Random Compressed Coding with Neurons

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

April 19, 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 [48, 42, 82, 66, 16, 25, 50]. 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

*Current affiliation: Department of Applied Science and Technology (DISAT), Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino tuning curves are arranged in such as way as to achieve the most accurate coding possible given a constraint on the neural resources engaged [9, 4, 59]. The latter is often interpreted as an metabolic constraint on the maximum firing rate of the single neuron or on the mean firing rate of the whole population [94, 12, 84].

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 **[PLEASE DON'T CHANGE CORRECT ENGLISH INTO BAD ENGLISH!!! IT MAKES ME QUITE ANGRY AS IT IS A BIG WASTE FO TIME. YOU SHOULD BE MORE CAREFUL, AND CHECK THE SUGGESTIONS OF YORAM BEFORE MAKING CHANGES!.]** when information is decoded from the activity of a population of model neurons [94, 26, 90, 38, 34]. In the resulting optimal populations, and if noise among neurons is independent, the coding error typically scales like $1/\sqrt{N}$ (or $1/N$ some cases, where the tuning width can be optimized accordingly**[We should discuss this, and, at a minimum, this should be explained better. Just a parenthesis is not enough]**), where N is the number of model neurons [74, 10, 53]. 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.

This is true in 1d, should we specify?

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 [44, 52, 92, 27], 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 [49, 77, 58, 41, 30]. 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 [32, 85], grid cells can represent spatial location with an accuracy which scales exponentially (rather than algebraically, as above) in the number of neurons [78, 65, 18]. 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 **[SAME REMARK!!!!]** 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 **[SAME REMARK: YOU SIMPLY IMPLEMENTED YORAM'S REMARK, BUT HIS WORDING WAS LOGICALLY WRONG!!!!! THIS HAS NOTHING TO DO WITH DECODING; THE ERROR COMES FROM AMBIGUOUS ENCODING!!!!]**, 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 **[WHY DID YOU NOT FINALIZE THIS? YOU WERE SUPPOSED TO SEND ME A FINALIZED VERSION!!! YOU SHOULD HAVE ASKED YORAM FOR REFERENCES AND LOOK AT**

'We may need to relate to other works on transformation between

THEM. FURTHERMORE, THIS IS NOT THE RIGHT PLACE FOR SUCH COMMENTS; THEY SHOULD BE ENTERED IN TEH DISCUSSION.].

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 **suspection [YOU DIDN'T CORRECT THIS, EVEN THOUGH IT WAS POINTED OUT BY YORAM!]**, 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. **??**A). Simple tuning curves generate a smooth curve, implying that similar stimuli 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 [88, 78]. 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. **[AGAIN, YORAM HAD A QUESTION HERE THAT YOU DID NOT ADDRESS!]**

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. **??**B. In the first layer, a large population of L *sensory* neurons encode 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. **[YOU USE R LATER FOR SOMETHING ELSE!!! I REPLACED R BY r, HERE.]** 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 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,

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

by choosing the value of the prefactor in Eq. (1), A. 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, 28, 96] **[IF YOU CITE ATICK'S PAPER, THEN ALSO CITE VAN HATEREN]**.

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. **??**C). 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 **[IN ENGLISH, ALL VERBS IN A SENTENCE/PASSAGE MUST BE IN TEH SAME TENSE!!!]** out' and winds in such a ways as to fit within the allowed space of population response 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 Sec.2, 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 Sec.4.

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. **??**C). 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 stimulus with values $x_j = j/L$, with $j = 1, \ldots, L$.

Each of these stimuli elicits a mean response

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

in neuron i of the second layer. 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. **??**A 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. **??**A). 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. **??**A). 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. **??**A and Eq. (4)), errors of any magnitude are equiprobable. As a result, in a model with narrow sensory tuning curves which result 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 decoded makes a mistake.

In Methods, we provide a derivation of this quantity in the case where it is much smaller than. 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. **??**B). 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. **??**A). 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 (Fig. **??**A); 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. **??**C). 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. **??**B). 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

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

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. ??C). 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., [25]). 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 [74, 17], 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 error dominate (Fig. ??C). 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. ??D, 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. ??E). Furthermore, the strong (exponential) reduction in MSE with increasing N occurs only up to a crossover value depending on σ (Fig. ??F); 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. ??D 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. ??A and B). Equation (8) and Fig. ??B 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. ??C 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 theese two coding schemes as *conjunctive* and *pure*, following Ref. [33] (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 that is exponential in the number of dimensions **[REWORD: a "constraint" cannot be "exponential"!]**.

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. **[[***Figure* **??***A illustrates the MSE as a function of the width of the (possibly multi-dimensional) tuning curves in the sensory layer. The error behavior is similar to the one-dimensional case, with an optimal width which balances the two contributions to the error, but there are some qualitative differences in the two cases. In order to get an intuition about these differences, it is instructive to consider how the stimulus space is mapped into a K-dimensional manifold described by the N-dimensional activity patterns of representation neurons (similarly to what is shown in Fig.* **??***C for a one-dimensional stimulus). For the sake of illustration, we showed how a two-dimensional sheet is subsequently transformed by the two layers to produce a complex manifold (Fig.* **??***B). In the pure case, each neuron of the sensory layer can be thought as 'folding' the sheet across a direction, which depends on its preferred position and dimension. Representation neurons, in turn, randomly sum these transformations operated by upstream neurons. As a result, the final manifold will look like a planar sheet repeatedly folded across some random directions. On the other hand, a sensory neuron which is conjunctively tuned to both stimulus dimensions, will create a 'bump' in the planar sheet. If each representation neuron randomly sum these bumps, the total transformation will rather look like a 'crumpled' sheet.*

Local errors depend on the curvature of the manifold, specified in both cases by the value of σ. In the pure case, the folding directions will be flatter and, therefore, more sensitive to noise. Instead, in the conjunctive case, the sheet will be curved more or less uniformly in all directions; this will lead to a lower local error in the conjunctive scheme. We notice that this advantage, with our choices of comparison between populations (equal size and equal response variance across stimuli, see Methods), would be present already if stimuli were decoded from a noisy sensory layer, similarly to what has been obtained in [33]. As for global errors, there are differences in the generalization of Eq.(5). In the pure case, the tuning curves in the representation layer are linear superposition of the one-dimensional ones of sensory neurons. Neglecting global errors occurring in more than one stimulus direction at a time, which are rare if N is large enough and provided that σ is not too low, the probability of a global error is simply the sum of probability of global errors along each stimulus dimension. This results in a prefactor scaling linearly, K/σ. The rate of the exponential scaling is governed by the signal-to-noise ratio; since the variance of the signal across a single dimension will be $\sim 1/K$ of the total variance, also the rate will be reduced by a factor of K. In the conjunctive case, in extending the prefactor of Eq.(5), we have to consider the probability of error between 'balls' of radius $1/\sigma$. There are $\sim 1/\sigma^K$ such balls, this leads to a scaling which is exponential with stimulus dimensionality. Since in this case global errors will not happen independently for each stimulus dimension, the rate of the exponential scaling is preserved, and will correspond to the total signal-to-noise ratio. **YOU HAVE TO REWRITE THIS COMPLETELY: First, the order in which you say things appears random (you first talk about a final result—MSE—then you go back to more basic points, then you give scaling arguments, etc.); second, you use 'manifold', then 'sheet', you don't explain the difference between 'folded' and 'crumpled', then you talk about 'bump', 'balls', etc. It's way too messy/sloppy. Start by explaining the difference between 1D and higher-D, the geometry clearly, then the intuitive implications for the various errors, then the result. Define clearly all teh terms you use (e.g., 'manifold'), and make sure they apply correctly. Explain whay you mean by 'folded' and 'crumpled' and what's the difference. Make shorter paragraphs (your paragraphs are too long, and don't seem to have a beginning and an end: each paragraph should have a clear topic and a clear flow. Etc. You have to rewrite all this from scratch.]]]**

We quantified numerically 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 by plotting the ratio between the MSE **[You had written 'error' here, which was imprecise. If we defined 'MSE', then use this consistently.]** in the two cases (Fig. **??**C). (See Methods for detailed analytical calculations.) **[[***It is possible to distinguish* *different regions in this landscape* **[What is a 'landscape'???!!!]**. *When N is low, global errors are dominant in both coding schemes, but the multiplicative factor of the global error (Eq.(51) - (53)) is lower in the pure case. We notice that this is not a good coding regime, as the error is typically large. As soon as the size of the representation layer increases, the balance between local and global errors becomes less trivial and the advantages of the conjunctive scheme emerge. In the high N - high σ region, where local errors are dominant, the conjunctive population outperforms the pure one, yielding a lower local error (Eq.(50)-(52)). The performance of the conjunctive scheme increases even further as compared to the pure scheme, due to faster suppression of global errors, in the large N and moderately low σ region. However, this scaling breaks down at very low values of σ, where a population with pure selectivity tiles the space more efficiently than one of conjunctive neurons of the same size. This phenomenon is due to the limited first layer size, and therefore does not depend on N, and it will be stronger for increasing stimulus dimensionality.*

