ESTIMATING DIMENSIONALITY OF NEURAL REPRESENTATIONS FROM FINITE SAMPLES

Anonymous authors
Paper under double-blind review

ABSTRACT

The global dimensionality of a neural representation manifold provides rich insight into the computational process underlying both artificial and biological neural networks. However, all existing measures of global dimensionality are sensitive to the number of samples, i.e., the number of rows and columns of the sample matrix. We show that, in particular, the participation ratio of eigenvalues, a popular measure of global dimensionality, is highly biased with small sample sizes, and propose a bias-corrected estimator that is more accurate with finite samples and with noise. On synthetic data examples, we demonstrate that our estimator can recover the true known dimensionality. We apply our estimator to neural brain recordings, including calcium imaging, electrophysiological recordings, and fMRI data, and to the neural activations in a large language model and show our estimator is invariant to the sample size. Finally, our estimators can additionally be used to measure the local dimensionalities of curved neural manifolds by weighting the finite samples appropriately.

1 INTRODUCTION

How does a population of a million neurons encode an input or stimulus? This question is central in neuroscience and machine learning (ML). In a standard geometric view, the population response to a stimulus is a vector in a high-dimensional space whose axes are individual neurons' activation levels. Varying the stimulus forms a set of representations, a neural manifold, in this space. A basic question in this framework is: what is the dimensionality of that manifold? Although fundamental, many dimensionality estimators are sensitive to sample size (the number of stimuli and recorded neurons) and to measurement noise. Local (intrinsic) dimensionality estimators exist that are invariant to sample size, such as the TwoNN method, but they cannot measure global dimensionality, and they are often highly sensitive to noise (Facco et al., 2017; Denti et al., 2022). Despite the long history and proven utility of global dimensionality in neuroscience and ML, there is no estimator that is resistant to both finite sample size and noise (Mineault et al., 2024). We aim to close this gap.

Global dimensionality, understood as an effective rank given by the number of nonzero or effectively nonzero singular values of a data matrix, provides rich insights. It has been used to understand computation in brains and deep networks, to quantify classification (Chung et al., 2018; Cohen et al., 2020; Sorscher et al., 2022) and regression performance (Zhang, 2002; Caponnetto and De Vito, 2007; Bach, 2013), to train linear probes in artificial neural networks (Shah et al., 2025), and to design brain–computer interface (BCI) decoders (Willett et al., 2021; Menendez et al., 2025; Willsey et al., 2025). Cohen et al. (2020), Chou et al. (2025), and Sorscher et al. (2022) relate global dimensionality to the linear separability of manifolds and observe that dimensionality tends to decrease across successive processing stages in both the convolutional neural network and visual cortex, which improves linear separability. Estimating global dimensionality is also useful for interpretability research in large language models (LLMs). For example, linear probes applied to intermediate LLM layers can reliably classify harmful versus non-harmful content (Shah et al., 2025; Kantamneni et al., 2025; Smith et al., 2025), so tracking global dimensionality across layers yields important insights for AI safety and interpretability (Shah et al., 2025). In BCI, it is a standard practice to estimate the global dimensionality of neural activations to first understand the encoding scheme of the motor cortex and utilize this insight to guide the design of BCI decoders (Willett et al., 2021; Menendez et al., 2025; Willsey et al., 2025). Beyond these applications, analyzing global dimensionality has

long been a staple in neuroscience, ML, genomics (Pocrnic et al., 2016), and behavior science Woo et al. (2023) for characterizing collective behavior in large systems.

Global dimensionality estimated from finite data is systematically biased. In neuroscience, the observed activation matrix has shape $P \times Q$ (stimuli × neurons) and is effectively a random submatrix of a much larger, unobserved matrix. One can present only a small subset of stimuli and record only a subset of neurons. Counting "significant" singular values of the observed matrix, or of its covariance, is highly sensitive to the available numbers of rows and columns, a widely noted challenge across many fields (Woo et al., 2023; Pocrnic et al., 2016; Lehky et al., 2014; Mineault et al., 2024). Common workarounds include subsampling to produce saturation curves that are checked visually as the matrix approaches the full dataset size (Woo et al., 2023), and ad-hoc extrapolation when saturation is not observed (Lehky et al., 2014).

