning curves in the population we did not aim at fitting individual tuning curves; instead, we allowed for randomly distributed synaptic weights (as in our original model) and we fitted a single parameter, the width of the tuning curves in the first layer, σ. The fit was aimed at reproducing specific summary statistics of the data referred to as *complexity measure* (discrete version of the Lipschitz derivative that quantifies the degree of smoothness of a curve, see Methods and Ref. [50]). The complexity measure varies from neuron to neuron, and we chose σ so as to minimize the Kolmogorov-Smirnov distance between the distribution implied by our model and the one extracted from the data. While our model is somewhat simpler than a model of irregular M1 tuning curves employed previously [50], it yields comparable fits. In addition to fitting the population of tuning curves, we extracted from the data a quantification of the noise in the response of individual neurons. For each recorded neuron, we computed the variance of the signal as the variance, across different stimuli, of the mean firing rate (left hand side of Eq. (3)). Then, we estimated the variance of the noise by averaging the trial-to-trial variability of responses to the same stimulus. These two quantities allowed us to define a signal-to-noise ratio for each neuron of the population, Eq. (62). As in simulations we fixed the variance of the signal for each neuron to a constant, the heterogeneity of the signal-to-noise ratio is modeled as a heterogeneous noise variance.

With a neural response model in hand, we can evaluate the coding performance; to do so, we consider a finer, $21\times21\times21$ grid of spatial locations as our test stimuli. We quantify the merit of a compressed code making use of irregular tuning curves by computing the MSE, $\varepsilon^2_{\text{irr}}$, and comparing the latter with the corresponding quantity in a coding scheme with smooth tuning curves as defined in Eq. (9), $\varepsilon^2_{\text{lin}}$. We plot our results in terms of the 'mean percent improvement', $\Delta\varepsilon \equiv (\varepsilon_{\text{lin}} - \varepsilon_{\text{irr}})/\varepsilon_{\text{lin}}$. $\Delta\varepsilon$ is positive when irregularities favor coding, and at most equal to one (in the extreme case in which irregularities allow for error-free coding).

We explore the performance of the two coding schemes for different values of the parameters N and σ, in an ideal case in which all neurons have the same noise variance (Fig. 6A). We note the existence of a crossover value of N, N^*. When $N < N^*$, small values of σ induce prohibitively frequent global errors in the compressed (irregular) coding scheme, and linear tuning curves are more efficient. For $N > N^*$, however, irregularities are always advantageous, and the more so the smaller the value of σ. Because global errors are suppressed exponentially with N, N^* typically takes a moderate value (which depends upon the magnitude of the noise); the larger the noise, the larger N^*. Figure 6B illustrates this noise-dependent behavior of the crossover population size, for the best-fit value of σ (~ 23).

For a more realistic modeling of M1 neurons, we analyzed the performance of a model in which each neuron's noise variance is extracted from the data (Figs. 6C and D). The distribution of noise variances in the population is heterogeneous, and yields a fraction of neurons with low signal-to-noise ratios (Fig. 6C, inset). For each value of N, we sampled eight different pools of N neurons from the population, and we averaged the corresponding mean percent improvement, $\Delta\varepsilon$. We found, again, that the relative merit of compressed coding (with irregular tuning curves) grows with the population size; interestingly, when compressed coding becomes advantageous ($\Delta\varepsilon > 0$ in Fig. 6C), the MSE is still appreciable (Fig. 6D). This means that even though local and global errors are balanced, both occur with non-negligible likelihood. $\Delta\varepsilon$ continues to grow with N until global errors are suppressed; beyond this second crossover value, N_{local}, $\Delta\varepsilon$ saturates because in both coding schemes (with irregular and linear tuning curves) local errors dominate. Correspondingly, the MSE scales differently for N above or below N_{local}. When $N < N_{\text{local}}$ the MSE decreases exponentially with N, due to the suppression of global errors, while when $N > N_{\text{local}}$, the suppression of the MSE is hyperbolic in N, reflecting the behavior of local errors only (Fig. 6D). Interestingly, this second crossover occurs at $N_{\text{local}} \approx 100$, a figure comparable to the number of neurons that control individual muscles in this specific task, as estimated from decoding EMG signals from individual muscles from subsets of M1 neurons [50].

Compressed coding with noisy sensory neurons

Until now, we have considered the presence of response noise only in second-layer neurons. In this case, as long as sensory neurons are tiling the stimulus space (i.e., unless there are regions in stimulus space in which sensory neurons are strictly unresponsive), stimuli are encoded with perfect accuracy in the activity of the first layer, and the MSE in the second layer can be made arbitrarily small for sufficiently large N. If sensory neurons are also noisy, then they represent stimuli only up to some degree of precision. Furthermore, because of the (non-sparse) projection from the first to the second layer of neurons, independent noise in sensory neurons induces correlated noise in representation neurons. If the independent noise in sensory neurons is Gaussian with variance equal to ξ^2, then the covariance of the noise in the second layer becomes $\Sigma = \eta^2\mathbf{I} + \xi^2\mathbf{W}\mathbf{W}^\mathbf{T}$. Thus, sensory noise affects the nature of the 'representation noise', and it is natural to ask how this changes the population coding properties.

As we shall show, in the compression regime ($N \ll L$) on which we focus, the kind of correlations generated by sensory noise have a negligible effect on the coding performance. Obviously, the introduction of sensory noise degrades coding, so the comparison of the noisy and noiseless systems is not very telling. Instead, we compare population coding in the presence of the full covariance matrix, Σ, and in the presence of a diagonal covariance matrix. By considering the distribution of the synaptic weights, the matrix $\mathbf{W}\mathbf{W}^\mathbf{T}$ follows a Wishart distribution with mean the identity matrix (see Methods); therefore we defined the average variance-matched diagonal covariance matrix as $\Sigma_{\text{ind}} = \left(\eta^2 + \xi^2\right)\mathbf{I}$. The latter corresponds to a network with noiseless sensory neurons, but enhanced independent noise in representation neurons, with variance $\tilde{\eta}^2 \equiv \eta^2 + \xi^2$. In numerical studies, we observe, first, that the MSE depends only weakly on the noise correlations, as a function of σ. This behavior obtains because noise correlations affect primarily local errors, not global errors. In theory, one could argue that noise correlations reduce the noise entropy, shrinking the volume of the cloud of possible responses, with respect to the diagonal case, and this should reduce the probability of having global errors. Nevertheless, in numerical simulations this effect is negligible; this is probably due to the random, and usually large, magnitude of global errors. On the other hand, local errors can be either suppressed or enhanced by correlated noise [22].

We can show analytically that, here, local errors are enhanced; from a perturbative expansion of the inverse covariance matrix (see Methods for details), we obtained that the local contributions to the MSE in orders of $\xi^2/\tilde{\eta}^2$ is given by

$$\varepsilon_l^2 = \varepsilon_{l,\text{ind}}^2\left(1 + \frac{N\xi^2}{L\tilde{\eta}^2} - \frac{N\xi^4}{L\tilde{\eta}^4} + \dots\right), \tag{10}$$

where $\varepsilon_{l,\text{ind}}^2$ is the corresponding quantity calculated for the variance-matched covariance matrix Σ_{ind} rather than the full covariance matrix Σ. From Eq. (10), it appears that the effect of noise correlations on the MSE is deleterious but scales only weakly with $N/L \ll 1$. We checked this behavior numerically (Fig. 7A), and found a good match with the analytical result. We also compared the impact of different values of ξ^2, while keeping the effective noise variance, $\tilde{\eta}^2$, fixed (i.e., varying the relative contribution of input and output noise). Both Eq. (10) and Fig. 7B indicate that there exists a regime in which increasing ξ^2 in fact mitigates the deleterious effect of the correlated noise (this is seen in Eq. (10) as a partial cancelation of the second- and fourth-order terms).

Finally, we ask whether the impact of the noise correlation results specifically from the form with which sensory noise invests it. To answer this question, we examine a network with noiseless sensory neurons, but in which representation neurons exhibit correlated Gaussian noise, with a covariance matrix that has the same statistic as those of Σ, but in which the form of correlations is not inherited from the network structure through the synaptic matrix $\mathbf{W}$; specifically, we consider a random covariance matrix, $\Sigma_{\text{rand}} = \eta^2\mathbf{I} + \xi^2\mathbf{X}\mathbf{X}^\mathbf{T}$, where $X_{ij} \sim \mathcal{N}(0, 1/L)$. In this case, noise correlations *suppress* the MSE as compared to the independent case (with Σ_{ind}), because the 'cloud' of possible noisy responses is reoriented randomly with respect to the curve of mean responses. Analytically, the analog of Eq. (10) for the case of covariance matrix given by Σ_{ind} is similar, but skips the lowest-order, deleterious term:

$$\varepsilon_{l,\text{rand}}^2 \approx \varepsilon_{l,\text{ind}}^2\left(1 - \frac{N\xi^4}{L\tilde{\eta}^4}\right). \tag{11}$$

This result, as well as numerical simulations (Fig. 7B), demonstrates that generically coding is improved by random noise correlations, and that this improvement increases with N and with ξ^2. In sum, noise correlations in representation neurons are deleterious if they are inherited the noise in sensory neurons—yet, the effect is quantitatively modest.

3 Discussion

Summary. We analyzed the coding properties of neural populations beyond classical models of tuning curves, by considering irregular response profiles resulting from random feedforward connectivity. Our model can interpolate between an irregular coding scheme, locally accurate but prone to catastrophic errors, and a smooth one, more robust to noise. Optimality is achieved at an intermediate level of irregularity, which depends on the population size and on the variance of the noise. Remarkably, at this optimal configuration, the error is strongly suppressed, i.e., exponentially, as the population size increases. As a result, we are able to compress the information about the stimulus using the activity of a neural population of relatively small size, with respect to the first layer one. We extended these results to the case of multi-dimensional stimuli, by illustrating the differences arising when the first layer neurons exhibit pure or conjunctive selectivity to stimulus dimensions. We showed how the relative advantage of one scheme with respect to the other, a question recently explored in the context of sensory neurons [30, 40], depends on the number of representation neurons and on the tuning parameters. Finally, we applied the extension of our model to the case of three-dimensional stimuli to recordings of motor cortex neurons in monkey [50], where we illustrated the advantage of irregularities in spatial tuning curves for the accurate coding of hand position.

Exponentially strong population codes. Our results about the exponential scaling of the error with the population size are similar to what found in models analyzing the coding of position by grid cells[29, 62, 53, 69]. Using the terminology of these papers, the random coding scheme of neurons in the second layer is an *exponentially strong* population code. Our work shows that grid cells are a particular biological realization, characterized by a highly structured organization, of a more general class of coding schemes whose general principles were already introduced by Shannon. In [60], he introduced a geometrical representation of a general communication system, as a map between points in the space of *messages* (which in our case correspond to stimuli) to points in the space of *signals* (patterns of neural activity), which are then decoded by the receiver. As a specific case, he considered the problem of encoding a one-dimensional, continuous message space, into signals of higher dimension. Geometrically, this corresponds to map a one-dimensional curve into a higher dimensional space, similarly to what is done by our network (Fig. 1C, and Fig.9A). In order for such a map to be efficient, the region of uncertainty created by the noise should be small as compared to the total length of the line (high dynamic range). This is achieved by increasing the length of the line as much as possible, winding and twisting it such to fill the available signal space. Nevertheless, the magnitude of the noise puts a limit on much the signal space can be filled before having large ambiguities in the represented messages, a phenomenon that Shannon called *threshold effect* (which corresponds to global errors in our case).

Astonishingly, he showed that such signal space filling map, need not to be carefully designed: a map which associates randomly points in the message space to points in the signal space achieves the maximal capacity of a system. At this maximal capacity, it is possible to send, with arbitrarily small error rate, unambiguous information about a number of messages which is exponential in the dimensionality of the signal. In our work, this fully random scheme is equivalent to the case of extremely narrow tuning width (Fig. 2). In addition, we were able to control the smoothness of this random map (i.e., how far similar stimuli evoke similar responses) through a parameter, σ. In this more general coding scheme, optimality, if the population size is not large enough (or the variance is too high), is achieved with a finite level of smoothness, such to trade a lower local accuracy for less catastrophic errors.

Coding with complex tuning curves. A large body of literature has addressed the problem of coding low-dimensional stimuli in populations of neurons with simple tuning curves. The most common assumption is that of bell-shaped tuning curves which have been used to examine sensory coding in peripheral neurons. The optimal tuning curve width was studied as a function of population size for different stimulus dimensionality [77], stimulus geometry [55] and time scale of coding [11, 73]. Recent work analyzed the influence of a non-uniform prior distribution over stimuli on the optimal distribution of the tuning parameters across the population; in particular, it was predicted that the tuning width should be narrower for neurons encoding a stimulus with higher prior probability [70, 34, 76]. On the other hand, the effects of random heterogeneity in the tuning parameters on the coding performance was also studied [72, 59, 31]. In this kind of populations, a single neuron is substantially informative about the stimulus, although the best-encoded region of stimulus space may vary from the region of maximal slope (flanks of the tuning curve) or the region of maximal response (peak), depending on the noise level [18, 74].

In this paper, we followed this line of work with two relevant differences. We considered coding properties of neurons with irregular, multi-peaked tuning profiles, and which are *deep*, as they are one-synapse distant from the sensory layer. As a result, the optimal tuning properties of the first layer neurons (e.g., tuning width, but also selectivity to different stimulus dimensions) depend on the characteristics of the second layer. The assumption of random connectivity allowed us to consider a benchmark model, an approach employed also in other studies analyzing the transmission of information between different layers of neural populations [6, 5, 52]. Given the complex selectivity profiles which are generated, the activity of a single representation neuron conveys an ambiguous information about the stimulus. Rather, the information is distributed across the activity of all neurons, such that the neural population, as a whole, is the relevant unit of computation [57].

Such 'distributed' codes have been shown to possess interesting coding properties, even in the absence of an evident structure. As a closely related example, [1] analyzed the capacity of face coding in monkey cortical neurons (superior temporal sulcus). Although the single neuron selectivity profiles as a function of different faces-stimuli were quite heterogeneous, the number of distinguishable stimuli was shown to grow exponentially with the population size. Our work shows an example of random distributed code for continuous stimuli, which possesses similar scaling properties. A relevant difference is that, in our case, it is not sufficient to distinguish between different stimuli. As the stimuli have an order, not all errors are equivalent; the objective is to reduce the mean error, and this is achieved by constraining the smoothness of the responses.