This complex relative advantage of one scheme with respect to the other is also reflected by the scaling of the optimal width and its relative error (Fig. **??***D,E). For low values of N, a pure code is more advantageous. Increasing the population size, the exponential scaling of the optimal width and the optimal error is stronger for the conjunctive case. Nevertheless, the optimal width has a higher lower bound, imposed by the finite number of neurons of the first layer. Once this bound is reached, the error suppression is hyperbolic in N. On the other hand, in the pure case, lower values of σ still allow the coverage of the stimulus space. The exponential scaling regime, both for the optimal width and the optimal error, is maintained for larger values of N. To summarize, these results emphasize the complex trade-offs in encoding multi-dimensional stimuli. By considering limitations in the downstream structure, we explored under a different light the relative advantage of having neurons tuned to single or multiple stimulus dimensions, a question which was the subject of recent theoretical investigations [33, 45]). In the next section, we will interpret experimental results in a specific region of the cortex through the developed theory.* **YOU HAVE TO REWRITE THIS COMPLETELY: First, there is no clearly apparent logic of the order in which you describe things (even if there is a hidden logic in your head—but it's not the reader's job to have to guess it). Second, it's completely impossible to understand your arguments for someone how has not internalized the calculations in Methods; no rewiever will take the time to decode this text, nor should she/he. You have to find a way to describe arguments and results in a way that can be understood by reading the text and, if needed, looking at the figure. (If this is impossible, then one can spell out teh results abd refer to Methods; but I believe that it is possible.)]]]**

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' [51]. 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 [51, 83]. 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 [83]. A recent study [58] 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). **[PLEASE BE ATTENTIVE TO TYPOS: I corrected "include," "behaviour."]** This more elaborate model [58] 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 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 **[AGAIN, TYPO: you had "displays."]** spatially extended and irregular tuning profiles. In fitting our model to the M1 recordings [58], we considered the arrangement of stimuli used in the experiment, namely 27 spatial locations arranged in a $3 \times 3 \times 3$ grid in a 40 cm-high cube. We then followed a previous approach [58, 3]: given the diversity of the irregular tuning curves in the population we did not aim at fitting individual tunng 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. [58]). 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 emplyed previously [58], it yields comparable fits.

In addition to fitting the population of tuning curves, we extracte from the data a quantification of the noise in the response of individual neurons. **[[[***Given the heterogeneous distribution of the variance of firing rates, both across different neurons and across different stimuli, we proceeded in the following way. In simulations, we assigned to each neuron i a specific noise variance* η_i^2 *. For each recorded neuron, we computed a signal-to-noise ratio, as the variance of the mean response across different stimuli divided by the trial-to-trial variance of responses for a single stimulus, averaged over different stimuli:* $\frac{1}{\eta_i^2} = \frac{\left\langle \left(v_i(x) - \langle v_i(x) \rangle_x\right)^2 \right\rangle_x}{\left\langle \langle (r - v(x))^2 \rangle_{r|x} \right\rangle_x}$ *. The data in our possession are not suitable to include also the effect of noise correlations; we leave for the following section a general analysis of the impact of non-diagonal noise covariance matrix in this type of coding scheme.* **YOU HAVE TO REWRITE THIS COMPLETELY: First, it's very unclear. Second, you had written "noise model" in the previous sentence (which I modified), but then you don't talk about any model! Third, don't use such big equations in-line; find another way to express your material.]]]**

With a neural response model in hand, we can evaluate the coding performance; to do so, we consider a finer, $21 \times 21 \times 21$ 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', $\Delta\varepsilon \equiv \left(\varepsilon_{\text{lin}} - \varepsilon_{\text{irr}}\right)/\varepsilon_{\text{lin}}$. $\Delta\varepsilon$ 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. **??**A). 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 **??**B 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. **??**C 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. **??**C, 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. **??**C), the MSE is still appreciable (Fig. **??**D). 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. **??**D). 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 [58].

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 variance-matched **[THIS IS WRONG!!! Strictly speaking, we are matching the variances only on average; moreover, given our new formulation with R instead of 1 this is not matched any longer. You have to rework this to change the coefficient of ξ^2 and to explain the issue that it is matched on average, as clearly as possible!]**, diagonal covariance, $\Sigma_{\text{ind}} = \left(\eta^2 + \xi^2\right)\mathbf{I}$. 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 **[AGAIN, YOU CHANGED SOMETHING THAT WAS PERFECTLY CORRECT!!! YOU JUST CANNOT MAKE CHANGES UNLESS YOU ARE *SURE* AND YOUR *CHECK* CAREFULLY!!!]** 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 [24]

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

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

where $\varepsilon_{l,\text{ind}}^2$ is the corresponding quantity calculated for the reduced covariance matric 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. **??**A), 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. **??**B **[AGAIN, TYPOS: "Figure" instead of "Fig." and an extra ")"!!!]** indicate that there exist 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 **[TYPO: IF YOU SAY "we ask," IT HAS TO BE "results," NOT "resulted."]** 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, $\Sigma_{\text{rand}} = \tilde{\eta}^2\mathbf{I} + \xi^2\mathbf{X}\mathbf{X}^{\mathbf{T}}$, where $X_{ij} \sim \mathcal{N}(0, 1/L)$ **[AGAIN, THIS IS WRONG, because in our new formulation we have R instead of 1.]** 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 manifold 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:

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

This result, as well as numerical simulations (Fig. **??**B), demonstrate 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.**[THIS IS VERY INCOMPLETE FOR A SUMMARY: YOU HAVE TO SUMMARIZE THE RESULTS!!!]** [[*With a straightforward extension of the model, we studied how limitations in the downstream structure affect the optimal arrangement of multi-dimensional tuning curves, dis-*

tinguishing between the two extreme cases of 'pure' and 'conjunctive' selectivity [33, 45] **REWRITE: IT IS VERY UNCLEAR, THE READER HAS NO IDEA WHAT IS MEANT BY "straightforward extension" AND EVEN I HAVE NO IDEA WHAT IS MEANT BY "limitations in the downstream structure"!!! ALSO, THE ENGLISH IS WRONG: GERUNDS (LIKE "distinguishing") MUST HAVE A SUBJECT; FOR EXAMPLE, YOU CAN SAY "Distinguishing two cases, we approached the problem..." BECAUSE THEN "we" DID THE ACTION OF "distinguishing"; YOU CANNOT SIMPLY PARACHUTE A GERUND IN A SENTENCE AND HOPE THAT IT MAKES SENSE]]**. We applied the extension of our model to the case of three-dimensional stimuli to recordings of motor cortex neurons in monkey [58], where we illustrated the advantage of irregularities in spatial tuning curves for the accurate coding of hand position.

Population coding and geometry of neural responses. A large body of literature has addressed the problem of coding low-dimensional stimuli in populations of neurons with simple **[YOU KEEP SAYING 'PARAMETRIC'—BUT EVEN A COMPLEX TUNING CURVE CAN BE PARAMETRIC!]** 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 **[I HAVE ALREADY TOLD YOU MORE THAN ONCE THAT YOU CANNOT SAY "in function" IN ENGLISH, AND THAT THE CORRECT EXPRESSION IS "as a function"]** of population size **[REFERENCES?]**, stimulus dimensionality [94], stimulus geometry [67], time scale of decoding **[DO YOU MEAN TIME SCALE OF ENCODING???]** [12, 90].

***

[I HAVE NOT MADE CHANGES BELOW. YOU HAVE TO REWORK EVERY LINE CAREFULLY.]

Moreover, several studies analyzed the region of the stimulus space which was best encoded by a single neuron, showing that it varies from the region of maximal slope (the 'flanks of the tuning curve) or the region of maximal response (the peak), depending upon the population size and the signal-to-noise ratio [19, 91]. Few works considered how heterogeneity of the tuning parameters affects the coding properties of the neural population [89, 75, 34]. Finally, recent papers showed how such an homogeneous population can be warped to optimally encode stimuli with a non uniform prior distribution [86, 38, 93]. In this paper, we followed this line of works in examining the optimal tuning width as a function of limitations in downstream areas. This has the merit of applying the efficient coding hypothesis in *deeper* neurons, which usually show complex and heterogeneous tuning properties.