We address this problem with a principled estimation-theoretic approach. We correct the finite-sample bias of a widely used global dimensionality metric, the participation ratio (PR) of covariance eigenvalues. The PR is a soft count of nonzero singular values and is widely used in neuroscience and machine learning (Menendez et al., 2025; Meissner-Bernard et al., 2024; Fortunato et al., 2024; Harvey et al., 2024; Beiran et al., 2023; Niemeyer et al., 2022; Susman et al., 2021; Gao et al., 2017; Rajan et al., 2010; Kirsanov et al., 2025; Schrage et al., 2024; Kuoch et al., 2024; Yerxa et al., 2023; Sorscher et al., 2022; Fort et al., 2022; Mel and Ganguli, 2021; Cohen et al., 2020; Harvey et al., 2024). However, just like other global dimensionality estimators, the existing PR estimators exhibit substantial finite-sample bias (Recanatesi et al., 2022; Menendez et al., 2025). Here, we present a bias-corrected estimator of PR by deriving the unbiased estimators of the numerator and denominator of PR. We also provide an extension that removes the bias contributed by noise, and a variation that measures local dimensionality, which is resistant to noise, unlike the existing popular local dimensionality estimator, TwoNN (Denti et al., 2022).

2 Definitions and Problem Setup

Here, we provide an informal formulation of the problem setup. A more rigorous and thorough problem formulation is presented in the Supplementary section Sec. A.

2.1 Representation Matrix

In neuroscientific and ML experiments, the neural representation data takes the form of a matrix $\Phi \in \mathbb{R}^{P\times Q}$, where each row represents an input (stimulus), and each column represents a single feature (neuron). We assume that the neural activation $\Phi_{i\alpha}$ is given by a hypothetical map ϕ that maps an input x_i and a parameter w_α that parameterizes the neuron:

$$\Phi_{i\alpha} = \phi(x_i, w_\alpha) \tag{1}$$

As it will be evident later, our dimensionality estimator is agnostic of ϕ and the distributions of x_i and w_α. Nonetheless, we define this generative process to establish a clear mental framework.

2.2 Participation Ratio

We are interested in measuring the dimensionality of the activation matrix in the limit where the number of neurons and inputs approach infinity. Let us denote the matrix in this limit as $\Phi^{(\infty)} \in \mathbb{R}^{P^{(\infty)}\times Q^{(\infty)}}$ where $P^{(\infty)}, Q^{(\infty)} \to \infty$. A dimensionality measure should quantify how many non-zero eigenvalues the covariance matrix $\mathbf{K}^{(\infty)} := \frac{1}{Q^{(\infty)}}\Phi^{(\infty)}\Phi^{(\infty)\top}$ has. In the most strict sense, this is given by the rank of $\mathbf{K}^{(\infty)}$. However, the rank is sensitive to small eigenvalues that one might want to ignore. Therefore, a softer count of eigenvalues, participation ratio (PR) γ, has become a popular measure of dimensionality, which is defined as the following:

$$\gamma_0 := \frac{\left(\sum_i \lambda_i\right)^2}{\sum_i \lambda_i^2}$$

where $\{\lambda_i\}$ is the set of all eigenvalues of $\mathbf{K}^{(\infty)}$. Suppose D number of the eigenvalues all take a single value c, and the rest are all zero. In this case, γ_0 is D, which equals the matrix rank. However, if

there is an additional small eigenvalue ϵ, then the PR changes only by a small amount $\gamma_0 \approx D + \mathcal{O}(\epsilon)$, whereas the matrix rank abruptly becomes $D+1$. Therefore γ_0 forms a lower bound on the matrix rank. It has been shown that γ_0 number of the largest eigenvalues typically explain 70–80% of the variance in the data (Gao et al., 2017).