Statistic of population responses and geometry. In the last decade, recent experimental methods allowed the recording of large scale neural populations [21, 57]. In order to extract and interpret the properties of the neural codes implied by such large populations, going beyond single neuron tuning curves, many studies employed a geometrical approach [32, 33, 64, 47]. The set of joint neural responses as a function of continuous stimuli, or as a function of time, is often interpreted as a 'manifold' in the high dimensional space of neural activities, implicitly assuming a local homeomorphism to a Euclidean space ('smoothness').

In our work, we analyzed, through a geometrical picture, the coding properties of manifolds generated by complex single neuron tuning curves (Fig. 1C). In particular, as the coordinates (tuning curves) are independent samples from a one-dimensional Gaussian process, the manifolds we considered are Gaussian manifolds [49]. Although, in case of one-dimensional stimuli, the manifold of mean activities occupies always a one-dimensional subspace, by varying σ we vary its 'complexity'. Following [36], we can give a measure of this complexity, or *intrinsic dimensionality*, mathematically defined as a function of the eigenvalues μ_i of the neural responses covariance matrix, $d_i \equiv (\sum_{i=1}^{N} \mu_i)^2 / \sum_{i=1}^{N} \mu_i^2$. Such a definition is based on linear techniques of dimensionality reduction (PCA) and, roughly speaking, measures how the responses are 'spread' in the N dimensional space. When applied to our case, this definition yields a value close to N in the case of $\sigma \to 0$, where responses spread equally in all directions in the space of neural activity, and a value close to 1 in case of broad tuning curves, where responses are arranged along few axes. By varying σ, we effectively explore different intrinsic manifold dimensionality, and we quantify their coding properties.

[64] also related the geometrical properties of the manifold of neural activities to the coding properties. By analyzing a large population of visual neurons responding to a large, but discrete, set of images, the authors found that the eigenvalues of the correlation matrix of the noise-free neural responses scale as a power law. It was argued that such a scaling leads to a dimensionality which is as high as possible, such to represent stimuli with different activity patterns, still allowing the differentiability of the manifold as $N \to \infty$ (smoothness). In our case, the spectrum of the eigenvalues is not a power law, rather is flat until a crossover value, which is higher as σ decreases, and then falls rapidly (Fig. 9C). We encounter a problem of infinite derivative when $\sigma \to 0$, with arbitrary close stimuli represented by uncorrelated responses (Fig. 2A). Nevertheless, we show that this limit is not always beneficial for coding, as noise may cause large errors. More generally, our result shows how, at least in our benchmark model, the optimal arrangement of the manifold of neural responses in the activity space depends jointly on the population size and noise variance.

Compressed sensing. We studied a network where the information encoded in a high-dimensional activity pattern is compressed into the activity of a small number of neurons, a setting which exhibits analogies with the one of Compressed Sensing [35]. Compressed Sensing (CS) is a signal-processing technique for acquiring and reconstructing high (L)-dimensional signals, which are K-sparse in some basis (meaning that they can be expressed as vectors with only K elements are different from 0) from a small (N) number of linear and noisy measurements [27]. In our problem, the low dimensionality of the stimulus, x, implies the sparsity of the L dimensional activity of the sensory layer (but see [8] for more details). An important result in the field of CS is that the measurement matrix need not to be carefully designed, as random matrices achieve near-optimal results. Indeed, in this case the number of necessary measurements scales only logarithmically with the dimensionality of the signal, $N > \mathcal{O}(L \log{(L/K)})$ [19, 7]. We obtain an analog scaling by inverting Eq. (5): the number of random projections, N, such that is possible to decode L different stimuli with an arbitrary small error probability, scales only logarithmically with the number of stimuli. We notice that our task slightly differs from the typical one of CS, which is to reconstruct the L-dimensional vector. In our case we are not interested in reconstructing the activity pattern of the first layer, but rather to estimate the stimulus which evoked it.

Encoding and decoding. In order to focus on the *encoding* properties of the neural population, we assumed the existence of an ideal decoder which extracts all the available information from the noisy responses. In principle this is not a limitation, as we show in Methods that such a decode can be implemented in a simple neural network. A first layer computes a discrete approximation of the posterior distribution over stimuli, and the second one returns the posterior distribution averaged over different stimuli, computing the correct Minimum-MSE estimator. The first layer activity is obtained by passing the noisy responses through a linear-non linear filter, and then by normalizing the activity of the population such to return a correct probability distribution. All the implied operations, linear filtering, non linearity and normalization, have been assumed as canonical computations in neural circuits [24, 48, 20]. Nevertheless, the parameters of this network decoder (i.e., the synaptic weights) depend on the knowledge of the mean neural responses (tuning curves) and noise variance. An important question which concerns the *decoding* aspect of the problem, is if these ideal parameters can be learnt with biologically plausible rules, and, if not, how such non-ideal decoders compare with the ideal one. Closely related to these questions, [13] analyzed how the generalization error, in a deep neural network trained with gradient descent, depends on the number of training samples and on the spectrum of the target function. A key result, also explored in the context of neural codes [14], is that to learn high frequency components of the target function requires a higher number of examples with respect to low frequency components. In our regression task, this would imply a slower learning and a higher generalization error in case of narrow σ. Another question is how the presence of noise in the training samples, in particular in case of global errors, would impact the learnability of the decoder. Overall, the impact of limitations in the decoding architecture on the optimal encoding parameters is a relevant issue, which we leave for future research.

4 Methods

Throughout the discussion, bold letters denote vectors $\mathbf{r} = \{r_1, r_2, ..., r_N\}$. $\|\mathbf{r}\|_2^2 = \sum_i r_i^2$ represents the L_2 norm. Capital bold letters $\mathbf{W}$ denote matrices. Numerical simulations and data analysis were done using a custom code written in Julia [12].

Model description: one-dimensional stimulus

Network definition and constraints. The first, sensory layer is made by L neurons, encoding a continuous scalar stimulus $x \in [0,1]$, with Gaussian tuning curves. The firing rate of neuron j as a function of x is given by

$$u_j(x) = A \exp\left(-\frac{(x-c_j)^2}{2\sigma^2}\right), \tag{1 restated}$$

where c_j is the preferred stimulus of neuron j, σ is the tuning width, and A is a gain which will be chosen accordingly. The preferred stimuli are evenly spaced, $c_j = j/L$. These neurons are all-to-all connected to the N representation neurons, with i.i.d Gaussian weights, $W_{ij} \sim \mathcal{N}(0, 1/L)$. The response of representation neuron i is therefore obtained as

$$v_i(x) = \sum_{j=1}^{L} W_{ij} u_j(x). \tag{2 restated}$$

The gain A is chosen such as to maintain the variance of responses, averaged over different realizations of the synaptic weights, equal to a constant. This constraint reads

$$\begin{aligned}
R &= \left\langle \int_0^1 dx \left[v_i(x) - \int_0^1 dx' v_i(x') \right]^2 \right\rangle_W \\
&= \left\langle \int_0^1 dx v_i(x)^2 - \left(\int_0^1 dx v_i(x) \right)^2 \right\rangle_W \\
&= \left\langle \sum_{j,j'} W_{ij} W_{ij'} \left(\int_0^1 dx u_j(x) u_{j'}(x) \right) - \sum_{j,j'} W_{ij} W_{ij'} \left(\int_0^1 dx u_j(x) \right) \left(\int_0^1 dx u_{j'}(x) \right) \right\rangle_W \\
&= \int_0^1 dx u_j(x)^2 - \left(\int_0^1 dx u_j(x) \right)^2,
\end{aligned} \tag{12}$$

where $\langle \cdot \rangle_W$ indicates the average over the distribution of synaptic weights, and in the last line we used the fact that they are i.i.d. Gaussian, $\langle W_{ij} W_{ij'} \rangle_W = \frac{1}{L} \delta_{jj'}$. Here and in following calculations, we use the approximation for the Gaussian integral

$$\int_0^1 dx u_j(x) \approx \int_{-\infty}^{\infty} u_j(x) = A\sqrt{2\pi\sigma^2}, \tag{13}$$

which is strictly valid when c_j is far from stimulus boundaries and σ is small with respect to the stimulus space. As we consider a large number of neurons in the first layer and relatively small σ (up to $1/10$ of the stimulus space), the effects of this approximation in our results are negligible. By inserting Eq. (13) and a similar approximation for $\int_0^1 dx u_j(x)^2$ into Eq. (12), and solving for A, we obtain the final expression for the gain

$$A^2 = \frac{R}{\sqrt{\pi\sigma^2} - 2\pi\sigma^2}. \tag{14}$$

Gaussian Processes analogy. The response of each neuron of the second layer to a fixed stimulus x is a sum of Gaussian random variables. As a result, it is also a Gaussian random variable, with mean

$$\langle v_i(x)\rangle_W = \sum_{j=1}^{L} \langle W_{ij}\rangle_W u_j(x) = 0. \tag{15}$$

The covariance between the response of the same neuron to two different stimuli, x and x', is given by

$$\langle v_i(x) v_i(x')\rangle_W = \sum_{j,j'} \langle W_{ij} W_{ij'}\rangle_W \, u_j(x) u_{j'}(x') = \sum_{j=1}^{L} \frac{1}{L} u_j(x) u_j(x'). \tag{16}$$

We can approximate the discrete sum with the integral $\sum_{j=1}^{L} f(c_j)\Delta c_j \approx \int_0^1 f(c_j) dc_j$, as the error of this approximation will be of order $1/L$. We obtain therefore

$$\begin{aligned} \langle v_i(x) v_i(x')\rangle_W &\approx \int_0^1 dc_j u_j(x) u_j(x') \\ &= A^2 \int_0^1 dc_j \exp\left(-\frac{\left((x-c_j)^2 + (x'-c_j)^2\right)}{2\sigma^2}\right) \\ &\approx A^2 \sqrt{\pi\sigma^2} \exp\left(-\frac{\Delta x^2}{4\sigma^2}\right), \end{aligned} \tag{17}$$

where $\Delta x = x - x'$. In the last line we took the limit of integration going to infinity similarly to Eq. (13), an approximation which is valid when x and $x'$are far from the stimulus boundaries, in order to obtain a translational invariant expression. Equation (15) and Equation (17) show that each neuron tuning curve is a sample from a one-dimensional Gaussian process with 0 mean and squared exponential kernel with correlation length equal to $\sqrt{2}\sigma$ [56].

Encoding - decoding

Noise Model. Representation neurons are affected by additive isotropic Gaussian noise. At each trial, the vector of responses to a given stimulus x is obtained as

$$\mathbf{r} = \mathbf{v}(x) + \mathbf{z}, \tag{18}$$

where $\mathbf{z}$ is a noise vector of independent Gaussian entries with a fixed variance, $z_i \sim \mathcal{N}(0, \eta^2)$. Here, $\mathbf{v}(x) = \{v_1(x), v_2(x), ..., v_N(x)\}$ is the vector containing the mean responses of second layer neurons to the same stimulus x, Eq. (2). As a result, we can write the likelihood of a response given a stimulus as

$$p(\mathbf{r}|x) = \frac{1}{(2\pi\eta^2)^{N/2}} \exp\left(-\frac{\|\mathbf{r} - \mathbf{v}(x)\|_2^2}{2\eta^2}\right). \tag{19}$$

This equation can be written in a more general form, in case we want to model noise correlations or different noise variances for each neuron. Denoting as Σ the noise covariance matrix, the likelihood is given by

$$p(\mathbf{r}|x) = \frac{1}{(2\pi)^{N/2} (\det(\Sigma))^{1/2}} \exp\left(-(\mathbf{r} - \mathbf{v}(x))^T \Sigma^{-1} (\mathbf{r} - \mathbf{v}(x))\right). \tag{20}$$

Loss function and decoder. We employed the Mean Squared Error (MSE) in stimulus estimate to measure the coding properties of the neural population [23]. For a generic decoder, or estimator, $\hat{x} = f_{dec}(\mathbf{r})$, the MSE is defined as

$$E^2 = \int dx \int d\mathbf{r} p(\mathbf{r}|x) \left(\hat{x} - x\right)^2 . \tag{21}$$

We considered this quantity averaged over network realizations, $\varepsilon^2 \equiv \langle E^2 \rangle_W$; we often showed the square root of this quantity, $\varepsilon \equiv \sqrt{\langle E^2 \rangle_W}$, as it has the same unit of measurement of the stimulus. The extension of this measure to multi-dimensional stimuli is done by averaging the squared norm, $\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2$.

The estimator which minimizes the MSE (Minimum-MSE or MMSE) is given by the average of the posterior distribution. As we assume an uniform prior over stimuli $p(x) \sim \mathcal{U}(0,1)$, we can write the estimator as a function of the likelihood, as

$$\hat{x}_{MMSE} = \int_0^1 dx p(x|\mathbf{r}) x = \frac{\displaystyle\int_0^1 dx p(\mathbf{r}|x) x}{\displaystyle\int_0^1 dx p(\mathbf{r}|x)} . \tag{22}$$

By approximating the integrals with a discrete sum over M values, and by inserting Eq. (19), we obtain

$$\begin{aligned}\hat{x}_{MMSE} \approx \frac{\sum_{m=1}^M x_m p\left(\mathbf{r}|x_m\right)}{\sum_{m=1}^M p\left(\mathbf{r}|x_m\right)} &= \frac{\sum_{m=1}^M x_m \exp\left(-\frac{1}{2\eta^2}\sum_{i=1}^N r_i^2 + v_i^2(x_m) - 2v_i(x_m)r_i\right)}{\sum_{m=1}^M \exp\left(-\frac{1}{2\eta^2}\sum_{i=1}^N r_i^2 + v_i^2(x_m) - 2v_i(x_m)r_i\right)} \\ &= \frac{\sum_{m=1}^M x_m \exp\left(\frac{1}{2\eta^2}\sum_{i=1}^N 2v_i(x_m)r_i - v_i^2(x_m)\right)}{\sum_{m=1}^M \exp\left(\frac{1}{2\eta^2}\sum_{i=1}^N 2v_i(x_m)r_i - v_i^2(x_m)\right)},\end{aligned} \tag{23}$$

where in the last passage the term $\sum_i r_i^2$ cancel, as it is common to both numerator and denominator. This function can be implemented in a two layer network. The normalized activity of the first layer, populated by M neurons, computes a discretized approximation of the likelihood function, as

$$p(\mathbf{r}|x_m) = \tilde{h}_m = \frac{h_m}{\sum_{m=1}^M h_m} . \tag{24}$$

The (unormalized) activity of neuron m is expressed as a linear combination of the activity of the representation neurons (plus a bias term) passed through an exponential non-linearity, $h_m = \exp\left(\sum_{i=1}^M \lambda_{mi} r_i + b_m\right)$. The synaptic weights between the m-th decoder neuron and the i-th representation neuron, are a function of the true mean response of neuron i to stimulus x_m and the variance of the noise, $\lambda_{mi} = v_i(x_m)/\eta^2$. Similarly, the bias term is obtained as $b_m = -\sum_i v_i(x_m)^2/2\eta^2$. The MMSE estimator is obtained by weighting the normalized activity of the M neurons according to their 'preferred stimulus', x_m, as

$$\hat{x}_{MMSE} = \sum_{m=1}^M x_m \tilde{h}_m . \tag{25}$$

In the following discussion, we will often use the Maximum a Posterior(MAP) estimator, defined as

$$\hat{x}_{MAP} = \arg\max_{x_m} h_m = \arg\min_{x_m} \|\mathbf{r} - \mathbf{v}(x_m)\|_2^2 , \tag{26}$$

as it has a simpler geometrical interpretation: it finds the stimulus which corresponds to the closest vector of mean responses to the noisy output. In numerical simulations, the MSE for these two estimators are very similar.