With such a complex, multi-peaked shape of tuning curves, the activity of a single representation neuron is not very informative about the stimulus; rather, the neural population, as a whole, is the relevant unit of computation [72]. In this sense, many coding properties can be derived by analyzing the geometry of the neural responses as stimuli parameters are varied (Fig. ??A,C, Fig. ??B), an approach which has been proved to bring fruitful insights in different brain areas [35, 36, 81, 55]. In our setting, by tuning σ, we effectively vary the *intrinsic dimensionality* of the coding manifold, defined as the minimum number of coordinates needed to describe it, interpolating between $\sim N$ in case of random uncorrelated responses and ~ 1 in case of very smooth manifolds. In close relation to our results, the work of [81] suggests that the manifold evoked by the joint neural activity of neurons in V1 possesses a fractal-like structure, with progressively less coding resources employed to encode finer details of the stimulus. Such an arrangement was suggested to balance the accuracy, given by fine scale irregularities, and the noise-robustness, given by smoother manifolds. We gave new insights on this idea, by quantifying how the balance between the two instances depends on the number of neural resources and the magnitude of the noise. The kind of manifolds we considered belong to a class defined in [57, 40] as *random Gaussian manifolds.* These are of theoretical interest, because they saturate the upper bound on the intrinsic dimensionality, given the smoothness imposed by biological constraints. (Which can be measured experimentally, as the autocorrelation length of responses variability in function of stimulus parameters). Thanks to this property, they can be used as null model to quantify the dimensionality of neural trajectories in experimental data. Our results can be used as a benchmark to be compared with recorded neural populations; we gave an example of this approach by re-analyzing the data from [58].

Combinatorial codes and randomness. At the optimal network configuration, the error decreases exponentially fast with the number of neurons, similarly to observations on the coding of position by grid cells [32, 78, 65, 85]. Using the terminology of these works, the random coding scheme of neurons in the second layer is an another example of *exponentially strong* population code. This result is tightly related to ideas that were explored already by Claude Shannon [76].He proposed a geometrical representation of an abstract communication system, as a map which associates points in the space of *messages* (corresponding to stimuli in our case) to points in the space of *signals* (neural activity). The decoding process, which a *receiver* is supposed to do, corresponds to the inverse mapping from signals to messages. By using the one-dimensional message space case as example, he noticed that, in order for such a map to be as efficient as possible, the corresponding one-dimensional curve should wander back and forth through the high dimensional space of signals, to be as long as possible. In this way, the region of uncertainty created by the noise will be small relatively to the length of the line, leading to a higher dynamic range. Nevertheless, this 'signal space filling' map has to be such that the noise does not create large scale ambiguities in the represented message (what he called *threshold effect*, corresponding to global errors in our case). Astonishingly, he showed that this map, which achieves the maximal transmission capacity, need not to be carefully designed, and that optimality can be achieved through random associations between messages and corresponding signals.

The existence in the brain of distributed codes with high (exponential) capacity, and without any evident structure, has been showed in the context of discrete stimuli [1]. For example, neural populations in the cortex exhibit great diversity in their responses to face stimuli, and this allows a population of N neurons to encode exponentially many faces. Here, we extended these ideas to continuous stimuli. The treatment of continuous stimuli introduces a notion of magnitude of errors, not present in the context of discrete ones, where the task is simply to discriminate between two different stimuli. This gives rise to the trade-off between local and global errors, constraining the smoothness of the random code.

Compression and expansion in neural systems. Random *divergent* connectivity have been used as a benchmark model to study the *expansion* of low-dimensional, *dense* neural patterns, into high-dimensional, *sparse* representations [6, 5, 60, 64]. This process features neurons with the so called *mixed selectivity* [70, 35]. The resulting representations of input patterns facilitates the readout and the associative learning in downstream areas [61], and it has been suggested to play a role in the flexibility of working memory [15]. Such divergent pathways and mixed selectivity have been observed in many sensory and cortical regions, e.g., prefrontal cortex, cerebellum, insects' mushroom body and hippocampus [11, 22]

On the other side, in as many cases neural systems exhibit *convergent* pathways, or bottlenecks, where the information encoded in a large population is compressed into a lower number of neurons [39]. In signal processing, techniques for acquiring high-dimensional, sparse signals with a small number of measurements goes under the name of Compressed Sensing (CS) [29]. One of the key results in this field is that, given a high (L)-dimensional signal, which is K-sparse in some basis (meaning that it is possible to express it as a vector with only K components different from 0), it is possible to reconstruct it using a minimal number of noisy 'measurements' (linear projections) which scales only logarithmically with the dimensionality, $N > O\left(K \log\left(L/K\right)\right)$. Notably, the acquisition matrix need not to be carefully designed, as random matrices achieve optimality [20, 7, 8]. In neuroscience, the framework has been successfully applied to model the coding properties of neurons in the olfactory pathway: these sensory neurons do not show any evident structure of selectivity to odors, which can be considered as sparse combinations of molecules [80, 95, 73, 68].

The analogy with our setting is clear, as we considered a low-dimensional stimulus x, encoded in the high-dimensional activity of L neurons, then projected onto N neurons. Indeed, by inverting the Eq.(5) to compute the minimal number of random projections, N, such that it is possible to decode the L stimuli with a given error probability, we obtain that this number grows only logarithmically with the number of stimuli. The difference is that the CS task is to reconstruct the high-dimensional vector, while we are not interested in the reconstruction of the pattern of activity of the first layer, but rather in obtaining an estimate of the low dimensional variable which evoked it. Although in many cases the connectivity of a neural circuit is deeply linked to the underlying functions [54, 62, 31], the brain is a complex system and it is plausible that exhibits some 'unstructured' components. These observations, together with the computational properties of random matrices, lead to consider random convergent synapses as a benchmark for the study of compressed neural representations.

Efficient coding criteria and decoders. In order to compute the optimal coding properties of the network, we used the error in the stimulus estimate as obtained from an ideal decoder. The use of this loss function is justified in information theory [23] and neuroscience [71, 25]. Due to the difficulty in treating analytically the MSE, several studies used the Fisher information as a proxy. According to the Cramer-Rao inequality, this quantity sets a lower bound to the variance of an unbiased estimator. Furthermore, it is related to other interesting measures, such as the mutual information [17, 87, 47], which was the loss function originally assumed in Barlow's seminal work [9]. Nevertheless, Fisher information is a local quantity and therefore fails in keeping track of global errors; our results show a relevant example where its use leads to wrong conclusions about the optimal coding parameters [12, 90, 10].

In our work, a strong assumption is made about the decoder. The ideal decoder is implementable as a two-layer neural network (see Methods): a first layer computes a discrete approximation of the posterior distribution over stimuli and the second one computes an average, returning therefore the Minimal MSE estimator. A similar network decoder has been used by [37]; for the similarity with the classical population vector [43], it has been called *Bayesian* population vector. (Since, instead of weighting the preferred stimuli through the relative neural responses, it uses the correct posterior probability). All the required operations, linear filtering, non linearity and normalization, have been assumed as canonical computations in neural circuits [26, 56, 21]. Nevertheless, the ideal parameters of the decoder (i.e., the synaptic weights) depend on the knowledge of the mean neural responses (tuning curves) and noise variance. It is not clear if these ideal parameters can be learnt with simple, biologically plausible, rules, and how global errors and small scale irregularities affect the learning process. In close relation to these considerations, [14] showed how deep networks trained with gradient descent fit firstly low components of the target function, suggesting how irregularities are learnt much slowly, and require an higher number of examples. The impact of limitations in the decoding architecture on the optimal encoding parameters is an interesting question, which we leaves for future research.

4 Methods

Throughout the paper, 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 [13].

Model description: one-dimensional stimulus

Random Feedforward Network. We considered a two layer architecture. A one-dimensional stimulus, x is encoded by a sensory layer of L neurons, indexed by j, with Gaussian tuning curves centered on a preferred stimulus c_j. This layer projects onto a layer of N neurons ($N < L$) with normally distributed random weights: [1]

$$\begin{aligned} u_j(x) &= A \exp\left(-\frac{(x-c_j)^2}{2\sigma^2}\right) \qquad j = 1, ..., L, \\ v_i(x) &= \sum_{j=1}^{L} W_{ij} u_j(x) \qquad i = 1, ..., N, \\ W_{ij} &\sim \mathcal{N}(0, \frac{1}{L}). \end{aligned} \tag{12}$$

Without loss of generality, we can restrict the stimulus space to be $x \in [0,1]$. In this way, the ‘dynamic range’ , defined as the ratio between the range of represented stimuli and the accuracy, is simply the inverse of the error. We considered an uniform prior on the stimuli and an uniform arrangement of neurons’ preferred positions in the stimulus space, $c_j = j/L$.