We introduce equivalent ways of expressing the PR, which will be useful in the later sections. We use the fact that the sum of the eigenvalues of a positive semi-definite matrix is given by its trace, and the sum of the squares of the eigenvalues is given by the trace of its square:

$$\gamma_0 \equiv \frac{\frac{1}{P^{(\infty)2}} \mathrm{tr}\left(\mathbf{K}^{(\infty)}\right)^2}{\frac{1}{P^{(\infty)2}} \mathrm{tr}\left(\mathbf{K}^{(\infty)2}\right)} \equiv \frac{\left\langle \bar{v}^{\alpha\beta}_{iijj} \right\rangle}{\left\langle \bar{v}^{\alpha\beta}_{ijij} \right\rangle}, \quad \text{where} \quad \bar{v}^{\alpha\beta}_{ijkl} \coloneqq \Phi^{(\infty)}_{i\alpha}\Phi^{(\infty)}_{j\alpha}\Phi^{(\infty)}_{k\beta}\Phi^{(\infty)}_{l\beta}$$

and $\langle \cdot \rangle$ is a notation for averaging over all free indices, e.g.

$$\left\langle \bar{v}^{\alpha\beta}_{ijlr} \right\rangle = \frac{1}{\#\text{ of summands}} \sum_{ijlr} \sum_{\alpha\beta} \bar{v}^{\alpha\beta}_{ijlr}.$$

The second equality is obtained by simply expanding the matrix notation in terms of the activations, e.g. $\Phi^{(\infty)}_{i\alpha} \equiv \phi(x_i, w_\alpha)$. We normalize both terms so they are $\mathcal{O}(1)$.

We consider the case where each column (neuron) is centered before computing the dimensionality, which is a common practice in neuroscience and ML. Consider a neural manifold in $\mathbb{R}^{Q^{(\infty)}}$ where each point in the manifold is a row vector of $\Phi^{(\infty)}$. This centering operation simply shifts the manifold such that its center of mass is at the origin. The dimensionality γ of the centered manifold is given by

$$\gamma \coloneqq \frac{A}{B} \tag{2}$$

where

$$A \coloneqq \left\langle \bar{v}^{\alpha\beta}_{iijj} \right\rangle - 2\left\langle \bar{v}^{\alpha\beta}_{iijl} \right\rangle + \left\langle \bar{v}^{\alpha\beta}_{ijlr} \right\rangle, \quad \text{and}$$

$$B \coloneqq \left\langle \bar{v}^{\alpha\beta}_{ijij} \right\rangle - 2\left\langle \bar{v}^{\alpha\beta}_{ijjl} \right\rangle + \left\langle \bar{v}^{\alpha\beta}_{ijlr} \right\rangle.$$

Note that we use the notation γ_0 to refer to the uncentered dimensionality, and γ to refer to the centered dimensionality. We study γ for the rest of the paper.

2.3 Sample matrix and naive estimator

The sample activation matrix $\Phi \in \mathbb{R}^{P \times Q}$ is a random submatrix of $\Phi^{(\infty)}$, obtained by selecting the P rows and Q columns of $\Phi^{(\infty)}$ independently and uniformly at random, and then collecting the entries at the intersections of the selected rows and columns. In most neuroscience experiments, one can only observe a subset of neurons in a given brain region, which corresponds to the column sampling. In both neuroscience and ML experiments, one can only present a subset of stimuli (e.g., it is impossible to present all possible natural images), which corresponds to the row sampling. In terms of the generative process defined in Eqn. 1, the sampling of the submatrix is equivalent to sampling P stimuli $\{x_i\}_{i=1}^{P}$ and Q neuronal parameters $\{w_\alpha\}_{\alpha=1}^{Q}$.

Currently, in the literature, the PR is commonly estimated by simply substituting $\Phi^{(\infty)}$ with Φ in Eqn. 2:

$$\gamma_{\text{naive}} \coloneqq \frac{A_{\text{naive}}}{B_{\text{naive}}}, \tag{3}$$

where

$$A_{\text{naive}} \coloneqq \left\langle v_{iijj}^{\alpha\beta} \right\rangle - 2\left\langle v_{iijl}^{\alpha\beta} \right\rangle + \left\langle v_{ijlr}^{\alpha\beta} \right\rangle, \quad B_{\text{naive}} \coloneqq \left\langle v_{ijij}^{\alpha\beta} \right\rangle - 2\left\langle v_{ijjl}^{\alpha\beta} \right\rangle + \left\langle v_{ijlr}^{\alpha\beta} \right\rangle$$

and

$$v_{ijkl}^{\alpha\beta} \coloneqq \Phi_{i\alpha}\Phi_{j\alpha}\Phi_{k\beta}\Phi_{l\beta}.$$