The same decoder can be extended to deal with the case of non-diagonal noise covariance matrix Σ, plugging Eq. (20) into Eq. (22) . The decoding weights and biases are now correlated, $\lambda_m = \mathbf{v}^T(x_m)\Sigma^{-1}$ and $b_m = \mathbf{v}^T(x_m)\Sigma^{-1}\mathbf{v}(x_m)$, where λ_m denotes the m-th row of λ.

In numerical simulations, we computed the MSE with standard Monte Carlo method. We generated the noisy responses to sampled stimuli and we decoded them using the ideal decoder, updating the estimated MSE until convergence. We set the number of decoder neurons equal to the number of sensory neurons, $M = L$, with uniformly space preferred stimuli, $x_m = m/M$.

Errors computation

Narrow tuning curves. If $\sigma \to 0$, the first layer neurons respond only to their preferred stimulus. For this limit case, we consider that the stimulus can assume only L discrete values, $x_j = j/L$. The responses of the second layer neurons are given by $v_i(x_j) = \tilde{A}W_{ij}$, with $\tilde{A}^2 = LR$ such to have $v_i(x_j) \sim \mathcal{N}(0, R)$.

Let's denote with $p_e(\mathbf{r}|x_j) = p(\mathbf{r}|x_j)\Theta\left(|\hat{x} - x_j|\right)$ the probability density function that the noise will produce an error in decoding the response associated to stimulus x_j. With a small abuse of notation, we define the Heaviside function as $\Theta(z) = 1$ if $z > 0$, and 0 otherwise. When we take the average over synaptic weights, the probability of having an error on a stimulus x_j is independent from the decoded stimulus $\hat{x}$. The average MSE, Eq. (21), can be therefore approximated as

$$\begin{aligned}\langle E^2 \rangle_W &= \frac{1}{L}\sum_{j=1}^{L}\left\langle \int d\mathbf{r} p_e(\mathbf{r}|x_j)\left(\hat{x} - x_j\right)^2 \right\rangle_W \\ &\approx \langle P(E)\rangle_W \left\langle \frac{1}{L}\sum_{j=1}^{L}(\hat{x} - x_j)^2 \right\rangle_W \quad , \end{aligned} \tag{27}$$

where $\langle P(E)\rangle_W = \left\langle \int d\mathbf{r} p_e(\mathbf{r}|x_j)\right\rangle_W$ is the average probability that, given a stimulus, the noise will cause an error in its estimate. Despite the notation, it does not depend on the specific value of x_j. This formula has an intuitive interpretation: the MSE is the mean probability of having an error on a stimulus multiplied by the average squared error. By noticing that, if there is an error, the decoder can output any of the others $L - 1$ stimuli, we obtain

$$\left\langle \frac{1}{L}\sum_{j=1}^{L}(\hat{x} - x_j)^2 \right\rangle_W = \frac{1}{L^2}\sum_{j=1}^{L}\sum_{j'=1, j'\neq j}^{L}\left(\frac{j'}{L} - \frac{j}{L}\right)^2 \approx \frac{1}{6}, \tag{28}$$

where the last approximation holds for large L. The thing to notice here is that this quantity is of order 1, the size of the stimulus space. The average probability of error is the probability that it exists one j' such that $\mathbf{r}$ is closer to $\mathbf{v}(x_{j'})$ than to $\mathbf{v}(x_j)$. We can express this probability as a function of the probability of the complementary event,

$$\langle P(E)\rangle_W = 1 - \left\langle P\left(\|\mathbf{r} - \mathbf{v}(x_{j'})\|_2^2 > \|\mathbf{r} - \mathbf{v}(x_j)\|_2^2 \quad \forall j \neq j'\right)\right\rangle_W . \tag{29}$$

Averaging over different realizations of the synaptic matrix, the probability of not having an error on $x'$are i.i.d for different j', and we can write

$$\begin{aligned}\langle P(E)\rangle_W &= 1 - \left(1 - \left\langle P\left(\|\mathbf{r} - \mathbf{v}(x_{j'})\|_2^2 < \|\mathbf{r} - \mathbf{v}(x_j)\|_2^2\right)\right\rangle_W\right)^{L-1} \\ &\approx L\left\langle P\left(\|\mathbf{r} - \mathbf{v}(x_{j'})\|_2^2 < \|\mathbf{r} - \mathbf{v}(x_j)\|_2^2\right)\right\rangle_W \\ &= L\left\langle P\left(\sum_{i=1}^{N}\left(v_i(x) - v_i(x_{j'})\right)^2 - 2\left(v_i(x) - v_i(x_{j'})\right)z_i < 0\right)\right\rangle_W \quad . \end{aligned} \tag{30}$$

In the passage from the first to the second line, we assumed that the probability of having an error on a stimulus x_j is small, and $L - 1 \approx L$ is large, such that we can approximate $(1 - z)^L \approx 1 - Lz$. In the last passage we simply inserted Eq. (18). The difference between the response of the same neuron to two different stimuli, averaged over different synaptic weights realizations, has a Gaussian distribution, $\tilde{v}_i \equiv v_i(x_j) - v_i(x_{j'}) = \tilde{A}(W_{ij} - W_{ij'}) \sim \mathcal{N}(0, 2R)$. By averaging over the noise distribution too, the probability of error reads

$$\langle P(E)\rangle_W \approx L\int \prod_{i=1}^{N} d\tilde{v}_i \prod_{i=1}^{N} dz_i p(\tilde{v}_i)p(z_i)\Theta\left(-\sum_{i=1}^{N}\tilde{v}_i^2 + 2\sum_{i=1}^{N}\tilde{v}_i z_i\right). \tag{31}$$

This is the probability that the quantity $\rho = \sum_i \tilde{v}_i^2 - 2\tilde{v}_i z_i$ is less than 0, where $\tilde{v}_i \sim \mathcal{N}(0, 2R)$ and $z_i \sim \mathcal{N}(0, \eta^2)$. We can compute this quantity by noticing that, if we fix $\zeta = \sum_i \tilde{v}_i^2$, the conditional distribution of ρ is Gaussian, $\rho|\{\tilde{v}_i^2\} \sim \mathcal{N}(\zeta, 4\zeta\eta^2)$. By using the definition of error function we can rewrite the error probability as

$$\begin{aligned}\langle P(E)\rangle_W &\approx L\int_0^\infty d\zeta p(\zeta)\int_{-\infty}^{0} d\rho p(\rho|\zeta) \\ &= \frac{L}{2}\int_0^\infty d\zeta p(\zeta)\,\mathrm{erfc}\left(\sqrt{\frac{\zeta}{8\eta^2}}\right), \end{aligned} \tag{32}$$

where $p(\zeta) = \frac{(\zeta/2R)^{N/2-1}\exp(-\zeta/4R)}{2^{N/2+1}\Gamma(N/2)}$ is the probability density function of a Chi-squared distribution. Computing this integral, we obtain

$$\begin{aligned}\langle P(E)\rangle_W &\approx L\frac{(\frac{\eta^2}{2R})^{\frac{N}{2}}\Gamma(N)}{\Gamma(\frac{N}{2})}\,{}_2\tilde{F}_1\left(\frac{N}{2},\frac{1+N}{2},\frac{2+N}{2},-2\frac{\eta^2}{R}\right)\\ &= L\frac{\left(\frac{\eta^2}{2R}\right)^{\frac{N}{2}}\Gamma\left(N\right)}{\Gamma\left(\frac{N}{2}\right)\Gamma\left(\frac{2+N}{2}\right)}\sum_{n=0}^{\infty}\frac{\left(\frac{N}{2}\right)_n\left(\frac{N+1}{2}\right)_n}{\left(\frac{N+2}{2}\right)_n n!}\left(-2\frac{\eta^2}{R}\right)^n,\end{aligned} \tag{33}$$

where ${}_2\tilde{F}_1(a,b,c,x)$ is the regularized 2F1 Hypergeometric function and in the second line we substituted its definition. The Pochammer symbol is also defined through Gamma functions, $(x)_n = \frac{\Gamma(x+n)}{\Gamma(x)}$. Simplifying and using the identity $\sum_{n=0}^{\infty}\frac{(x)_n}{n!}a^n = (1-a)^{-x}$, we obtain the expression for the error probability which appears in the main text

$$\begin{aligned}\langle P(E)\rangle_W &\approx L(\frac{\eta^2}{2R})^{\frac{N}{2}}\frac{\Gamma(N)}{\Gamma^2(\frac{N}{2})\frac{N}{2}(1+2\eta^2/R)^{\frac{N+1}{2}}}\\ &\approx \frac{L}{\sqrt{2\pi N}}\exp\left(-\log\left(1+\frac{R}{2\eta^2}\right)\frac{N}{2}\right),\end{aligned} \tag{5 restated}$$

where in the last step we used the Stirling approximation for the Gamma functions.

Broad tuning curves. As soon as $\sigma > 1/L$, we return to consider continuous stimuli. In this case the noise can also produce small scale errors. We split the error in two contributions, local and global. Since our system has a natural correlation length, Eq. (17), we define as global an error when the difference between the stimulus and its estimate is greater than σ, $|\hat{x}-x| > \sigma$. This definition is a bit tricky, as for very large σ all the errors will be local. Anyway, we are interested in the case where σ is relatively small with respect to stimulus space. We rewrite the MSE as

$$\varepsilon^2 = \langle E^2\rangle_W = \langle E_l^2 + E_g^2\rangle_W = \left\langle\int dx d\mathbf{r} p_l\left(\mathbf{r}|x\right)\left(\hat{x}-x\right)^2\right\rangle_W + \left\langle\int dx d\mathbf{r} p_g(\mathbf{r}|x)\left(\hat{x}-x\right)^2\right\rangle_W, \tag{34}$$

where $p_{l/g}(\mathbf{r}|x) = p(\mathbf{r}|x)\Theta\big(\pm(\sigma - |\hat{x}-x|)\big)$ denotes the probability density function that, given x, the noise will cause a local/global error. It holds the following normalization $\int d\mathbf{r} p_l(\mathbf{r}|x) + p_g(\mathbf{r}|x) = 1$.

Local error. By using the MAP decoder, Eq. (26), the stimulus estimate will correspond to the x' such that $\mathbf{v}(x')$ has the minimal distance from $\mathbf{r}$. If the error is local, this point corresponds to the projection of the noise vector onto the response curve. By expanding linearly the response curve around $\mathbf{v}(x)$, we obtain

$$\left\|\mathbf{r}\cdot\hat{\mathbf{v}'}(x)\right\|_2^2 \approx \|\mathbf{v}(x+\Delta x) - \mathbf{v}(x)\|_2^2 \approx \|\mathbf{v}'(x)\|_2^2\,\Delta x^2, \tag{35}$$

where $\mathbf{v}'(x) = \partial\mathbf{v}(x)/\partial x$ and $\hat{\mathbf{v}'}(x)$ is the normalized vector with the same direction. The resulting error will be $\Delta x^2 = (\hat{x}-x)^2 = \frac{\left\|\mathbf{r}\cdot\hat{\mathbf{v}'}(x)\right\|_2^2}{\|\mathbf{v}'(x)\|_2^2}$. We will show that the probability of global error will be exponentially small in N, therefore we may approximate $p_l(\mathbf{r}|x)$ with the whole Gaussian, Eq. (19). When integrating over the isotropic Gaussian noise, the magnitude of the projection onto a fixed unit vector will be simply equal to the the variance. We obtain therefore

$$\begin{aligned}\langle E_l^2\rangle_W &= \left\langle\int dx\int d\mathbf{z}\frac{\left\|\mathbf{z}\cdot\hat{\mathbf{v}'}(x)\right\|_2^2}{\|\mathbf{v}'(x)\|_2^2}\right\rangle_W\\ &= \left\langle\int dx\frac{\eta^2}{\|\mathbf{v}'(x)\|_2^2}\right\rangle_W\end{aligned} \tag{36}$$

We now approximate the average of the inverse with the inverse of the average of the derivative of the tuning curves, $\langle 1/\cdot\rangle_W \approx 1/\langle\cdot\rangle$, an approximation which is valid if L is large and σ is small with respect to stimulus space. For the average derivative of the tuning curves, we obtain

$$\begin{aligned}\left\langle \|\mathbf{v}'(x)\|_2^2 \right\rangle_W &= \left\langle \left(\sum_{i=1}^{N} W_{ij} \frac{\partial u_j(x)}{\partial x} \right)^2 \right\rangle_W = \left\langle \left(\sum_{i=1}^{N} W_{ij} \frac{(x-c_j)}{\sigma^2} u_j(x) \right)^2 \right\rangle_W \\ &= \sum_{i=1}^{N} \sum_{jj'} \langle W_{ij} W_{ij'} \rangle_W \frac{(x-c_j)(x-c_{j'})}{\sigma^4} u_j(x) u_{j'}(x) \\ &= \frac{N}{\sigma^4} \sum_{j=1}^{L} \frac{1}{L} \left(x-c_j\right)^2 u_j^2(x) \approx \frac{N}{\sigma^4} \int_0^1 dc_j \left(x-c_j\right)^2 u_j^2(x) \\ &\approx \frac{NA^2\sqrt{\pi\sigma^2}}{2\sigma^2}, \end{aligned} \tag{37}$$

where in the last line we approximated the sum with the integral and we took the limit of integration going to infinity, similarly to previous calculations. Finally, by using the approximation for small σ, $A^2 \approx R/\sqrt{\pi\sigma^2}$, we get the expression of the main text

$$\varepsilon_l^2 = \langle E_l^2 \rangle_W \approx \frac{2\sigma^2\eta^2}{RN}. \tag{38}$$

This expression is equivalent to the inverse of the average Fisher Information, which, in case of neurons affected by i.i.d Gaussian noise, is given by $J(x) = \|\mathbf{v}'(x)\|_2^2/\eta^2$.