Constraint on neural resources. A is a normalization constant; for different widths, we put a constraint on the variance of responses of second layer neurons across all stimuli, R. For each neuron, this quantity depends

[1] In the following, all the computations and simulations are done for a one-dimensional linear stimulus encoded by Gaussian tuning curves. At the same time, we will often assume translational invariance; this will unavoidably introduce edge effects. A more rigorous way would be to consider a circular stimulus and von Mises tuning curves in the first layer:

$$u_j(x) = Aexp\left(\mathcal{K}cos\left(2\pi\left(x - c_j\right)\right)\right).$$

This complicates the form of the correlation function and it would require a modification of the error function. Anyway, we considered regimes where $\mathcal{K}$ is large and the von Mises function can be approximated locally as a Gaussian with width $\sigma^2 = 1/\mathcal{K}$. In the regimes of σ considered in the simulations, and considering that L is very large, the edge effects are small and simulations with circular stimulus did not change qualitatively the results. In Fig. **??**B only, for visualization purposes, we plotted the result for a circular stimulus.

from the specific realization of the synaptic weights, therefore we imposed the constraint on average. Namely:

$$\begin{aligned} R &= \left\langle \left[v_i\left(x\right) - \int_0^1 dx' v_i\left(x'\right) \right]^2 \right\rangle \\ &= \left\langle \int_0^1 dx \left[\sum_j W_{ij} u_j(x) - \left(\int_0^1 dx \sum_j W_{ij} u_j(x) \right)^2 \right]^2 \right\rangle_W \\ &= \left\langle \sum_{jj'} W_{ij} W_{ij'} \int_0^1 dx u_j(x) u_{j'}(x) \right\rangle_W + \left\langle \sum_{jj'} W_{ij} W_{ij'} \int_0^1 dx u_j(x) \int_0^1 dx u_{j'}(x) \right\rangle_W . \end{aligned} \tag{13}$$

where $\langle \dots \rangle_W$ denotes the mean over the synaptic weights. We used the approximation $\int_0^1 dx \exp\left(-\frac{(x-c_j)^2}{2\sigma^2}\right) \approx \sqrt{2\pi\sigma^2}$, which is valid if c_j is sufficiently far from the borders and σ is small (this introduce some edge effects, negligible in the regime of σ and L we considered). Using also the fact that weights are statistically independents and have zero mean, $\langle W_{ij} W_{ij'} \rangle = \frac{1}{L}\delta_{jj'}$, we obtain the equation for A :

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

In the following, we will often use an approximation for small widths: $A^{-2} \approx \frac{\sqrt{\pi\sigma^2}}{R}$. Finally, note that since we employed a linear projection, we could have simply re-scaled the variance of the synaptic weights, but we preferred to keep separated this two contributions.

Gaussian Processes analogy. If we assume that the spacing between preferred positions is small, we can approximate the sum in Eq.(12) with a convolution integral of a random noise process (the synaptic weights) with a smoothing kernel (the tuning curves of first layer neurons)[2]:

$$v_i(x) = \sum_{j=1}^{L} W_{ij} u_j(x) \approx L \int_0^1 dc_j u(c_j - x) W_i(c_j).$$

This gives rise to a Gaussian process, as described in [46, 69]. Computing the covariance function is straightforward:

$$\begin{aligned} \langle v_i(x) v_i(x') \rangle_i = \left\langle \sum_j W_{ij} W_{ij'} u_j(x) u_{j'}(x') \right\rangle_i &= A^2 \sum_j \frac{1}{L} \delta_{jj'} \exp\left(-\frac{(x-c_j)^2}{2\sigma^2}\right) \exp\left(-\frac{(x'-c_{j'})^2}{2\sigma^2}\right) \\ &\approx A^2 \int dc_j \exp\left(-\frac{(x-c_j)^2 + (x'-c_j)^2}{2\sigma^2}\right). \end{aligned} \tag{15}$$

Assuming translational invariance, we obtain that the tuning curves are samples from a one-dimensional Gaussian process with 0 mean and Gaussian kernel with correlation length $\sqrt{2}\sigma$:

$$\begin{aligned} \langle v_i(x) \rangle &= 0 \\ K(x, x') = \langle v_i(x) v_i(x + \Delta x) \rangle &\approx A^2 \sqrt{\pi\sigma^2} \exp - \left(\frac{\Delta x^2}{4\sigma^2}\right). \end{aligned} \tag{16}$$

This network maps the one-dimensional stimulus space $[0, 1]$ onto a manifold embedded in the N dimensional space of neurons' activity. The coordinates of this manifold are described by N independents samples of the aforementioned Gaussian process. This corresponds to the definition of "Random Gaussian Manifold" proposed in [57, 40].

[2] This is a delicate integral to treat, as it is not properly defined. One should pass through Ito integration and Ito isometry properties to define this object rigourously.

Coding - decoding process

Noise Model. We considered an isotropic Gaussian model for the noise affecting second layer neurons. At each trial, the vector of responses to a given stimulus x is given by

$$\mathbf{r} = \mathbf{v}(x) + \mathbf{z}, \tag{17}$$

where $\mathbf{z}$ is a noise vector of independent Gaussian entries with a fixed variance, $\mathbf{z} \sim \mathcal{N}(0, \eta^2 \mathbf{I})$. The likelihood of a response vector given a stimulus, for a fixed realization of the synaptic weights, can be written as

$$p\left(\mathbf{r}|x\right) = \frac{1}{\left(2\pi\eta^2\right)^{N/2}} \exp\left(-\frac{\|\mathbf{r}-\mathbf{v}(x)\|_2^2}{2\eta^2}\right). \tag{18}$$

The error will be governed by the signal-to-noise ratio ($SNR = \frac{R}{\eta^2}$); in numerical simulations we set the variance of the responses $R = 1$ and we varied η^2 to explore different noise regimes. The noise model can be extended to include correlations in the noise affecting different neurons. Denoting with Σ the full noise covariance matrix, the likelihood of the neural response in second layer neurons can be written as a multivariate gaussian distribution,

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

Loss function and decoder. We used the Mean Squared Error (MSE) in stimulus estimate as loss function to measure the coding properties of the neural population. An estimator (or decoder), $\hat{x}(\mathbf{r})$, is a function that takes in input a noisy response and output an estimate of the stimulus that evoked it. Given a decoder, the MSE is defined as

$$E^2 = \int dx \int d\mathbf{r} p(\mathbf{r}|x)\left(\hat{x}\left(\mathbf{r}\right) - x\right)^2. \tag{20}$$

We considered this quantity averaged over network realizations, $\varepsilon^2 = \langle E^2 \rangle_W$; in order to show a quantity which has the same measurements units of the stimulus, we often plotted the Root-MSE $\varepsilon = \sqrt{\langle E^2 \rangle_W}$. This quantity is generally hard to compute, even knowing a closed form for the estimator. In numerical simulations we computed this integral with standard Monte Carlo method. At each step we extracted a set of L stimuli (one for each preferred position of the first layer neurons) from the uniform distribution and we extracted samples from the response distribution. We then passed the noisy responses through an ideal decoder (see below) and we updated the error estimate. We iterated this process until when the MSE estimate was within a tolerance of 10^{-7} in the last 50 steps, after 100 steps of relaxation.

The estimator which minimizes the MSE is called Minimal Mean Squared Error estimator (MMSE), and it is given by the average of the posterior distribution. Using Bayes theorem, we obtain that the posterior is simply proportional to the likelihood, due to the choice of uniform prior, and the MMSE estimator can be written as

$$\hat{x}_{MMSE} = \int_0^1 dx p(x|\mathbf{r})x = \frac{\displaystyle\int_0^1 dx x p(\mathbf{r}|x)}{\displaystyle\int_0^1 dx p(\mathbf{r}|x)}. \tag{21}$$

This function can be approximated by a simple neural network. Discretizing the stimulus space in M values, $x_m = \frac{m}{M}$, and substituting the expression for the likelihood, Eq.(18), we can approximate the integrals as discrete sums

$$\begin{aligned}\hat{x} &\approx \frac{\sum_m x_m p\left(\mathbf{r}|x_m\right)}{\sum_m p\left(\mathbf{r}|x_m\right)} = \frac{\sum_m x_m \exp\left(-\frac{1}{2\eta^2}\sum_i r_i^2 + v_i^2(x_m) - 2v_i(x_m)r_i\right)}{\sum_m \exp\left(-\frac{1}{2\eta^2}\sum_i r_i^2 + v_i^2(x_m) - 2v_i(x_m)r_i\right)} \\ &= \frac{\sum_m x_m \exp\left(\frac{1}{2\eta^2}\sum_i 2v_i(x_m)r_i - v_i^2(x_m)\right)}{\sum_m \exp\left(\frac{1}{2\eta^2}\sum_i 2v_i(x_m)r_i - v_i^2(x_m)\right)}.\end{aligned} \tag{22}$$

where we removed $\sum_i r_i^2$, common to both numerator and denominator. A layer of M neurons can compute the likelihood function for different stimuli x_m. Calling λ the connectivity matrix between the N neurons of the output layer and the M neurons of the decoder, its entries are proportional to the true responses to the preferred stimulus of the decoder neurons: $\lambda_{mi} = v_i(x_m)/\eta^2$. The sum is passed through an exponential non linearity with the addition of a bias term $b_m = \sum_i v_i(x_m)^2/2\eta^2$, to obtain the output of a single neuron $h_m = \exp\left(\sum_i \lambda_{mi} r_i - b_m\right)$. This layer could implement a winner-take-all dynamic to output the maximum a posteriori (MAP) estimator:

$$\hat{x} = \arg\min_x \|\mathbf{r} - \mathbf{v}(x)\|_2^2 = \arg\max_{x_m} h_m. \tag{23}$$

Alternatively, the output of each neuron can be weighted according to its preferred stimulus (with the addition of a divisive normalization) to obtain the MMSE estimator

$$\hat{x} = \frac{\sum_m x_m h_m}{\sum_m h_m}. \tag{24}$$

In numerical simulations, we adopted for the decoder the same discretization of the stimulus of the first layer, using $M = L$ and spacing uniformly the preferred stimuli x_m. Note that the decoder is ideal, since it is assumed to know the true responses and the variance of the noise.

The same decoder can be extended to deal with the case of non-diagonal noise covariance matrix Σ, with the difference that the decoding weights and biases are now correlated: $\lambda_m = \mathbf{v}^T(x_m)\Sigma^{-1}$, $b_m = \mathbf{v}^T(x_m)\Sigma^{-1}\mathbf{v}(x_m)$, where λ_m denotes the m-th row of λ. In order to estimate the scaling of the error, in the following sections we will often use the MAP estimator, since it has an easier geometrical interpretation (minimal distance). In the main text we showed results for the optimal decoder (MMSE), but the performances for the two are very similar.

Errors' computation

Narrow tuning curves. If $\sigma \to 0$, the first layer neurons respond only to their preferred stimulus. For this extreme case, we supposed that the stimulus can assume only L discrete values, $x_j = j/L$. The responses of the second layer neurons are given by $v_i(x_j) = W_{ij}$, with $W_{ij} \sim \mathcal{N}(0, R)$, and are uncorrelated for different stimuli. Let's denote with $p_e(\mathbf{r}|x) = p(\mathbf{r}|x)\Theta\left(|\hat{x} - x|\right)$ the (conditioned) probability density function that the noise will produce an error, where we introduced the Heaviside function $\Theta(x) = 1$ only if $x > 0$ (and 0 otherwise). We notice that, taking the average over the synaptic weights, the magnitude of an error is independent from its probability and no more depends on the specific realization of the noise $\mathbf{r}$. The average MSE can be rewritten as

$$\begin{aligned} \langle E^2 \rangle_W &\approx \frac{1}{L} \sum_x \left\langle \int d\mathbf{r} p_e(\mathbf{r}|x) \left(\hat{x}(\mathbf{r}) - x\right)^2 \right\rangle_W \\ &= \langle P(E) \rangle_W \left\langle \frac{1}{L} \sum_x (\hat{x} - x)^2 \right\rangle_W , \end{aligned} \tag{25}$$

where $\langle P(E) \rangle_W = \left\langle \int d\mathbf{r} p_e(\mathbf{r}|x) \right\rangle_W$ 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. This formula has an intuitive interpretation: the average MSE is the mean probability of having an error on a stimulus multiplied by the mean error magnitude. Let's suppose now to estimate the stimulus through a ML decoder, Eq.(23): we will obtain an error if there exists at least one x' such that $\|\mathbf{r} - \mathbf{v}(x')\|_2^2 < \|\mathbf{r} - \mathbf{v}(x)\|_2^2$. Since, averaging over the synaptic weights, all x' have the same probability to cause such an error, the average size of the squared error will be

$$\left\langle \frac{1}{L} \sum_x (\hat{x} - x)^2 \right\rangle_W = \frac{1}{L^2} \sum_{j=1}^{L} \sum_{j'=1}^{L} \left(\frac{j'}{L} - \frac{j}{L} \right)^2 \approx \frac{1}{6}, \tag{26}$$

where the last approximation holds for large L. Let's now compute the average probability of error; this quantity can be expressed in terms of the probability of the complementary event

$$\langle P(E) \rangle_W = 1 - \left\langle P\left(\|\mathbf{r} - \mathbf{v}(x')\|_2^2 > \|\mathbf{r} - \mathbf{v}(x)\|_2^2 \quad \forall x' \neq x \right) \right\rangle_W . \tag{27}$$

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

$$
\begin{aligned}
\langle P(E)\rangle_W &= 1-\Big(1-\Big\langle P\left(\|\mathbf{r}-\mathbf{v}(x')\|_2^2 < \|\mathbf{r}-\mathbf{v}(x)\|_2^2\right)\Big\rangle_W\Big)^{L-1} \\
&\approx L\left\langle P\left(\sum_i \big(v_i(x')-v_i(x)\big)^2 + z_i^2 - 2\big(v_i(x)-v_i(x')\big)z_i < \sum_i z_i^2\right)\right\rangle_W ,
\end{aligned}
\tag{28}
$$

where we explicitly substituted Eq.(17), we supposed that the average probability of having an error is small (much smaller than $1/L$), and we considered $L-1 \approx L$. The average difference between the response of the same neuron to two different stimuli is normally distributed $\tilde{v}_i = v_i(x) - v_i(x') = W_{ij} - W_{ij'} \sim \mathcal{N}(0, 2R)$. Averaging also over the noise distribution, we obtain

$$
\langle P(E)\rangle_W \approx L \int \prod_i d\tilde{v}_i \prod_i dz_i p(\tilde{v}_i) p(z_i) \Theta\left(-\sum_i \tilde{v}_i^2 + 2\sum_i \tilde{v}_i z_i\right). \tag{29}
$$

In words, we have to compute 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 $z_i \sim \mathcal{N}(0, \eta^2)$. Fixing $\zeta = \sum_i \tilde{v}_i^2$, the conditional distribution $\rho|\{\tilde{v}_i^2\} \sim \mathcal{N}(\zeta, 4\zeta\eta^2)$ is gaussian. Therefore, using the definition of error function, we can rewrite the error probability as

$$
\begin{aligned}
\langle P(E)\rangle_W &\approx L \int_0^\infty d\zeta p(\zeta) \int_{-\infty}^0 d\rho p(\rho|\zeta) \\
&= \frac{L}{2} \int_0^\infty d\zeta p(\zeta) \operatorname{erfc}\left(\sqrt{\frac{\zeta}{8\eta^2}}\right),
\end{aligned}
\tag{30}
$$

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

$$
\begin{aligned}
\langle P(E)\rangle_W &\approx L \frac{(\frac{\eta^2}{2R})^{\frac{N}{2}}\Gamma(N)}{\Gamma(\frac{N}{2})} {}_2\tilde{F}_1\left(\frac{N}{2}, \frac{1+N}{2}, \frac{2+N}{2}, -2\frac{\eta^2}{R}\right) \\
&= L \frac{(\frac{\eta^2}{2R})^{\frac{N}{2}}\Gamma(N)}{\Gamma(\frac{N}{2})\Gamma(\frac{2+N}{2})} \sum_{n=0}^{\infty} \frac{(\frac{N}{2})_n(\frac{N+1}{2})_n}{(\frac{N+2}{2})_n n!}(-2\frac{\eta^2}{R})^n,
\end{aligned}
\tag{31}
$$

where ${}_2\tilde{F}_1(a, b, c, x)$ is the regularized 2F1 Hypergeometric function and 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

$$
\begin{aligned}
\langle P(E)\rangle_W &\approx L(\frac{\eta^2}{2R})^{\frac{N}{2}} \frac{\Gamma(N)}{\Gamma^2(\frac{N}{2})\frac{N}{2}(1+2\eta^2/R)^{\frac{N+1}{2}}} \\
&\approx \frac{L}{\sqrt{2\pi N}} \exp\left(-\log\left(1+\frac{R}{2\eta^2}\right)\frac{N}{2}\right),
\end{aligned}
\tag{32}
$$