Global error. In computing an approximation for the scaling of global errors, we extend the reasoning we have done for discrete stimuli. By assuming that the magnitude of a global error is independent from its probability, we write an expression similar to Eq. (27),

$$\left\langle E_g^2 \right\rangle_W = \langle P(E) \rangle_W \left\langle \int_0^1 dx (\hat{x}-x)^2 \right\rangle_W. \tag{39}$$

We use the approximation that, in case of global error, the decoded stimulus,averaging over different distributions of synaptic weights, is uniformly distributed in the interval $\hat{x} \notin [x-\sigma, x+\sigma]$. The average magnitude of global error is therefore

$$\bar{\varepsilon}_g = \left\langle \int dx \left(\hat{x}-x\right)^2 \right\rangle_W \approx \frac{1}{6}, \tag{40}$$

a quantity which is of order 1. The probability of an error being global, averaged over different realizations of W, does not depend on the specific value of the stimulus. Computing this probability rigorously is hard, due to the correlations between nearby responses. Nevertheless, we know that for stimuli at a distance greater than σ the two responses are uncorrelated, Eq. (17). We can therefore divide the curve into $1/\sigma$ uncorrelated discrete segments of responses, and approximate the global error as the probability of having an error between stimuli belonging to two different segments. The variance of the centers of these segments across the stimulus space will be equal to R , therefore, by substituting to L the number of segments in Eq. (5), we obtain the expression of the main text,

$$\varepsilon_g^2 = \left\langle E_g^2 \right\rangle_W \approx \frac{1}{\sigma\sqrt{2\pi N}} \bar{\varepsilon}_g \exp\left(-\log\left(1+\frac{R}{2\eta^2}\right)\frac{N}{2}) \right). \tag{41}$$

Input noise. We consider the case in which the first layer responses are affected by i.i.d Gaussian noise $\tilde{\mathbf{u}}(x) = \mathbf{u}(x) + \mathbf{z}^{\mathbf{u}}$, with $z_i^u \sim \mathcal{N}(0,\xi^2)$. This results in a multivariate Gaussian distribution for the responses of the second layer, Eq. (20), with covariance matrix $\Sigma = \eta^2\mathbf{I} + \xi^2\mathbf{W}\mathbf{W}^T$. The matrix $\mathbf{B} \equiv \mathbf{W}\mathbf{W}^T$ follows the well known Wishart distribution, with mean $\mathbf{I}$ and fluctuations of order $1/L$. We rewrite the covariance matrix as the sum of the identity plus a perturbation

$$\Sigma = \tilde{\eta}^2\mathbf{I} + \xi^2(\mathbf{B}-\mathbf{I}), \tag{42}$$

where $\tilde{\eta}^2 = \eta^2 + \xi^2$. In order to obtain an estimate of the effects of input noise on the local error, we consider the inverse of the Fisher Information (FI) as a lower bound to the MSE. For correlated populations, the FI is given by [59]

$$J(x) = \mathbf{v}'(x)^T \Sigma^{-1} \mathbf{v}'(x). \tag{43}$$

If the perturbation matrix is small, we can approximate the inverse of the correlation matrix at the second order $\Sigma^{-1} \approx \frac{1}{\tilde{\eta}^2}\mathbf{I} - \frac{\xi^2}{\tilde{\eta}^4}\left(\mathbf{B}-\mathbf{I}\right) + \frac{\xi^4}{\tilde{\eta}^6}\left(\mathbf{B}-\mathbf{I}\right)^2$, and write the FI as

$$\begin{aligned} J(x) &\approx J^{ind}(x) - \delta J(x) \\ &= \frac{\|\mathbf{v}'(x)\|_2^2}{\tilde{\eta}^2} - \frac{\xi^2}{\tilde{\eta}^4}\mathbf{u}'^T(x)\left(\mathbf{B}^2-\mathbf{B}\right)\mathbf{u}'(x) + \frac{\xi^4}{\tilde{\eta}^6}\mathbf{u}'^T(x)\left(\mathbf{B}^3-2\mathbf{B}^2+\mathbf{B}\right)\mathbf{u}'(x). \end{aligned} \tag{44}$$

We recognize in the first term, $J^{ind}(x)$, the FI in the case of i.i.d Gaussian output noise with variance $\tilde{\eta}^2$. All the correction terms to the FI are related to the moments of the matrix $\mathbf{B}$. Since all the entries are Gaussian, it is possible to compute their mean through the Isserlis' theorem. Using the identity $\langle W_{ij}W_{mn}\rangle_W = \frac{1}{L}\delta_{im}\delta_{jn}$, we obtain:

$$\langle B_{mn}\rangle_W = \left\langle \sum_{j=1}^{N} W_{jm}W_{jn} \right\rangle_W = \frac{N}{L}\delta_{mn}, \tag{45}$$

$$\langle B^2_{mn}\rangle_W = \left\langle \sum_{i=1}^{L}\sum_{j=1,j'=1}^{N} W_{jm}W_{ji}W_{j'i}W_{j'n} \right\rangle_W = \left(\frac{N}{L}+\frac{N^2}{L^2}+\frac{N}{L^2}\right)\delta_{mn}, \tag{46}$$

$$\langle B^3_{mn}\rangle_W = \left\langle \sum_{i=1,i'=1}^{L}\sum_{j=1,j'=1,j''=1}^{N} W_{jm}W_{ji}W_{j'i}W_{j'i'}W_{j''i'}W_{j''n} \right\rangle_W = \left(\frac{N^3}{L^3}+3\frac{N^2}{L^3}+3\frac{N^2}{L^2}+4\frac{N}{L^3}+3\frac{N}{L^2}+\frac{N}{L}\right)\delta_{mn}. \tag{47}$$

Expressing the results only with in the higher powers of N/L, the mean of the perturbation term is

$$\begin{aligned} \langle \delta J(x)\rangle_W &\approx \frac{N^2\xi^2}{L^2\tilde{\eta}^4}\mathbf{u}'(x)^T\mathbf{I}\mathbf{u}'(x) - \frac{N^2\xi^4}{L^2\tilde{\eta}^6}\mathbf{u}'(x)^T\mathbf{I}\mathbf{u}'(x) \\ &= \frac{N^2\xi^2A^2\sqrt{\pi\sigma^2}}{2L\tilde{\eta}^4\sigma^2}\left(1-\frac{\xi^2}{\tilde{\eta}^2}\right), \end{aligned} \tag{48}$$

where we computed $\mathbf{u}'(x)^T\mathbf{I}\mathbf{u}'(x) = \sum_{j=1}^{L} u_j'(x)^2$ in the same way of Eq. (37). By inserting Eq. (37) and Eq. (48) in Eq. (44), we obtain the expression for the FI in case of input noise

$$\langle J(x)\rangle_W \approx \frac{A^2N\sqrt{\pi\sigma^2}}{2\sigma^2\tilde{\eta}^2}\left(1-\frac{N\xi^2}{L\tilde{\eta}^2}+\frac{N\xi^4}{L\tilde{\eta}^4}\right), \tag{49}$$

and, by inverting it, the approximation for the MSE which appears in the main text,

$$\varepsilon_l^2 = \langle E^2\rangle_W \approx \frac{1}{\langle J(x)\rangle_W} \approx \varepsilon_{l,\text{ind}}^2\left(1+\frac{N\xi^2}{L\tilde{\eta}^2}-\frac{N\xi^4}{L\tilde{\eta}^4}\right), \tag{10 restated}$$

where $\varepsilon_{l,\text{ind}}^2 \approx 2\sigma^2\tilde{\eta}^2/RN$. Similar calculations can be done assuming a covariance matrix with the same statistic, but uncorrelated with the synaptic weights. As an example, we considered $\Sigma_{rand} = \eta^2 I + \xi^2\mathbf{X}\mathbf{X}^T$ with $X_{ij} \sim \mathcal{N}(0, \frac{1}{L})$ such that $\langle X_{ij}W_{mn}\rangle_{W,X} = 0$. In this case we have no first order corrections, and the FI is increased

$$\langle J(x)\rangle_{W,X} \approx \frac{A^2N\sqrt{\pi\sigma^2}}{2\sigma^2\tilde{\eta}^2}\left(1+\frac{N\xi^4}{L\tilde{\eta}^4}\right), \tag{50}$$

yielding a lower MSE, Eq. (11).

Extension to multidimensional stimuli

We consider stimuli in the hypercube $\mathbf{x} \in [0,1]^K$and the two extreme cases of pure and conjunctive sensory neurons.

Pure case. Each sensory neuron is sensitive to a single stimulus dimension, x_k. The L neurons are equally assigned to stimulus dimensions, such that each dimension is monitored by $M = L/K$ neurons. The activity of neuron j_k is given by

$$u^p_{j_k}(\mathbf{x}) = A_p \exp\left(-\frac{\left(x_k - c_{j_k}\right)^2}{2\sigma^2}\right), \tag{51}$$

with preferred stimuli evenly spaced, $c_{j_k} = j_k/M$ for $j_k = 1, ..., M$. The responses of second layer neurons are given by a random sum of all sensory neurons, and can be written as a superposition of one-dimensional tuning curves, independent for each dimension,

$$\begin{aligned} v_i^p(\mathbf{x}) &= \sum_{k=1}^{K} \sum_{j_k=1}^{M} W_{ij_k} u_{j_k}(\mathbf{x}) \\ &= \sum_{k=1}^{K} v_{i,k}^p(x_k). \end{aligned} \tag{52}$$

Imposing the resource constraint Eq. (12), with similar calculations, we obtain $A_p^2 = R/\left(\left(\pi\sigma^2\right)^{1/2} - 2\pi\sigma^2\right)$.

The local error along each dimension is computed, similarly to Eq. (35), expanding linearly the surface, and obtaining

$$\Delta x_k^2 \approx \frac{\left\|\mathbf{r} \cdot \hat{\mathbf{v}'}_k(x)\right\|_2^2}{\|\mathbf{v}'_k(x)\|_2^2}, \tag{53}$$

where $\mathbf{v}'_k = \partial \mathbf{v}(x)/\partial x_k$. The calculation is analog to the one-dimensional case, but the derivative along each dimension acts only on $1/K$ terms. As a consequence, the local error along each dimension is

$$\varepsilon_{l,p,k}^2 = \frac{2K\sigma^2\eta^2}{NA_p^2\sqrt{\pi\sigma^2}} \approx \frac{2K\sigma^2\eta^2}{RN}, \tag{54}$$

and the total one $\varepsilon_{l,p}^2 = K\varepsilon_{l,p,k}^2$.

As for global errors, since the multi-dimensional tuning curves are superposition of one-dimensional ones, we can obtain a global error on each dimension independently. By assuming that the probability of having a global error on more than one dimension is negligible, we can approximate the total probability of having a global error as the sum of probabilities along each dimension, $P(E_g) = \sum_{k=1}^{K} P(E_{g,k})$. In computing the probability along each dimension, we have to keep into account that, as the total variance across all stimulus space is equal to R, the variance across each dimension is reduced by a factor of K. By inserting R/K instead of R in Eq. (5) and summing over the dimensions, we obtain that the global error for the pure case scales approximately as

$$\varepsilon_{g,p}^2 \approx \frac{K\bar{\varepsilon}_g}{\sigma\sqrt{2\pi N}} \exp\left(-\log\left(1 + \frac{R}{2K\eta^2}\right)\frac{N}{2}\right), \tag{55}$$

where the average magnitude of global error, $\bar{\varepsilon}_g$, is again a term of order 1.

Conjunctive case. In this case the response of sensory neurons are multi-dimensional Gaussian,

$$u_j^c(\mathbf{x}) = A_c \exp\left(-\frac{\|\mathbf{x} - \mathbf{c}_j\|_2^2}{2\sigma^2}\right), \tag{56}$$

with preferred stimuli arranged on a K-dimensional square grid of side $1/M$,with $M = L^{1/K}$. The response of the second layer neurons are given by

$$v_i^c(\mathbf{x}) = \sum_j W_{ij} u_j(\mathbf{x}). \tag{57}$$

In this case, the tuning curves are multi-dimensional Gaussian processes with covariance function $\langle v(\mathbf{x})v(\mathbf{x}+\Delta\mathbf{x})\rangle_W = A_c^2 \exp\left(-\|\Delta\mathbf{x}\|_2^2/4\sigma^2\right)$. By imposing the resource constraint, we obtain $A_c^2 = R/\left(\left(\pi\sigma^2\right)^{K/2} - (2\pi\sigma^2)^K\right)$. We notice that, as K becomes large, the edge effects in the approximations such Eq. (13) become more relevant, and the denominator may become negative. For high number of dimensions, we may need broader widths to cover the stimulus space, and the difference between Gaussian integrals $\left(\int d\mathbf{x}u_j(\mathbf{x})\right)^2$ and $\int d\mathbf{x}u_j^2(\mathbf{x})$ change sign for large values of σ. We didn't encounter this problem for $K = 3$ (results of the main text) and in the range of values for σ we explored.

When we compute the local error, Eq. (53), the derivative acts on all the terms of the sum, as all sensory neurons are sensitive to stimulus variations. As a result, we obtain, for the local error along each dimension in the conjunctive case,

$$\varepsilon_{l,c,k}^2 = \frac{2\sigma^2\eta^2}{NA_c^2(\pi\sigma)^{K/2}} \approx \frac{2\sigma^2\eta^2}{RN}, \tag{58}$$

Simarly, the total one will be $\varepsilon^2_{l,c} = K\varepsilon^2_{l,c,k}$. The ratio between local errors in the two cases is therefore $\varepsilon^2_{l,c}/\varepsilon^2_{l,p} = 1/K$.