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

Broad tuning curves. As soon as $\sigma > 0$, we allow for continuous stimuli and the resulting manifold in the activity space is smooth. In this case, the noise can also produce small scale errors: we therefore split the error in two contributions, local and global. Since our system has a natural correlation length, we defined as global an error when the difference between the stimulus and its estimate is greater than σ : $|\hat{x}(\mathbf{r}) - x| > \sigma$. This definition is a bit tricky, since for very large σ all the errors will be local. Anyway, we are interested in the case where σ is relatively small, and what matters is that global errors are of the order of the size of the stimulus space. We rewrite the average error as

$$
\varepsilon^2 = \langle E^2\rangle_W = \langle E_l^2 + E_g^2\rangle_W = \left\langle \int dx d\mathbf{r} p_l(\mathbf{r}|x)\left(\hat{x}(\mathbf{r}) - x\right)^2 \right\rangle_W + \left\langle \int dx d\mathbf{r} p_g(\mathbf{r}|x)\left(\hat{x}(\mathbf{r}) - x\right)^2 \right\rangle_W , \tag{33}
$$

where with $p_{l/g}(\mathbf{r}|x) = p(\mathbf{r}|x)\Theta\big(\pm(\sigma - |\hat{x}(\mathbf{r}) - x|)\big)$ we denoted 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. A ML decoder will output the stimulus corresponding to the closest point of the manifold, which in case of local error will correspond to the projection of the noise vector onto the manifold. Expanding linearly the response around x, we obtain

$$\left\| \mathbf{r} \cdot \hat{\mathbf{v}'}(x) \right\|_2^2 = \|\mathbf{v}(x + \Delta x) - \mathbf{v}(x)\|_2^2 \approx \|\mathbf{v}'(x)\|_2^2 \, \Delta x^2, \tag{34}$$

where $\hat{\mathbf{v}'}(x)$ is the normalized vector in the direction of the derivative of the tuning curves. The resulting error will be $\Delta x^2 = (\hat{x}(\mathbf{r}) - x)^2 = \frac{\left\|\mathbf{r}\cdot\hat{\mathbf{v}'}(x)\right\|_2^2}{\|\mathbf{v}'(x)\|_2^2}$. 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 likelihood function, Eq.(18). Since the noise is isotropic, when integrating over it the average magnitude of the projection onto a fixed unit vector will be simply the variance, and we can write the local error as

$$\langle E_l^2 \rangle_W = \left\langle \int dx \frac{\eta^2}{\|\mathbf{v}'(x)\|_2^2} \right\rangle_W . \tag{35}$$

Computing the derivative of the tuning curves we obtain

$$\begin{aligned} \|\mathbf{v}'(x)\|_2^2 &= A^2 \sum_i \sum_{jj'} W_{ij} W_{ij'} \frac{(x - c_j)(x - c_{j'})}{\sigma^4} \exp\left(- \frac{(x - c_j)^2 + (x - c_{j'})^2}{2\sigma^2} \right) \\ &\approx \frac{A^2 \sum_j \exp\left(- \frac{(x - c_j)^2}{\sigma^2} \right)}{\sigma^4} \approx \frac{A^2 N \sqrt{\pi\sigma^2}}{2\sigma^2}, \end{aligned} \tag{36}$$

where we took the average over the weights $\langle \sum_{i=1}^N W_{ij} W_{ij'} \rangle_W = \frac{N}{L} \delta_{jj'}$ [3] and we substituted the sum with an integral $\sum_j \approx \frac{1}{L} \int dc_j$. Considering the limit of small σ for A^2, we finally obtain the expression

$$\varepsilon_l^2 = \langle E_l^2 \rangle_W \approx \frac{2\sigma^2 \eta^2}{RN}. \tag{37}$$

Note that this quantity corresponds to the inverse of the linear FI, as predicted by the Cramer-Rao bound.

Global error. We defined an error as global when the estimate of the stimulus is further than σ from the true value. In this case, we can make the same reasoning of the uncorrelated case, noticing that once we obtain an error of this kind, its average magnitude is independent from its probability and independent from the noise magnitude $\mathbf{r}$. Therefore we can write, similarly to Eq.(25), the expression for the global error

$$\langle E_g^2 \rangle_W = \langle P(E) \rangle_W \left\langle \int dx (\hat{x} - x)^2 \right\rangle_W . \tag{38}$$

We can assume that in such a case the estimate will be uniformly distributed in the interval $\hat{x} \notin [x - \sigma, x + \sigma]$, and obtain for the average magnitude of global error

$$\bar{\varepsilon}_g = \int dx \int d\hat{x} p(\hat{x}) \, (\hat{x} - x)^2 \approx \frac{1}{6} + O(\sigma), \tag{39}$$

where we underlined the fact that is a term of order 1 plus corrections of order σ. Finally, we have to compute the probability that, given a stimulus x, the error will be global. This quantity again will not depend from the specific choice of the stimulus. Computing this probability rigorously is hard, due to the correlations between nearby responses. Nevertheless, we know that at for stimuli at a distance of $> \sigma$ the two responses are uncorrelated, Eq.(16). We can therefore imagine to divide the manifold into $\frac{1}{\sigma}$ discrete correlated regions (segments) of responses: we will have a global error when the estimate of the stimulus belong to a segment other than the true response. We computed the probability of having an error with uncorrelated responses in the previous section, Eq.(32). We simply have to substitute to L the actual number of uncorrelated clusters $\frac{1}{\sigma}$, obtaining for the global error

$$\varepsilon_g^2 = \langle E_g^2 \rangle_W \approx \frac{1}{\sigma\sqrt{2\pi N}} \bar{\varepsilon}_g \exp\left(- \log\left(1 + \frac{R}{2\eta^2}\right) \frac{N}{2}) \right). \tag{40}$$

[3] Note that we are approximating the average of the inverse with the inverse of the average, but as soon as N is not too small these two quantities are very similar.

Input noise. We considered 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 $\mathbf{z}^{\mathbf{u}} \sim \mathcal{N}(0, \xi^2 \mathbf{I})$. This results in a multivariate Gaussian distribution for the responses of the second layer, Eq.(19), with covariance matrix $\Sigma = \eta^2 \mathbf{I} + \xi^2 \mathbf{W}\mathbf{W}^T$. The matrix $\mathbf{W}\mathbf{W}^T$ follows the well known Wishart distribution [63], with mean $\mathbf{I}$ and fluctuations of the terms of order $1/L$. Therefore the covariance matrix can be rewritten as the sum of the identity plus a perturbation

$$\Sigma = \tilde{\eta}^2 \mathbf{I} + \xi^2 (\mathbf{W}\mathbf{W}^{\mathbf{T}} - \mathbf{I}), \tag{41}$$

where we introduced an effective noise variance, which is the sum of input and output noise variance $\tilde{\eta}^2 = \eta^2 + \xi^2$. In order to obtain an estimate of the effects of input noise on the local error, we consider the Fisher Information (FI) as a lower bound to the MSE; the linear FI is computed as

$$J(x) = \mathbf{v}'(x)^T \Sigma^{-1} \mathbf{v}'(x), \tag{42}$$

where, again, $\mathbf{v}'(x)$ denotes the derivative of the tuning curves with respect to the stimulus variable. If the perturbation is small, we can approximate the inverse of the correlation matrix at the second order $\Sigma^{-1} \approx \frac{1}{\tilde{\eta}^2} \mathbf{I} - \frac{\xi^2}{\tilde{\eta}^4} \left(\mathbf{W}\mathbf{W}^T - I \right) + \frac{\xi^4}{\tilde{\eta}^6} \left(\mathbf{W}\mathbf{W}^{\mathbf{T}} - \mathbf{I} \right)^2$, and write the FI as:

$$\begin{aligned} J(x) &= J^{ind}(x) - \delta J(x) \\ &= \frac{\sum_i v_i'^2(x)}{\tilde{\eta}^2} - \frac{\xi^2}{\tilde{\eta}^4} \mathbf{u}'^T(x) \left(\mathbf{A}^2 - \mathbf{A} \right) \mathbf{u}'(x) + \frac{\xi^4}{\tilde{\eta}^6} \mathbf{u}'^T(x) \left(\mathbf{A}^3 - 2\mathbf{A}^2 + \mathbf{A} \right) \mathbf{u}'(x), \end{aligned} \tag{43}$$