Global errors can happen in all stimulus directions. Here, we extend the reasoning of the one-dimensional case, by observing that stimuli are correlated in a radius of σ. We can therefore divide the K-dimensional surface of responses into $1/\sigma^K$ regions. The variance of the centers of these regions across the stimulus space is equal to R. Therefore, by substituting the number of uncorrelated regions to L in Eq. (5), we obtain an approximate scaling of the global error as

$$\varepsilon^2_{g,c} \approx \frac{1}{\sigma^K\sqrt{2\pi N}}\bar{\varepsilon}_g \exp\left(-\log\left(1+\frac{R}{2\eta^2}\right)\frac{N}{2}\right). \tag{59}$$

Data analysis and model fitting

Data description and summary statistics The detailed data description is reported in [50], and data are publicly available at https://osf.io/u57df/. They consist of the responses (firing rates) of $N \sim 500$ neurons, recorded during an arm posture 'hold' task at 27 different positions (and with 2 hand orientation, up and down) arranged on a virtual cube of size 40x40x40 cm. The response of each neuron for each position is recorded for several trials ($\sim$ 10 trials per position). Tuning curves are computed by averaging over trials. In order to measure the level of irregularity of a single tuning curve in a non parametric form, the authors introduced a complexity measure. For each neuron, it is defined as the standard deviation of the discrete derivative between the response at one target position and its response at the closest target,

$$c(D_{min})_i = std\left(\frac{\|v(x)-v(x+\Delta x)\|}{\sqrt{\|\Delta x\|^2}} s.t. \|\Delta x\|^2_2 < D_{min}\right), \tag{60}$$

where $v(\mathbf{x})$ is the mean response at stimulus $\mathbf{x}$. In the data, the D_{min} is imposed by the experiment and is equal to 35. This limitation, inherent to the data themselves, prevent us from capturing high frequency components due to aliasing phenomena. The author measured also another summary statistic, the distribution of R^2 values resulting from the fit of the tuning curves with a linear model, Eq. (9),

$$R^2_i = 1 - \frac{RSS}{TSS} = 1 - \frac{\sum_{\mathbf{x}}(v_l(\mathbf{x})-v(\mathbf{x}))^2}{\sum_{\mathbf{x}} v(\mathbf{x})^2}, \tag{61}$$

where $v_l(\mathbf{x})$ is the response predicted by a fitted linear model, and the sum is over the stimuli used in the experiment. The distribution of these quantities across different neurons is a measure of the irregularity of the neural population; if the population would have been perfectly described by Eq. (9), the R^2 distribution would have been a delta function peaked at 1, while the complexity measure would have been biased towards low values.

Model fitting. We considered the tuning curves as a function of the position only, ignoring the difference in hand orientation. We analyzed neurons responding with at least 5 spikes/s at more than two positions. We shifted and normalized the data such that tuning curves have zero mean and unit variance across different stimuli. We generated an irregular population with our model, featured by a sensory layer of conjunctive neurons responding to a three-dimensional stimulus. We used $L = 100^3$ neurons, tiling a 200 by 200 by 200 cube, such that the stimulus space is covered without boundary effects, with preferred stimuli arranged on a square grid of side 2. For computational reasons, $\mathbf{W}$ is now a sparse random matrix (sparsity equal to 0.1) with Gaussian entries. The tuning curves in the second layer were normalized one by one to have variance equal to 1. With respect to the model of [50], there are two main differences: in their case the random weights were distributed according to a uniform distribution, and the random sum was passed through a threshold-linear function. With this formulation, the model had two tunable parameters: the tuning width of first layer neurons, σ, and the the threshold of the non linear function of the second layer. The only tunable parameter of our model is σ.

In order to fit the model, we generated the neural responses of a number of representation neurons equal to the number of recorded neurons at the same stimuli (27) of the data. We then computed the distribution of the complexity measure for different values of σ and we chose σ_f such as to minimize the Kolmogorov-Smirnov (KS) distance between the distribution of the model and the one of the data (Fig. 8A). At this σ_f, the two distributions are very similar, even if real data show a broader distribution of values in both directions; for comparison, the distribution implied by a linear model is biased towards lower values (Fig. 8B). For the sake of completeness, we computed the KS distance between the model and the data also for the R^2 measure (Fig. 8A, red line). This quantity simply decreases with σ, and the model at σ_f underestimate the linear component of the tuning curves (Fig. 8C). Nevertheless, this is expected, since our model has no non linearity, which potentially may increase the linear component of the tuning curves. It is worth noticing that in the original work, a model with two parameters still underestimates the distribution of R^2 values and only the complexity measure was considered in the fitting procedure. The authors obtained a good agreement only considering a model with more parameters (namely, different threshold for each neuron and different widths in the sensory layer).

We also did simulations with a heterogeneous noise variance across the population extracted from the data. We assigned to each recorded neuron a signal-to-noise ratio in the following way. We estimated the variance of the signal as the variance of the mean responses across different stimuli, $\hat{R}_i = \langle (v_i(x) - \langle v_i(x) \rangle_x)^2 \rangle_x$. Then, we averaged the trial to trial variability, across different stimuli, $\hat{\eta}_i^2 = \left\langle \langle r_i^t - v_i(x) \rangle_t \right\rangle_x$, where r^t is the response at each trial. As in our model we kept the variance of the signal equal to 1, the noise variance in the i-th neuron in simulations was set equal to

$$\eta_i^2 = \frac{\hat{\eta}_i^2}{\hat{R}_i}. \tag{62}$$

In principle, the noise may be dependent from the mean. To control for this effect, we also preprocessed the data with a variance stabilizing transformation (substituting $r(\mathbf{x})$ with $\sqrt{r(\mathbf{x})}$, [63]). The distribution of the noise variance across neurons obtained in this way does not vary substantially.

For numerical simulations in Fig. 6, the tuning curves are computed at a finer scale than the data (cubic grid of 21 by 21 by 21 points). As expected, the tuning curves show a broad range of behavior with respect to the linear fit, that goes from very linear to very irregular (Fig. 8 D-F). The linear population for the comparison is constructed by sampling the preferred vectors ,$\mathbf{P}_i = (b_1, c_2, d_3)$, uniformly on the unit sphere and using Eq. (9) to generate the responses. Similarly to the irregular ones, these tuning curves are shifted and normalized to have zero mean and unit variance.

5 Acknowledgements

References

[1] Abbott, L. F., Rolls, E. T., & Tovee, M. J. (1996). Representational capacity of face coding in monkeys. Cerebral Cortex, 6, 498–505.

[2] Andersen, R. A., Essick, G. K., & Siegel, R. M. (1985). Encoding of spatial location by posterior parietal neurons. Science, 230, 456–458.

[3] Arakaki, T., Barello, G., & Ahmadian, Y. (2019). Inferring neural circuit structure from datasets of heterogeneous tuning curves. PLoS Computational Biology, 15, e1006816–.

[4] Atick, J. J. & Redlich, A. N. (1990). Towards a Theory of Early Visual Processing. Neural Computation, 2, 308–320.

[5] Babadi, B. & Sompolinsky, H. (2014). Sparseness and Expansion in Sensory Representations. Neuron, 83, 1213–1226.

[6] Barak, O., Rigotti, M., & Fusi, S. (2013). The sparseness of mixed selectivity neurons controls the generalization-discrimination trade-off. Journal of Neuroscience, 33, 3844–3856.

[7] Baraniuk, R., Davenport, M., DeVore, R., & Wakin, M. (2008). A simple proof of the restricted isometry property for random matrices. Constructive Approximation, 28, 253–263.

[8] Baraniuk, R. G. & Wakin, M. B. (2009). Random projections of smooth manifolds. Foundations of Computational Mathematics, 9, 51–77.

[9] Barlow, H. B. (1961). Possible Principles Underlying the Transformations of Sensory Messages. Sensory Communication, pp. 216–234.

[10] Berens, P., Ecker, A. S., Gerwinn, S., Tolias, A. S., & Bethge, M. (2011). Reassessing optimal neural population codes with neurometric functions. Proceedings of the National Academy of Sciences of the United States of America, 108, 4423–4428.

[11] Bethge, M., Rotermund, D., & Pawelzik, K. (2002). Optimal short-term population coding: When Fisher information fails. Neural Computation, 14, 2317–2351.

[12] Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. SIAM Review.

[13] Bordelon, B., Canatar, A., & Pehlevan, C. (2020). Spectrum Dependent Learning Curves in Kernel Regression and Wide Neural Networks. In International Conference of Machine Learning (ICML).

[14] Bordelon, B., Paulson, J. A., & Pehlevan, C. (2021). Population Codes Enable Learning from Few Examples By Shaping Inductive Bias. bioRxiv.

[15] Bremmer, F., Ilg, U. J., Thiele, A., Distler, C., & Hoffmann, K. P. (1997). Eye position effects in monkey cortex. I. Visual and pursuit-related activity in extrastriate areas MT and MST. Journal of Neurophysiology, 77.

[16] Brunel, N. & Nadal, J. P. (1998). Mutual Information, Fisher Information, and Population Coding. Neural Computation, 10, 1731–1757.

[17] Burak, Y. (2014). Spatial coding and attractor dynamics of grid cells in the entorhinal cortex. Current Opinion in Neurobiology, 25, 169–175.

[18] Butts, D. A. & Goldman, M. S. (2006). Tuning curves, neuronal variability, and sensory coding. PLoS Biology, 4, 639–646.

[19] Candes, E. J. & Tao, T. (2006). Near-optimal signal recovery from random projections: Universal encoding strategies? IEEE Transactions on Information Theory.

[20] Carandini, M. & Heeger, D. J. (2012). Normalization as a canonical neural computation. Nature Reviews Neuroscience.

[21] Cunningham, J. P. & Yu, B. M. (2014). Dimensionality reduction for large-scale neural recordings. Nature Neuroscience, 17, 1500–1509.

[22] da Silveira, R. A. & Rieke, F. (2021). The Geometry of Information Coding in Correlated Neural Populations. Annu. Rev. Neurosci., pp. 1–30.

[23] Dayan, P. & Abbott, L. F. (2001). Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems. (MIT Press).

[24] Deneve, S., Latham, P. E., & Pouget, A. (1999). Reading population codes: A neural implementation of ideal observers. Nature Neuroscience, 2, 740–745.

[25] Doeller, C. F., Barry, C., & Burgess, N. (2010). Evidence for grid cells in a human memory network. Nature, 463.

[26] Doi, E., Gauthier, J. L., Field, G. D., Shlens, J., Sher, A., Greschner, M., Machado, T. A., Jepson, L. H., Mathieson, K., Gunning, D. E., Litke, A. M., Paninski, L., Chichilnisky, E. J., & Simoncelli, E. P. (2012). Efficient coding of spatial information in the primate retina. Journal of Neuroscience, 32, 16256–16264.

[27] Donoho, D. L. (2006). Compressed sensing. IEEE Transactions on Information Theory.

[28] Eliav, T., Maimon, S. R., Aljadeff, J., Tsodyks, M., Ginosaur, G., Las, L., & Ulanovsky, N. (2020). Multiscale representation of very large environments in the hippocampus of flying bats.

[29] Fiete, I. R., Burak, Y., & Brookings, T. (2008). What grid cells convey about rat location. Journal of Neuroscience, 28, 6858–6871.

[30] Finkelstein, A., Ulanovsky, N., Tsodyks, M., & Aljadeff, J. (2018). Optimal dynamic coding by mixed-dimensionality neurons in the head-direction system of bats. Nature Communications, 9.

[31] Fiscella, M., Franke, F., Farrow, K., Müller, J., Roska, B., da Silveira, R. A., & Hierlemann, A. (2015). Visual coding with a population of direction-selective neurons. Journal of Neurophysiology, 114, 2485–2499.

[32] Fusi, S., Miller, E. K., & Rigotti, M. (2016). Why neurons mix: High dimensionality for higher cognition. Current Opinion in Neurobiology, 37, 66–74.

[33] Gallego, J. A., Perich, M. G., Miller, L. E., & Solla, S. A. (2017). Neural Manifolds for the Control of Movement. Neuron, 94, 978–984.

[34] Ganguli, D. & Simoncelli, E. P. (2014). Efficient sensory encoding and Bayesian inference with heterogeneous neural populations. Neural Computation.

[35] Ganguli, S. & Sompolinsky, H. (2012). Compressed sensing, sparsity, and dimensionality in neuronal information processing and data analysis.

[36] Gao, P., Trautmann, E., Yu, B., Santhanam, G., Ryu, S., Shenoy, K., & Ganguli, S. (2017). A theory of multineuronal dimensionality, dynamics and measurement. p. 214262.

[37] Gaucher, Q., Panniello, M., Ivanov, A. Z., Dahmen, J. C., King, A. J., & Walker, K. M. (2020). Complexity of frequency receptive fields predicts tonotopic variability across species. eLife, 9.

[38] Georgopoulos, A. P., Kalaska, J. F., Caminiti, R., & Massey, J. T. (1982). On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. Journal of Neuroscience, 2, 1527–1537.

[39] Hafting, T., Fyhn, M., Molden, S., Moser, M. B., & Moser, E. I. (2005). Microstructure of a spatial map in the entorhinal cortex. Nature.

[40] Harel, Y. & Meir, R. (2020). Optimal multivariate tuning with neuron-level and population-level energy constraints.

[41] Hubel, D. H. & Wiesel, T. N. (1959). Receptive fields of single neurones in the cat's striate cortex. The Journal of Physiology.

[42] Kadia, S. C. & Wang, X. (2003). Spectral integration in A1 of awake primates: Neurons with single- and multipeaked tuning characteristics. Journal of Neurophysiology, 89.

[43] Kayaert, G., Biederman, I., Op De Beeck, H. P., & Vogels, R. (2005). Tuning for shape dimensions in macaque inferior temporal cortex. European Journal of Neuroscience, 22.

[44] Kettner, R. E., Schwartz, A. B., & Georgopoulos, A. P. (1988). Primate motor cortex and free arm movements to visual targets in three-dimensional space. III. Positional gradients and population coding of movement direction from various movement origins. Journal of Neuroscience, 8.

[45] Killian, N. J., Jutras, M. J., & Buffalo, E. A. (2012). A map of visual space in the primate entorhinal cortex. Nature.

[46] Kim, J. H. J., Fiete, I., & Schwab, D. J. (2020). Superlinear Precision and Memory in Simple Population Codes. pp. 1–5.

[47] Kobak, D., Pardo-Vazquez, J. L., Valente, M., Machens, C. K., & Renart, A. (2019). State-dependent geometry of population activity in rat auditory cortex. eLife, 8, 1–27.

[48] Kouh, M. & Poggio, T. (2008). A canonical neural circuit for cortical nonlinear operations. Neural Computation.

[49] Lahiri, S., Gao, P., & Ganguli, S. (2016). Random projections of random manifolds. pp. 1–45.

[50] Lalazar, H., Abbott, L. F., & Vaadia, E. (2016). Tuning Curves for Arm Posture Control in Motor Cortex Are Consistent with Random Connectivity. PLoS Computational Biology, 12, 1–27.

[51] Lewicki, M. S. (2002). Efficient coding of natural sounds. Nature Neuroscience, 5.

[52] Litwin-Kumar, A., Harris, K. D., Axel, R., Sompolinsky, H., & Abbott, L. F. (2017). Optimal Degrees of Synaptic Connectivity. Neuron, 93, 1153–1164.

[53] Mathis, A., Herz, A. V., & Stemmler, M. B. (2012). Resolution of nested neuronal representations can be exponential in the number of neurons. Physical Review Letters, 109, 1–5.

[54] Miller, J. P., Jacobs, G. A., & Theunissen, F. E. (1991). Representation of sensory information in the cricket cercal sensory system. I. Response properties of the primary interneurons. Journal of Neurophysiology, 66.

[55] Montemurro, M. A. & Panzeri, S. (2006). Optimal tuning widths in population coding of periodic variables. Neural Computation, 18, 1555–1576.

[56] Rasmussen, C. E. (2004). Gaussian Processes in machine learning. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).

[57] Saxena, S. & Cunningham, J. P. (2019). Towards the neural population doctrine. Current Opinion in Neurobiology, 55, 103–111.

[58] Seung, H. S. & Sompolinsky, H. (1993). Simple models for reading neuronal population codes. Proceedings of the National Academy of Sciences of the United States of America, 90, 10749–10753.

[59] Shamir, M. & Sompolinsky, H. (2006). Implications of neuronal diversity on population coding. Neural Computation, 18, 1951–1986.

[60] Shannon, C. E. (1949). Communication in the Presence of Noise. Proceedings of the IRE, 37, 10–21.

[61] Sofroniew, N. J., Vlasov, Y. A., Hires, S. A., Freeman, J., & Svoboda, K. (2015). Neural coding in barrel cortex during whisker-guided locomotion. eLife, 4.

[62] Sreenivasan, S. & Fiete, I. (2011). Grid cells generate an analog error-correcting code for singularly precise neural computation. Nature Neuroscience, 14, 1330–1337.

[63] SRJ & Everitt, B. S. (1999). The Cambridge Dictionary of Statistics. Journal of the American Statistical Association.

[64] Stringer, C., Michaelos, M., & Pachitariu, M. (2019). High precision coding in visual cortex. High precision coding in mouse visual cortex, p. 679324.

[65] Taube, J. S., Muller, R. U., & Ranck, J. B. (1990). Head-direction cells recorded from the postsubiculum in freely moving rats. I. Description and quantitative analysis. Journal of Neuroscience.

[66] Van Hateren, J. H. & Ruderman, D. L. (1998). Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. Proceedings of the Royal Society B: Biological Sciences, 265.

[67] Wang, W., Chan, S. S., Heldman, D. A., & Moran, D. W. (2007). Motor cortical representation of position and velocity during reaching. Journal of Neurophysiology, 97, 4258–4270.

[68] Wang, Z., Stocker, A., & Lee, D. (2016). Efficient neural codes that minimize Lp reconstruction error. Neural Computation, 28.

[69] Wei, X. X., Prentice, J., & Balasubramanian, V. (2015). A principle of economy predicts the functional architecture of grid cells. eLife, 4, 1–29.

[70] Wei, X. X. & Stocker, A. A. (2012). Efficient coding provides a direct link between prior and likelihood in perceptual Bayesian inference. Advances in Neural Information Processing Systems, 2, 1304–1312.

[71] Welinder, P. E., Burak, Y., & Fiete, I. R. (2008). Grid cells: The position code, neural network models of activity, and the problem of learning.

[72] Wilke, S. D. & Eurich, C. W. (2002). Representational accuracy of stochastic neural populations. Neural Computation, 14, 155–189.

[73] Yaeli, S. & Meir, R. (2010). Error-based analysis of optimal tuning functions explains phenomena observed in sensory neurons. Frontiers in Computational Neuroscience, 4, 1–16.

[74] Yarrow, S., Challis, E., & Seriès, P. (2012). Fisher and Shannon information in finite neural populations. Neural Computation, 24, 1740–1780.

[75] Yartsev, M. M., Witter, M. P., & Ulanovsky, N. (2011). Grid cells without theta oscillations in the entorhinal cortex of bats. Nature, 479.

[76] Yerxa, T. E., Kee, E., DeWeese, M. R., & Cooper, E. A. (2020). Efficient sensory coding of multidimensional stimuli. PLoS computational biology, 16, e1008146.

[77] Zhang, K. & Sejnowski, T. J. (1999). Neuronal tuning: To sharpen or broaden? Neural Computation, 11, 75–84.

[78] Zhaoping, L. (2014). Understanding Vision: Theory, Models, and Data. Perception, 17.

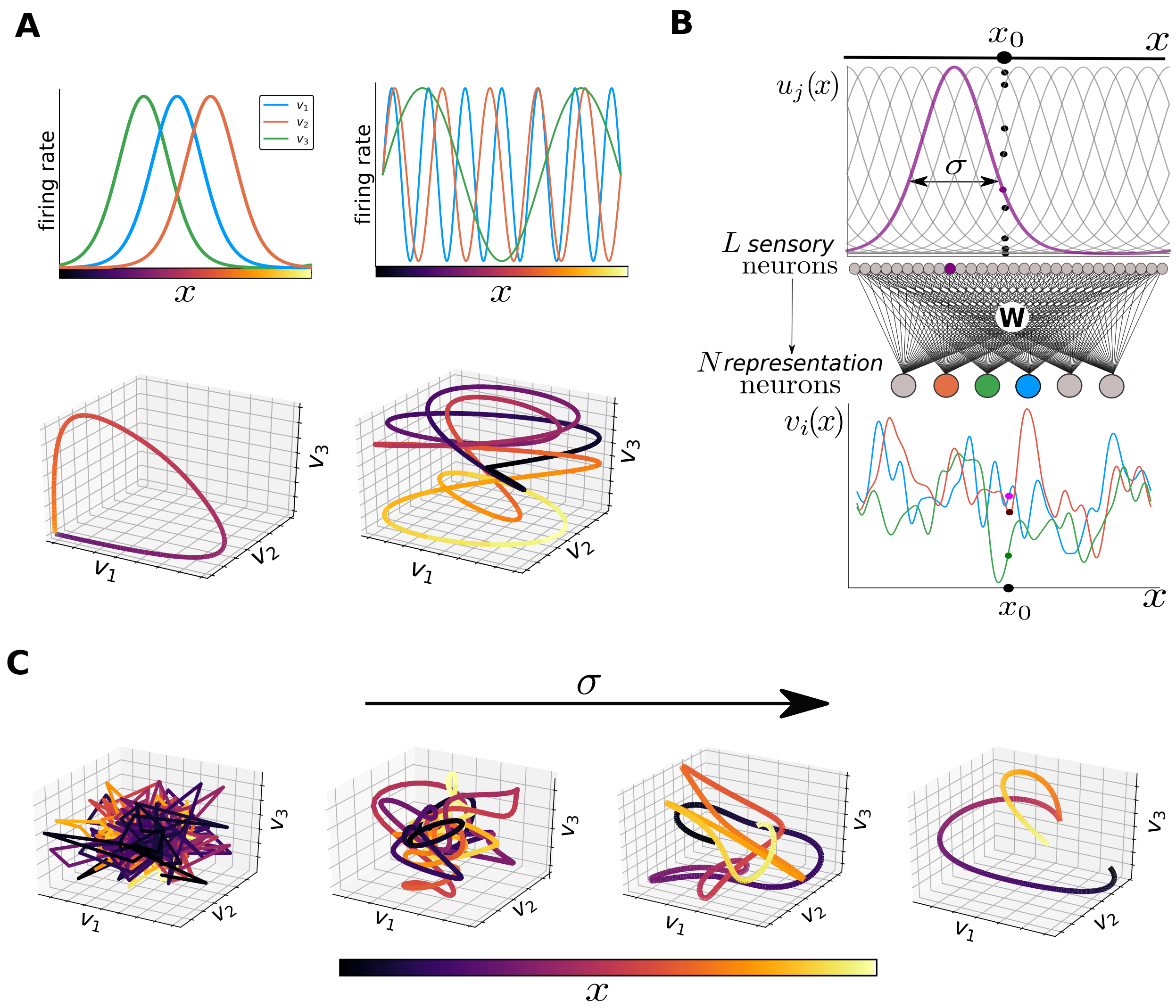

Figure 1: **Geometrical approach to coding, and the random feedforward neural network architecture.**(**A**) Top: mean responses of neural populations encoding a one-dimensional stimulus. Left: population of neurons with Gaussian, translationally invariant tuning curves. Right: population of neurons with periodic tuning curves. We note that grid cells are composed of periodic Gaussian activity bumps, and are thus not sinusoidal. For the sake of illustration, we plotted three sinusoids with three different periods. Bottom: joint activity of the neural population, as a function of the stimulus value, colored according to the legend, corresponds to a one-dimensional manifold in a N-dimensional space. We show a three-dimensional subspace, corresponding to the responses of the highlighted neurons. Unimodal tuning curves (left) evoke a single-loop manifold, which preserves the distances between stimuli in the evoked responses. Instead, periodic tuning curves (right) evoke a more complex manifold, and it can happen that two distant stimuli are mapped to nearby points in the activity space. At the same time, the manifold is longer, and fills up a larger portion of the possible activity space. (**B**) Feedforward neural network. An array of L sensory neurons with Gaussian tuning curves (one highlighted in purple) encodes a one-dimensional stimulus into an high dimensional representation. These tuning curves determine the response of the population for a given stimulus, x_0 (dots). This layer projects onto a smaller layer of N representation neurons with an all-to-all random connectivity matrix $\mathbf{W}$, generating irregular responses. We plotted the tuning curves of three sample neurons, highlighting their response to the stimulus x_0. (**C**) Example of joint activity as a function of the stimulus, colored according to the previous legend, of three sample neurons of the second layer, for increasing σ. When $\sigma \to 0$ (left), neurons generates uncorrelated random responses to different stimuli, generating a spiky curve made up by broken segments. As σ grows, irregularities are smoothed out, and nearby stimuli evoke increasingly correlated responses. By decreasing the complexity of the curve, we ultimately recover the scenario of unimodal tuning curves (right).

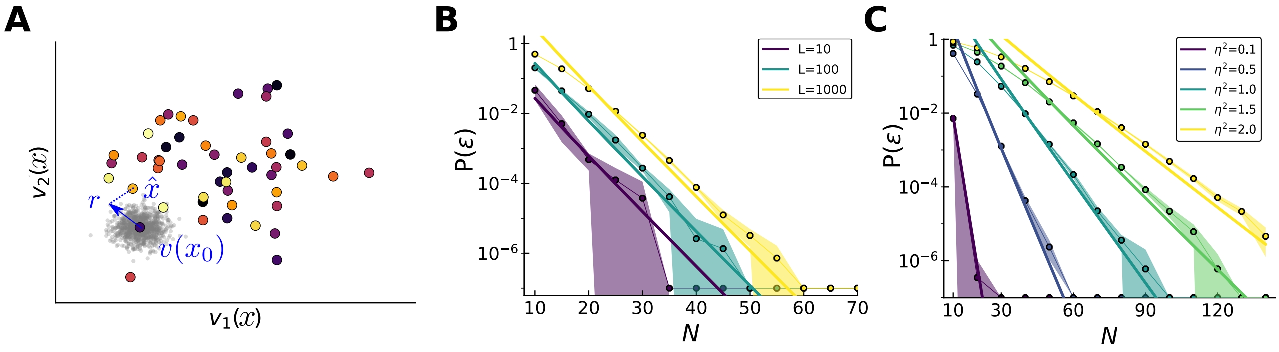

Figure 2: **Error probability for discontinuous random responses.** (**A**) Joint responses of two neurons to $L = 50$ stimuli, colored according to the previous legend. Noise is represented as a cloud of possible responses (in grey) around the mean. An error occurs when the noisy response $\mathbf{r}$ happens to be closer to a point representing another stimulus $\hat{x}$ than the true one x_0. Since responses are uncorrelated, that point may represent a distant stimulus. (**B**) Theoretical (solid curves) and numerical results (circles) for the probability of error as a function of the population size, for different numbers of discrete stimuli encoded with uncorrelated random responses ($\eta^2 = 0.5$, averaged over 8 network realizations, shaded region corresponds to 1 s.d.). The error probability scales exponentially with the number of neurons, with a multiplicative constant given by the number of stimuli. The high variance is due to the difficulty in estimating probabilities when they are very low. (**C**) Theoretical (solid curves) and numerical results (circles) for the probability of error as a function of the population size for $L = 500$ discrete stimuli, for different noise magnitudes. Results are averaged over 8 network realizations, shaded region corresponds to 1 s.d.

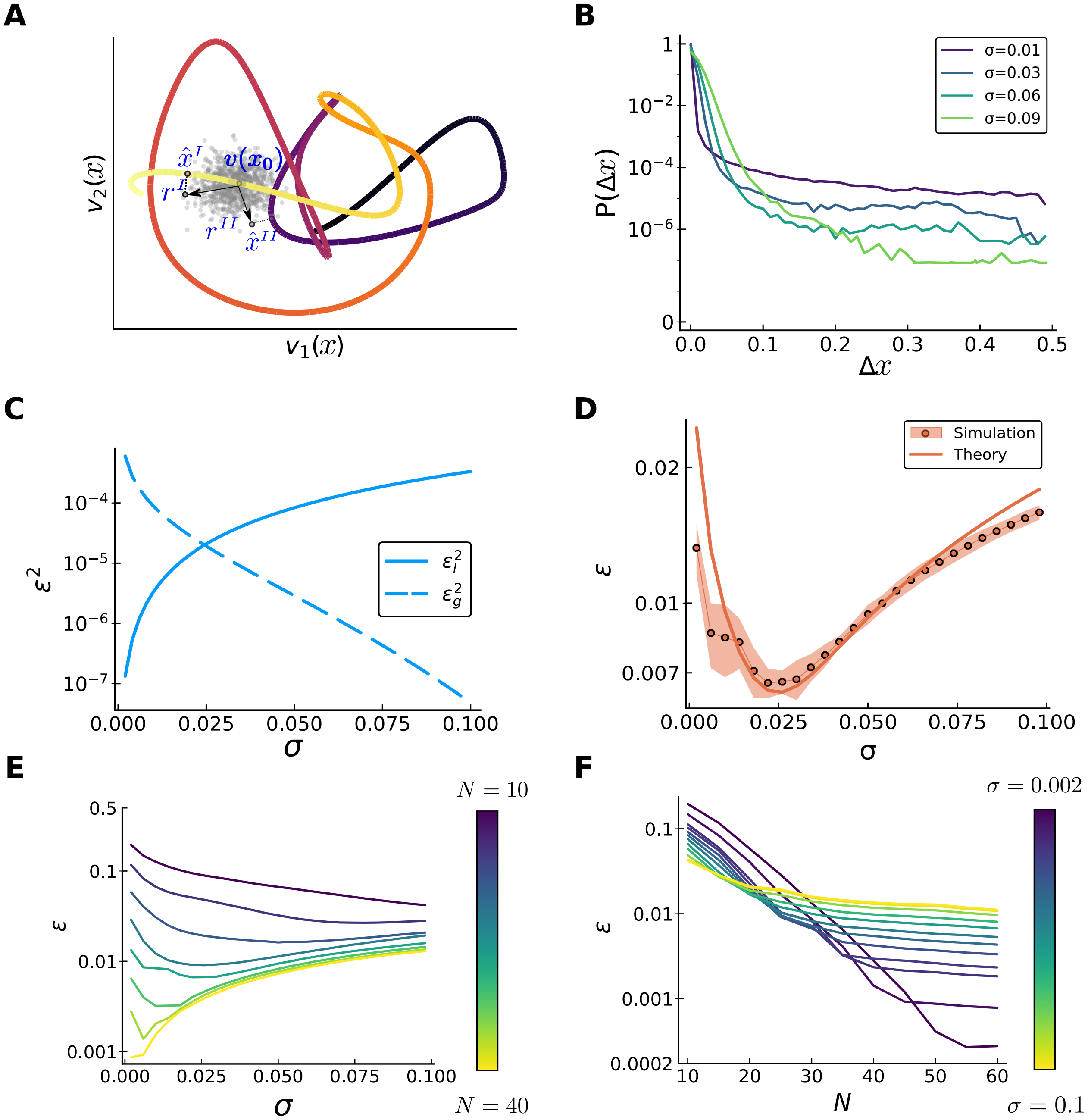

Figure 3: **Trade-off between local and global errors.** In all simulations, $L = 500$ and $\eta^2 = 0.5$, $R = 1$.
continue to next page