where $\mathbf{A} = \mathbf{W}^{\mathbf{T}}\mathbf{W}$ and we used the matrix notation $\mathbf{v}(x) = \mathbf{W}\mathbf{u}(x)$. We recognize in the first term,$J^{ind}(x)$, the FI in the case of output noise only with effective variance $\tilde{\eta}^2$. All the correction terms to the FI are related to the moments of the matrix $\mathbf{A} = \mathbf{W}^{\mathbf{T}}\mathbf{W}$. Since all the entries are Gaussian, it is possible to compute their mean through Isserlis' Wick? theorem. Using the fact that $\langle W_{ij} W_{mn} \rangle = \frac{1}{L} \delta_{im} \delta_{jn}$, we obtain:

$$\begin{aligned} \langle A_{mn} \rangle &= \left\langle \sum_{j=1}^{N} W_{jm} W_{jn} \right\rangle = \frac{N}{L} \delta_{mn} \\ \langle A^2_{mn} \rangle &= \left\langle \sum_{i=1}^{L} \sum_{j=1,j'=1}^{N} W_{jm} W_{ji} W_{j'i} W_{j'n} \right\rangle = \left\langle \frac{N}{L} + \frac{N^2}{L^2} + \frac{N}{L^2} \right) \delta_{mn} \\ \langle A^3_{mn} \rangle &= \left\langle \sum_{i=1,i'=1}^{L} \sum_{j=1,j'=1,j''=1}^{N} W_{jm} W_{ji} W_{j'i} W_{j'i'} W_{j''i'} W_{j''n} \right\rangle = \left(\frac{N^3}{L^3} + 3\frac{N^2}{L^3} + 3\frac{N^2}{L^2} + 4\frac{N}{L^3} + 3\frac{N}{L^2} + \frac{N}{L} \right) \delta_{mn} \end{aligned} \tag{44}$$

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

$$\langle \delta J(x) \rangle_W = \frac{N^2}{L^2} \frac{\xi^2}{\tilde{\eta}^4} \mathbf{u}'(x)^T \mathbf{I} \mathbf{u}'(x) - \frac{N^2}{L^2} \frac{\xi^4}{\tilde{\eta}^6} \mathbf{u}'(x)^T \mathbf{I} \mathbf{u}'(x). \tag{45}$$

In computing $\mathbf{u}'(x)^T \mathbf{I} \mathbf{u}'(x) = \sum_j u_j'(x)^2$ we may substitute the discrete sum with an integral, similarly to what we have done in Eq.(36), and after regrouping the terms we obtain the mean Fisher Information as

$$\langle J(x) \rangle_W \approx \frac{A^2 N \sqrt{\pi \sigma^2}}{2\sigma^2 \tilde{\eta}^2} (1 - \frac{N}{L} \frac{\xi^2}{\tilde{\eta}^2} + \frac{N}{L} \frac{\xi^4}{\tilde{\eta}^4}), \tag{46}$$

and consequently an approximation to the MSE

$$\varepsilon_l^2 = \langle E^2 \rangle_W \approx \frac{1}{\langle J(x) \rangle_W} \approx \varepsilon_{l,i}^2 (1 + \frac{N}{L} \frac{\xi^2}{\tilde{\eta}^2} - \frac{N}{L} \frac{\xi^4}{\tilde{\eta}^4}). \tag{47}$$

Similar computations can be done assuming a covariance matrix with the same statistic, but not related to the synaptic weights. For example, assuming $\Sigma_{rand} = \eta^2 I + \xi^2 \mathbf{X}\mathbf{X}^T$ with $X_{ij} \sim \mathcal{N}(0, \frac{1}{L})$ similarly to W, but with uncorrelated entries $\langle X_{ij} W_{mn} \rangle_{W,X} = 0$. In this case we have no more first order corrections, and the Fisher Information corrections are always positive

$$\langle J(x) \rangle_{W,X} \approx \frac{A^2 N \sqrt{\pi \sigma^2}}{2\sigma^2 \tilde{\eta}^2} (1 + \frac{N}{L} \frac{\xi^4}{\tilde{\eta}^4}). \tag{48}$$

Extension to multidimensional stimuli

We consider a stimulus in the hypercube $\mathbf{x} \in [0,1]^K$, and the MS, as the sum of the error across each dimension, $\varepsilon^2 = \sum_{k=1}^K \varepsilon_k^2$. Similarly to the previous case, the local error along each dimension is computed expanding linearly the tuning curves

$$\|\mathbf{v}(\mathbf{x} + \Delta x_k) - \mathbf{v}(\mathbf{x})\|_2^2 \approx \left\| \frac{\partial}{\partial x_k} \mathbf{v}(\mathbf{x}) \right\|_2^2 \Delta x_k^2. \tag{49}$$

We consider as global every error such that $\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2 > \sigma$. We consider the two extreme cases where all neurons in the second layer receive inputs from all neurons in the first one.

Pure case. In the case the first layer is made up by pure cells, neurons are sensitive to only one stimulus dimension. We assumed their tuning curves to be one-dimensional Gaussian functions $u_{j_k}(\mathbf{x}) = A_p \exp\Big(-\frac{(x_k - c_{j_k})^2}{2\sigma^2}\Big)$ with preferred positions uniformly arranged along each dimension, $c_{j_k} = j_k/M$ for $j_k = 1, ..., M$ and $M = L/K$. The second layer tuning curves are given by the linear superposition of uncorrelated Gaussian processes along each dimension $v_i^p(\mathbf{x}) = \sum_k \sum_{j_k} W_{ij_k} u_{j_k}(\mathbf{x})$. Using the same constraint as before, we obtain $A_p^{-2} = \Big(\big(\pi\sigma^2\big)^{1/2} - 2\pi\sigma^2\Big)/R$. In this case each dimension is encoded separately. When computing the local error, the squared norm of the derivative along one dimension is reduced by a factor of K (the derivative along each dimension will act only on $1/K$ terms), and consequently the local error along each dimension is

$$\varepsilon_{l,p,k}^2 = \frac{2K\sigma^2\eta^2}{A_p^2 N (\pi\sigma^2)^{1/2}} \approx \frac{2K\sigma^2\eta^2}{RN}. \tag{50}$$

Also the probability of having a global error is independent along each dimension (we can make a global error independently on each of the K dimensions of the stimulus). We can approximate the total probability of having a global error as the sum of probabilities along each dimension, $P(E_g) = \sum_k P(E_{g,k})$. Since in this case the tuning curves are described by a superposition of uncorrelated Gaussian processes and each dimension contributes equally to the variance, we obtain for the global error in the pure case

$$\varepsilon_{g,p}^2 \approx \frac{K\bar{\varepsilon}_g}{\sigma\sqrt{2\pi N}} \exp\left(-\log\left(1 + \frac{R}{2K\eta^2}\right)\frac{N}{2}\right) \tag{51}$$

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

Conjunctive case. In the conjunctive case the first layer neurons' responses are given by multi-dimensional Gaussian functions $u_j(\mathbf{x}) = A_c \exp\Big(-\frac{\|\mathbf{x} - \mathbf{c}_j\|_2^2}{2\sigma^2}\Big)$ with preferred positions arranged on a K-dimensional square grid of side $1/M$,with $M = L^{1/K}$. The tuning curves of the second layer neurons $v_i^c(\mathbf{x}) = \sum_j W_{ij} u_j(\mathbf{x})$ are multi-dimensional Gaussian processes with K-dimensional covariance function $\langle v(\mathbf{x}) v(\mathbf{x} + \Delta\mathbf{x})\rangle = A_c^2 \exp\Big(-\frac{\|\Delta\mathbf{x}\|_2^2}{2\sigma^2}\Big)$. The normalization term is given by $A_c^{-2} = \Big(\big(\pi\sigma^2\big)^{K/2} - (2\pi\sigma^2)^K\Big)/R$ (note that increasing the dimensionality of the stimulus, the edge effects become more relevant). In this case the derivative along one dimension will act on all the terms of the random sum, and the resulting local error is given by

$$\varepsilon_{l,c,k}^2 = \frac{2\sigma^2\eta^2}{A_c^2 N (\pi\sigma)^{K/2}} \approx \frac{2\sigma^2\eta^2}{RN}. \tag{52}$$

To compute the global error we simply extend the reasoning about uncorrelated regions. Stimuli evoke a correlated response within a radius of $\sim \sigma$, and we approximate the number of uncorrelated clusters as $1/\sigma^K$. The global error is given by

$$\varepsilon_{g,c}^2 \approx \frac{1}{\sigma^K\sqrt{2\pi N}} \bar{\varepsilon}_g \exp\left(-\log\left(1 + \frac{R}{2\eta^2}\right)\frac{N}{2}\right). \tag{53}$$

Data analysis and model fitting

Data description and summary statistics The detailed data description is reported in [58], and data are publicly available at https://osf.io/u57df/. They consists 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, it varies across different neurons and trials). Tuning curves are computed averaging over trials. In order to measure the level of irregularity of one 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

$$c(D_{min})_i = std\Big(\frac{\|v(x) - v(x + \Delta x)\|}{\sqrt{\|\Delta x\|^2}} s.t. \ \|\Delta x\|_2^2 < D_{min}\Big). \tag{54}$$

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),

$$R_i^2 = 1 - \frac{RSS}{TSS} = 1 - \frac{\sum_x (v_l(\mathbf{x}) - v(\mathbf{x}))^2}{\sum_x v(\mathbf{x})^2}, \tag{55}$$

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

Model fitting. We considered the tuning curves just in function of the position, ignoring the difference in hand orientation. We chose to analyze just 'tuned neurons', cells responding with at least 5 spikes/s at more than two positions. We mean-centered and standardized the tuning curves to have variance equal to 1. We generated an irregular population with a Random Feedforward Network with a sensory layer of conjunctive neurons responding to a three-dimensional stimulus. In order to tile the space and avoid boundary effects, we used $M = 100^3$, tiling a 200 by 200 by 200 cube with a grid of side 2 (such that the stimulus space in the experiment is fully included). For the connectivity matrix $\mathbf{W}$ we used a sparse random matrix (for computational purposes, sparsity = 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 [58], 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. Instead, the only tunable parameter of our model is σ. In order to fit the model, we generated the tuning curves, measured at the same 27 stimuli of the experiment, of a number of representation neurons equal to the number of recorded neurons. We then computed the distribution of the complexity measure (in a.u.) for different values of σ and we picked σ_f such that the Kolmogorov-Smirnov (KS) distance between the distribution of the model and the one of the data was minimal (Fig. ??A). At this optimal σ, the two distributions are very similar, even if real data show a broader distribution of values in both directions; for comparison, a linear model suffers an heavy underestimate of the complexity measure across all neurons (Fig. ??B). For the sake of completeness, we computed the KS distance between the model and the data also for the R^2 measure (Fig. ??A, red line). This quantity simply decreases with σ. The model at σ_f underestimate the linear components of the tuning curves (Fig. ??C). Nevertheless, this is expected since our model has no non linearity, which potentially increases the illusion of linear tuning. It is worth noticing that in the original work, the simpler model described 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 more complicate model with more parameters (namely, different thresholds for each neuron and different widths in the first layer).

We also did simulation with a noise model extracted from the data. To each neuron, we assigned a noise variance in the following way. We computed the variance of the signal of a sample neuron as the variance of the responses across all possible stimuli, $\text{Var}(v) = \langle v^2 \rangle_x - \langle v \rangle_x^2$. Then, we computed the mean variance of the trial to trial variability across all stimuli, $\text{Var}(\eta) = \langle \text{Var}(v(x)) \rangle_x$. Since our tuning curves in the simulations have a response range equal to one, we assigned to neuron i a variance of the noise equal to $\eta_i^2 = \frac{\text{Var}(\eta_i)}{\text{Var}(r_i)}$. The decoding error for a population size of N neurons was computed averaging over 8 independent pools of N neurons, each one associated with its noise variance. Also the decoder, Eq.(24), was modified to keep into account each neuron's noise variance. 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 $\sqrt{r(\mathbf{x})}$, [79]). The distribution of the noise variance across neurons obtained in this way does not vary substantially.

For numerical simulations in Fig. ??, the tuning curves were computed at a much 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. ??D-F). The linear population for the comparison was constructed sampling the preferred vectors ((a_1, a_2, a_3)) uniformly on the unit sphere and using Eq.(9) to generate the responses. The tuning curves were shifted and normalized to have vanishing 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] Bernardi, S., Benna, M. K., Rigotti, M., Munuera, J., Fusi, S., & Salzman, C. D. (2020). The Geometry of Abstraction in the Hippocampus and Prefrontal Cortex. Cell, 183, 954–967.

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

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

[14] 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).

[15] Bouchacourt, F. & Buschman, T. J. (2019). A Flexible Model of Working Memory. Neuron, 103, 147–160.

[16] 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.

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

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

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

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

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

[22] Cayco-Gajic, N. A. & Silver, R. A. (2019). Re-evaluating Circuit Mechanisms Underlying Pattern Separation. Neuron, 101.

[23] Cover, T. M. & Thomas, J. A. (2005). Elements of Information Theory. (Wiley), 2nd edn.

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

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

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

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

[28] 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.

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

[30] 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.

[31] Farrell, M., Recanatesi, S., Reid, R. C., Mihalas, S., & Shea-Brown, E. (2020). Autoencoder networks extract latent variables and encode these variables in their connectomes. bioRxiv, p. 2020.03.04.977702.

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

[33] 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.

[34] 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.

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

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

[37] Ganguli, D. & Simoncelli, E. P. (2014). Efficient Sensory Encoding and Bayesian Inference. Neural computation.

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

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

[40] 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.

[41] 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.

[42] 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.

[43] Georgopoulos, A. P., Schwartz, A. B., & Kettner, R. E. (1986). Neuronal population coding of movement direction. Science, 233.

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

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

[46] Higdon, D. (2002). Space and Space-Time Modeling using Process Convolutions. Quantitative Methods for Current Environmental Issues, pp. 37–56.

[47] Huang, W. & Zhang, K. (2019). Approximations of Shannon mutual information for discrete variables with applications to neural population coding. Entropy, 21.

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

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

[50] 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.

[51] 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.

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

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

[54] Kim, J. S., Greene, M. J., Zlateski, A., Lee, K., Richardson, M., Turaga, S. C., Purcaro, M., Balkam, M., Robinson, A., Behabadi, B. F., Campos, M., Denk, W., & Seung, H. S. (2014). Space-time wiring specificity supports direction selectivity in the retina. Nature.

[55] 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.

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

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

[58] 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.

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

[60] Lindsay, G. W., Rigotti, M., Warden, M. R., Miller, E. K., & Fusi, S. (2017). Hebbian learning in a random network captures selectivity properties of the prefrontal cortex. Journal of Neuroscience, 37, 11021–11036.

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

[62] Litwin-Kumar, A. & Turaga, S. C. (2019). Constraining computational models using electron microscopy wiring diagrams.

[63] Livan, G., Novaes, M., & Vivo, P. (2017). Introduction to Random Matrices - Theory and Practice.

[64] Maoz, O., Tkačik, G., Esteki, M. S., Kiani, R., & Schneidman, E. (2020). Learning probabilistic neural representations with randomly connected circuits. Proceedings of the National Academy of Sciences, p. 201912804.

[65] 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.

[66] 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.

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

[68] Qin, S., Li, Q., Tang, C., & Tu, Y. (2019). Optimal compressed sensing strategies for an array of nonlinear olfactory receptor neurons with and without spontaneous activity. Proceedings of the National Academy of Sciences of the United States of America, 116, 20286–20295.

[69] 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).

[70] Rigotti, M., Barak, O., Warden, M. R., Wang, X. J., Daw, N. D., Miller, E. K., & Fusi, S. (2013). The importance of mixed selectivity in complex cognitive tasks. Nature, 497, 585–590.

[71] Salinas, E. & Abbott, L. F. (1994). Vector reconstruction from firing rates. Journal of Computational Neuroscience.

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

[73] Schaffer, E. S., Stettler, D. D., Kato, D., Choi, G. B., Axel, R., & Abbott, L. F. (2018). Odor Perception on the Two Sides of the Brain: Consistency Despite Randomness. Neuron, 98, 736–742.

[74] 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.

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

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

[77] 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.

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

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

[80] Stettler, D. D. & Axel, R. (2009). Representations of Odor in the Piriform Cortex. Neuron.

[81] Stringer, C., Pachitariu, M., Steinmetz, N., Carandini, M., & Harris, K. D. (2019). High-dimensional geometry of population responses in visual cortex. Nature, 571, 361–365.

[82] 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.

[83] 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.

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

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

[86] 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.

[87] Wei, X. X. & Stocker, A. A. (2016). Mutual information, fisher information, and efficient coding. Neural Computation.

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

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

[90] 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.

[91] Yarrow, S. & Series, P. (2015). The influence of population size, noise strength and behavioral task on Best-Encoded stimulus for neurons with unimodal or monotonic tuning curves. Frontiers in Computational Neuroscience, 9, 1–20.

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

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

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

[95] Zhang, Y. & Sharpee, T. O. (2016). A Robust Feedforward Model of the Olfactory System. PLoS Computational Biology.

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