Figure 3 *(previous page)*: (**A**) Different types of error in a complex curve of mean population activities (joint response of two neurons, colored according to previous legend). Here, $\mathbf{r}^I$ and $\mathbf{r}^{II}$ are two possible noisy responses to the same stimulus, extracted from the Gaussian cloud surrounding the mean response, $\mathbf{v}(x_0)$. An ideal decoder will output the stimulus corresponding to the closest point of the curve. In one case, $\mathbf{r}^I$ will cause a local error, falling on a point of the curve that represents a similar stimulus, $\hat{x}^I$. Instead, $\mathbf{r}^{II}$ happens to be closer to a point of the curve which represents a stimulus quite far from the true one, $\hat{x}^{II}$, causing a catastrophic error. (**B**) Normalized histogram of absolute errors, $\Delta x = |\hat{x} - x|$, made by an ideal decoder, for different values of σ ($N = 25$). We tested the response to 10^7 stimuli, uniformly spaced between $[0, 1]$, and we averaged over 8 realizations of the connectivity matrix. For better visualization, we considered a stimulus with periodic boundary conditions, such that all global error magnitudes have the same probability. Contributions of the two types of error varies with σ. For small σ, coding is very precise locally (fast drop of the purple curve for small errors), but we have a great number of global errors (tail of the distribution is high). Vice versa, smoother codes (green curves) yield poor local accuracy (larger local errors), but high noise robustness (very few large scale errors). (**C**) Theoretical prediction for the two contributions to the MSE (log scale) in function of σ ($N = 30$). The magnitude of local errors increases with larger widths (solid curve), while the number of global errors decreases (dashed curve). (**D**) Root-MSE as a function of σ: comparison between numerical simulations (solid curve) and theoretical prediction of Eq.(6) (dots). Results are averaged over 8 network realizations, shaded region corresponds to 1 s.d. (**E**) Root-MSE, as a function of σ for different population sizes N (increasing from violet to yellow). The optimal error is attained at an optimal $\sigma^*(N)$, which decreases with increasing N. (**F**) Same data, but the error is showed as a function of N, for a fixed value of σ. The error at first decreases exponentially fast until global errors are suppressed, then the local errors are linearly reduced. Decreasing σ, we increase the N at which the transition occurs, but also the error at this critical value.

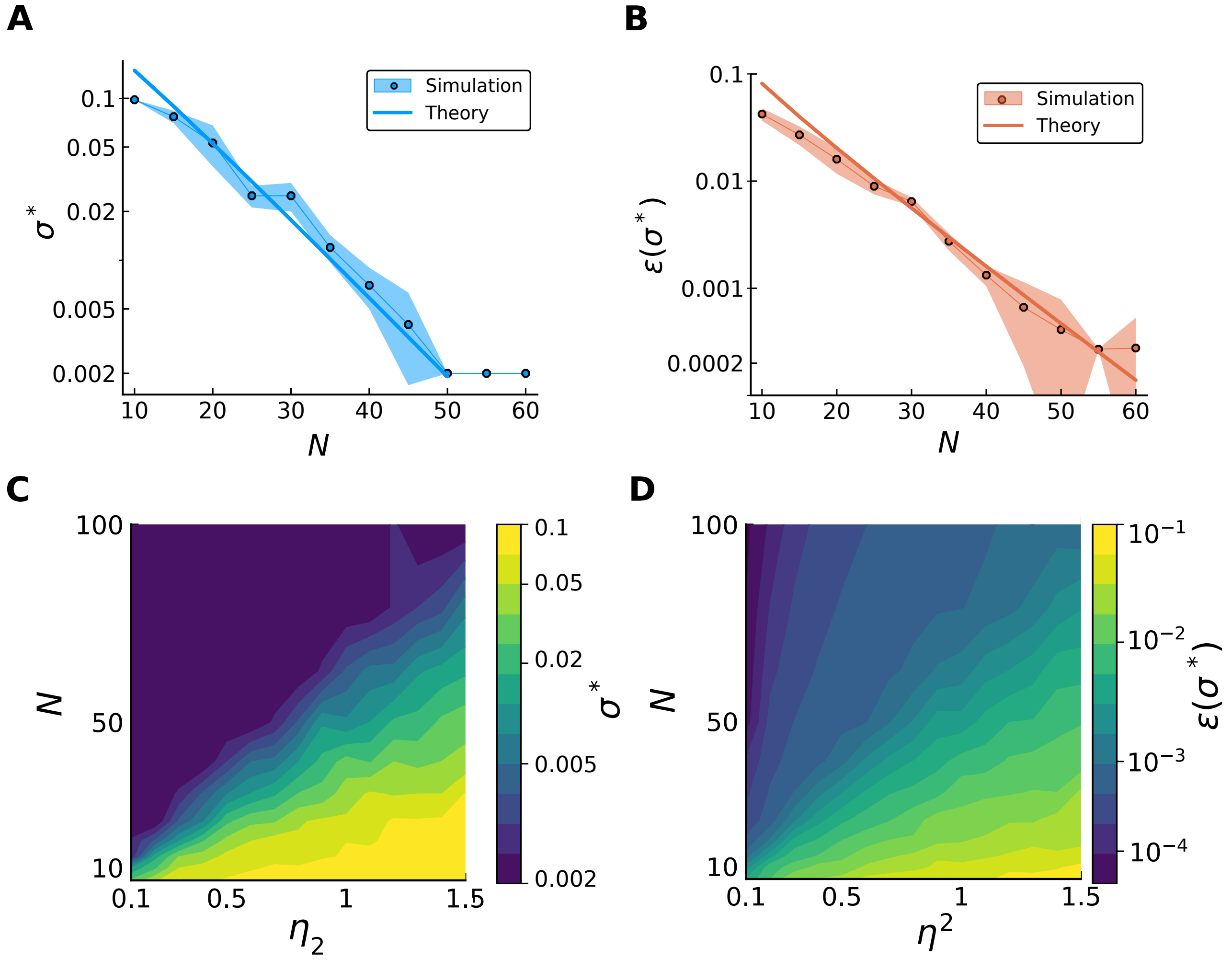

Figure 4: **Scaling of the optimal width and optimal error in function of population size and signal-to-noise ration.** In all simulations $L = 500$, and results are averaged over 8 network realizations. In (**A-B**) $\eta^2 = 0.5$. (**A**) The mean optimal σ^* decreases exponentially fast with the number of neurons, saturating the lower bound imposed by the finite number of neurons of the first layer (corresponding to the spacing of the preferred positions, $1/L$). Simulations (circles) show good agreement with the theory (solid line). Shaded region corresponds to 1 s.d. (**B**) As a consequence, the optimal error, $\varepsilon(\sigma^*)$, which is linear in σ, is also suppressed exponentially fast in N. As before, simulations (dots) are well predicted by the theory (solid curve). (**C,D**) Optimal width (**C**) and error (**D**) as a function of the parameters $N - \eta^2$. The color code is in log scale, used in order to highlight the exponential scaling.

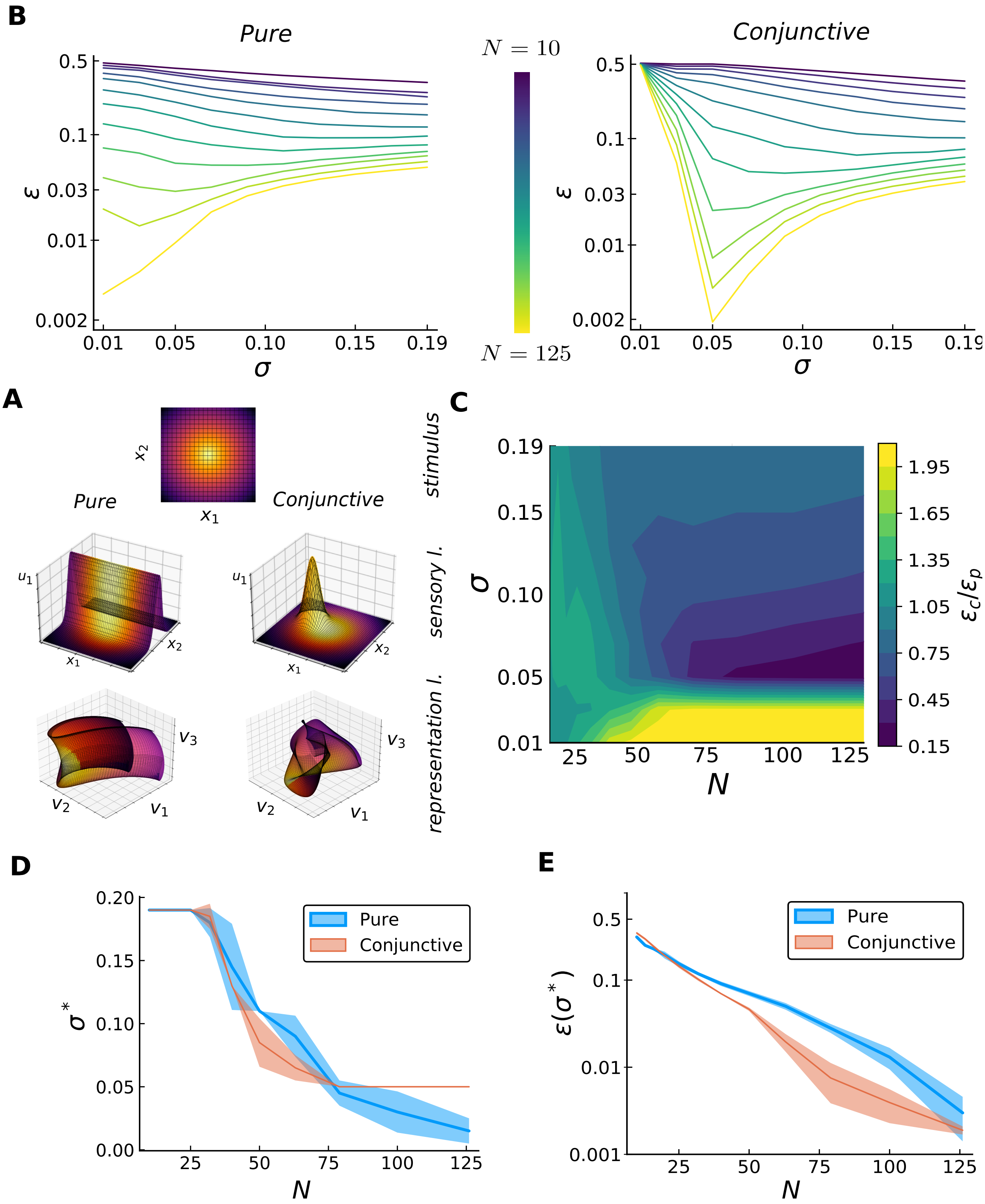

Figure 5: **Compressed coding for high-dimensional stimuli.** For numerical results, we illustrated the case of a three-dimensional stimulus, $L = 3375$, $\eta^2 = 1$, $R = 1$; results are averaged over 8 network realizations.

continue to next page

Figure 5 *(previous page)*: (**A**) Subsequent mapping of high-dimensional stimulus space into neural activity operated by the two layer coding scheme. Top: two-dimensional stimulus space, colors serving as legend for following plots. Middle: firing rate (z-axis) of a sample sensory neuron, for the two cases, as a function of the two stimulus coordinates (x- and y- axis), colored according to the previous legend. In the pure case (left), a single sensory neuron 'fold' the two-dimensional sheet across a direction, specified by its preferred position and dimension (here, x_2). In the conjunctive case (right), a sensory neuron creates a 'bump' in the sheet. Bottom: joint activity of three representation neurons as a function of the stimulus, colored according to the previous legend. Each of these neurons will randomly sum the transformations of sensory neurons, producing a randomly 'folded' sheet in the pure case (left) and a 'crumpled' sheet in the conjunctive case (right). (**B**) Root-MSE as a function of σ for different population sizes N (increasing from violet to yellow), when the first layer consists of pure (left) or conjunctive (right) cells. An optimal σ, decreasing with N, allows the balance between local and global errors, similarly to the one-dimensional case. In the conjunctive case the rapid increase of the error below $\sigma = 0.05$ is due to the sensory neurons not tiling the space, and it is independent from N. (**C**) Mean ratio between the error in the two cases, $\varepsilon_c/\varepsilon_p$, as a function of σ and population size. Yellow (violet) region indicates an outperformance of the pure (conjunctive) population. To aid visualization, the yellow region indicates all the values greater than 2. This regime of small sigma values is characterized by a better coverage of the pure population, independently of the output layer population size. Values greater than 1 occur also when N is small, due to the prefactor of the global error being lower in the pure case. As soon as N is sufficiently large and σ is sufficiently large to allow for good coverage of the stimulus space, the conjunctive case outperforms the pure case. This effect is stronger in the low σ region, due to the slower scaling of the global errors in the pure case. On the other hand, when sigma is sufficiently large the ratio saturates at the value given by the ratio of the local errors. (**D,E**) Optimal tuning width (**D**) and relative error (**E**), for pure (blue) and conjunctive (red) cases. Shaded region corresponds to 1 s.d. The global error decreases more slowly in the pure population, as one can see from both the optimal width and the total error being larger and with a smaller slope. At very low population sizes, the difference in the prefactor of global errors leads to slightly better performance of the pure code. For $N \gtrsim 75$ the optimal width in the conjunctive case saturates, due to the loss of coverage. The relative error stops decreasing exponentially and starts decreasing only linearly, while the pure population does not suffer from this limitation. Ultimately, since the optimal width will continue to decrease in the pure population, the error will become lower than the error in the conjunctive case.

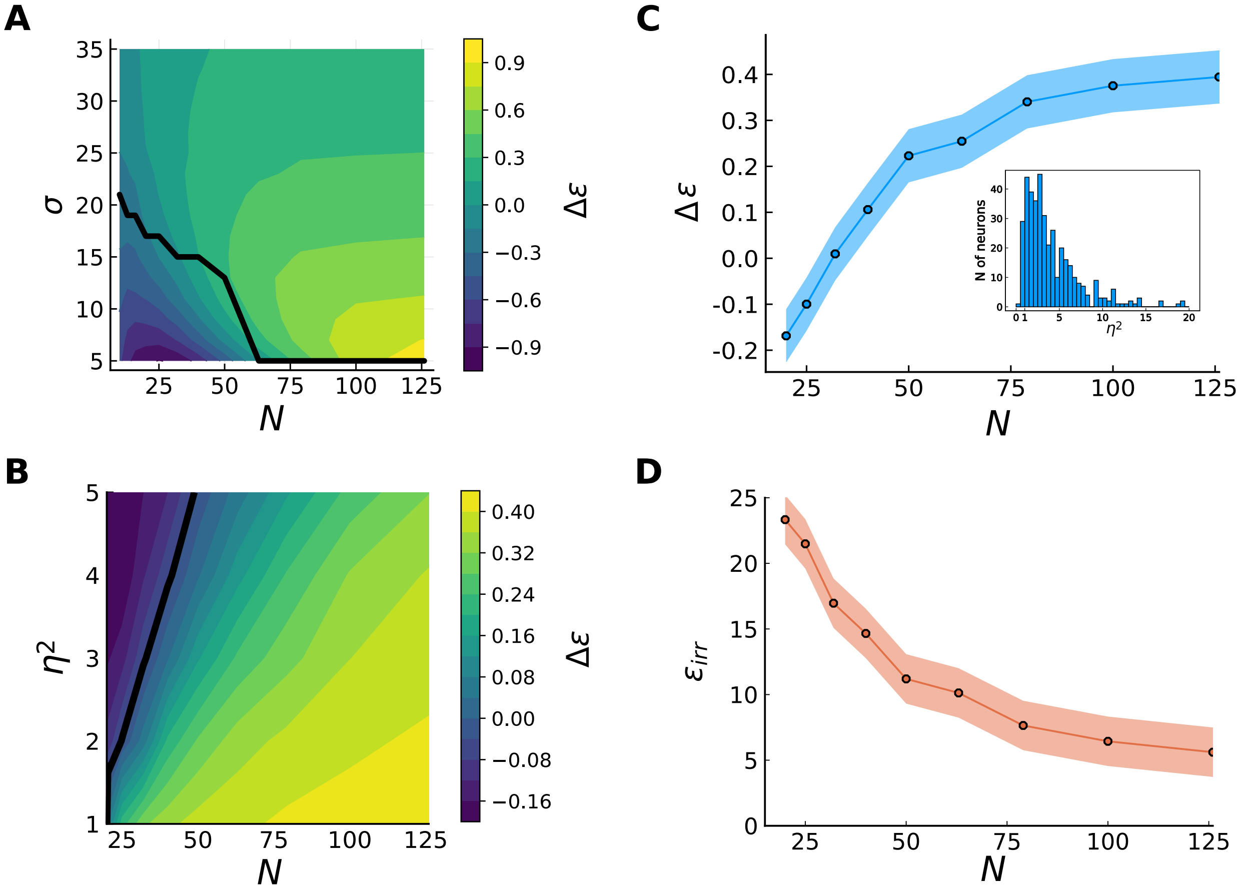

Figure 6: **Linear vs irregular tuning.** (**A**) MPI of the irregular population (averaged over 8 different pools of a given size) compared to the linear one, as a function of population size and tuning width. The black line indicates the critical values of $N-\sigma$ at which they perform equally. In the region below (violet) global errors heavily affect the irregular population, making a smoother code more efficient. With increasing N, global errors become more rare while irregularities improve the local accuracy of the code (yellow region). The advantage increases at smaller values of σ, but so does the required value of N for the irregular population to be advantageous. (**B**) Mean- MPI (over 8 different pools of a given size) of an irregular population, generated with the data-fitted model, compared to the linear one, as a function of $N-\eta^2$. At small population sizes, the irregular tuning produces global error and smoother tuning curves perform better (violet region, $\Delta\varepsilon < 0$). By increasing N, global errors are suppressed and irregularities improve the local accuracy (yellow region,$\Delta\epsilon > 0$). The black line marks the transition values. (**C**) MPI and (**D**) Root-MSE of the irregular population as a function of population size (8 different pools of neurons), for the noise model extracted from data. Shaded region corresponds to 1 s.d. A noise variance is assigned to each neuron, obtaining a very heterogeneous distribution of noise in the population, showed in the inset. For low levels of N, linear tuning produces better results. At $N\sim 40$, the higher local accuracy compensates for global errors, and the irregular code starts to perform better, although the error is still substantial. The improvement saturates to a finite value of ~ 0.4 at a value of $N \sim 100$, when global errors are fully suppressed; the scaling of the error as a function of the population size is no more exponential, but only hyperbolic.

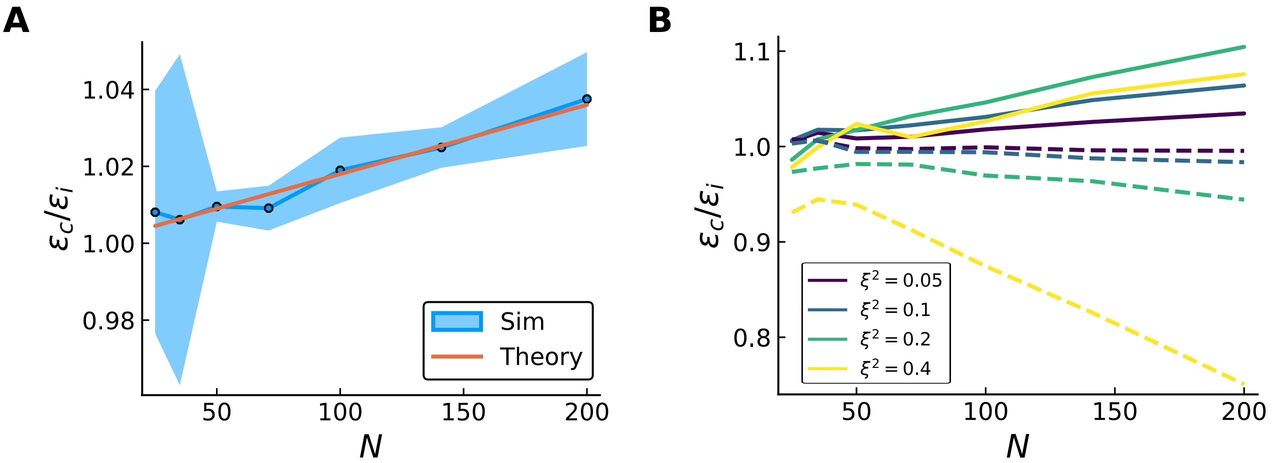

Figure 7: **Effects of correlated noise on compressed coding.** (**A**) Error ratio (MSE) between correlated noise due to input noise and variance-matched diagonal noise, as a function of N, and theoretical prediction, Eq.(10). $\sigma = 0.045$, $\tilde{\eta}^2 = 0.5$ and the contribution of input noise is small, $\xi^2 = 0.05$. Results areveraged over 8 network realizations, shaded region corresponds to 1 s.d. High variability for low values of N is expected, as global errors are more dependent on the specific realization of the weights. (**B**) Error ratio between correlated noise due to input noise and diagonal noise (solid lines), and error ratio between correlated noise with random covariance matrix and diagonal noise (dashed). Different colors denote different contributions coming from the off-diagonal terms ξ^2, increasing from violet to yellow, when the variance-matched noise is kept fixed, $\tilde{\eta}^2 = 0.5$. When correlations come from shared connections, the ratio is positive since we have information-limiting correlations. Their effect are a non-linear function of $\xi^2/\tilde{\eta}^2$, due to the competition between the first order (positive) and second order (negative) corrections. With a random covariance matrix, correlations decrease the error and enhance decoding precision.

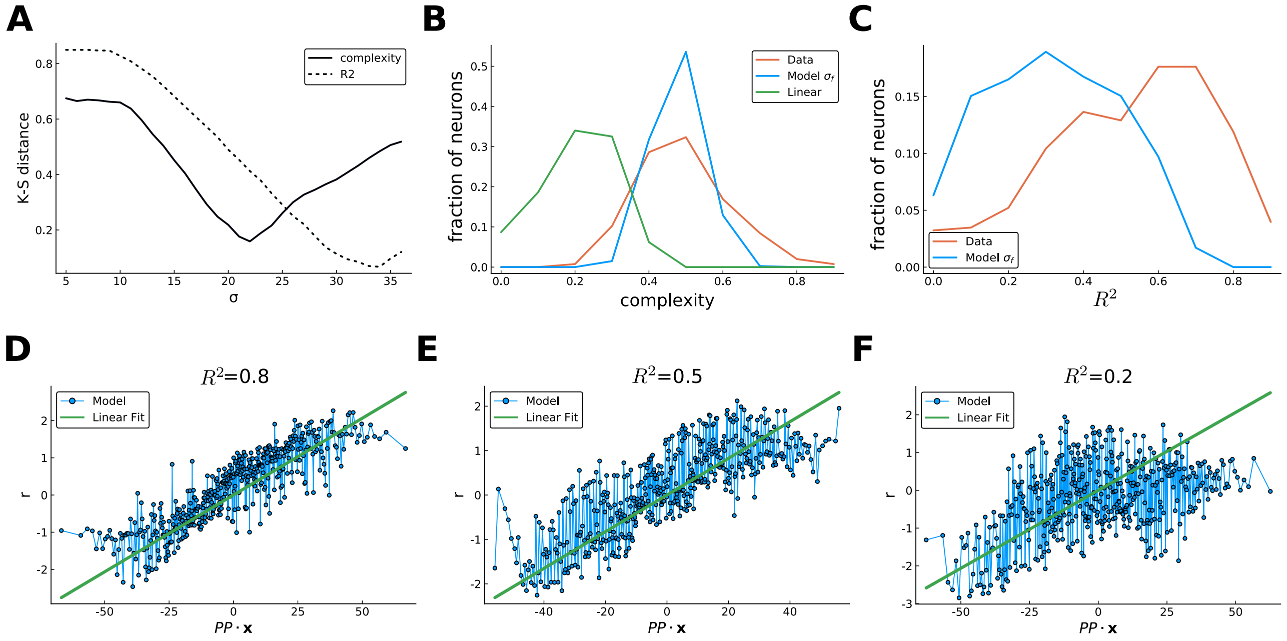

Figure 8: **Model fitting and tuning curves.** (**A**) Kolmogorov-Smirnov distance between the distributions of complexity measure (full line) and R^2 of fitting (dashed) across neurons from the data and the model for different σ. σ_f is chosen to be the value at which the minimum of the distance between complexity distributions is attained, $\sigma_f \sim 22$. (**B**) Normalized histogram of the distribution of complexity measure (arbitrary units) across the neurons of the data (red), the irregular population at σ_f (blue) and a linear population (green). The model is able to capture the bulk of the distribution of the real data much better than a linear model. Nevertheless, the data show a much broader distribution across the population. (**C**) Normalized histogram of the distribution of the R^2 of linear fit across neurons of the data and the irregular population at σ_f (red). Both distributions are broad, but the data show a more consistent linear part. (**D-F**) Three examples of tuning curves of the irregular population at σ_f, showing a broad range of behavior with respect to the linear fit. The tuning curves are plotted in function of the projection of the stimulus (target position) onto a preferred position, obtained by the fit with Eq.(9) (green line). Some neurons are well described by the parametric function (d), some others show consistent deviations (e), while in others the linear behavior is absent (f). This is reflected in the broadness of the distribution of the R^2.

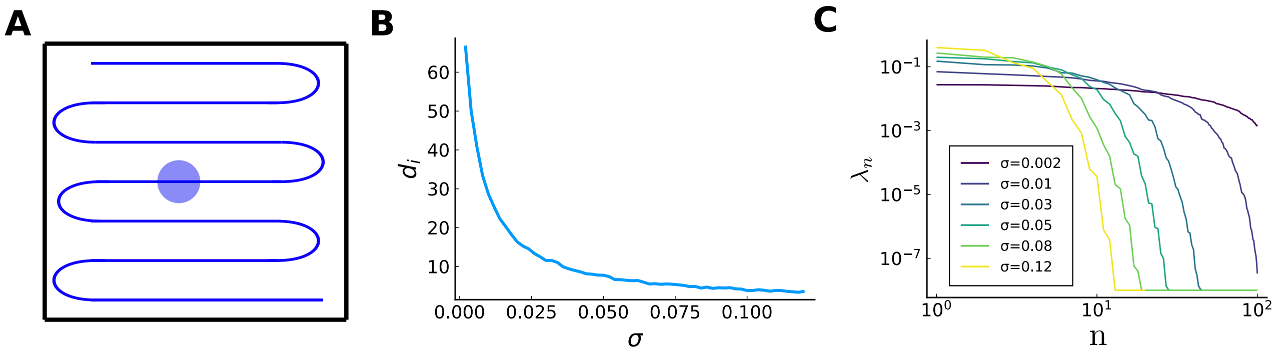

Figure 9: **Geometry of population response** (**A**) 'Efficient mapping of a line into a square', adapted from Fig. 4 of [60]. A continuous stimulus space, the blue line, is mapped into a two-dimensional signal space. The noise (shaded blue) creates a region of ambiguity around the encoded stimulus, which will yield a small 'local' error as the length of the line increases. Nevertheless, after a certain noise threshold, there will be ambiguity between the true stimulus and stimuli belonging to the upper or lower part of the curve, yielding a large scale error. (**B**) Intrinsic dimensionality, defined as the Participation Ratio, of the manifold of joint neural responses (Fig. 1C), as a function of σ, for $N = 100$. When $\sigma \to 0$, the intrinsic dimensionality approach the maximum value of N, with responses scattered in all available directions. As σ increases, the responses are increasingly correlated, and the neural responses are arranged along few directions. (**C**) Spectrum of the eigenvalues of the covariance matrix of neural responses ($N = 100$), for different values of σ. As the covariance function is Gaussian, the spectrum is rather flat until a crossover value (or cutoff), and then falls rapidly to 0. The cutoff value is inversely proportional to σ.