Note that the only difference between Eqn. 2 and Eqn. 3 is that the Eqn. 3 is computed on the submatrix Φ, whereas Eqn. 2 is computed on the true matrix $\Phi^{(\infty)}$. However, this naive estimator is very sensitive to the number of observed neurons Q and stimuli presented P. In the next sections, we identify the source of the sample-size-sensitive bias in this estimator, and then propose our estimator that corrects the bias.

3 Bias in naive estimator

Although the simple substitution in γ_{naive} is intuitive, it leads to a heavily biased estimation of γ. This is because both the numerator and denominator of γ_{naive} are biased estimates of the numerator and denominator of γ: $\mathbb{E}_\Phi\left[A_{\text{naive}}\right] \neq A$ and $\mathbb{E}_\Phi\left[B_{\text{naive}}\right] \neq B$ where $\mathbb{E}_\Phi\left[\cdot\right]$ denotes the average over the independent uniform sampling of the submatrix. In fact, each term in A_{naive} (or B_{naive}) is a biased estimate of a corresponding term in A (or B). For example, $\mathbb{E}_\Phi\left[\left\langle v_{iijj}^{\alpha\beta} \right\rangle\right] \neq \left\langle \bar{v}_{iijj}^{\alpha\beta} \right\rangle$, which are the first terms of A_{naive} and A. If we write this inequality explicitly in terms of the matrix entries $\Phi_{i\alpha} \equiv \phi(x_i, w_\alpha)$, we have

$$\mathbb{E}_\Phi\left[\frac{1}{P^2Q^2}\sum_{ij}\sum_{\alpha\beta}\phi(x_i,w_\alpha)^2\phi(x_j,w_\beta)^2\right] \neq \mathbb{E}_{x,w}\left[\phi(x,w)^2\right]^2,$$

where $\mathbb{E}_{x,w}\left[\cdot\right]$ denotes the average over the distributions of x and w. Consider decomposing the sum in the LHS into the terms where none of the indices coincide, and the rest:

$$\frac{1}{P^2Q^2}\mathbb{E}_\Phi\left[\sum_{i\neq j}\sum_{\alpha\neq\beta}\phi(x_i,w_\alpha)^2\phi(x_j,w_\beta)^2 + \text{rest}\right]$$

Moving the expectation inside the sum over unequal indices, for the first term, we have $\mathbb{E}_\Phi\left[\phi(x_i,w_\alpha)^2\phi(x_j,w_\beta)^2\right]$ for $i \neq j$ and $\alpha \neq \beta$. However, since the row sampling and column sampling are both independent, $\phi(x_i,\cdot)$ and $\phi(x_j,\cdot)$ are independent for $i \neq j$, and $\phi(\cdot,w_\alpha)$ and $\phi(\cdot,w_\beta)$ are also independent for $\alpha \neq \beta$. Therefore, the first term factorizes to $\mathbb{E}_\Phi\left[\phi(x_i,w_\alpha)^2\right]\mathbb{E}_\Phi\left[\phi(x_j,w_\beta)^2\right]$, which is simply $\mathbb{E}_{x,w}\left[\phi(x,w)^2\right]^2$, the quantity we want to estimate. However, the "rest" term is non-zero, i.e. $\mathbb{E}_\Phi\left[\text{rest}\right] \neq 0$, contributing to the bias. Note that in the "rest" term, the indices in $\phi(x_i,w_\alpha)^2\phi(x_j,w_\beta)^2$ are not all unequal, making $\phi(x_i,w_\alpha)^2$ and $\phi(x_j,w_\beta)^2$ correlated, so the expectation cannot be factorized, contributing as bias.

4 Biased-corrected global dimensionality estimator

Having identified in the previous section that overlapping indices in a sum contribute to the bias in each term of A_{naive} (or B_{naive}), we now know that unbiased estimators of A and B can be found by simply averaging over unequal indices. Let us first define a notation for averaging over unequal indices for both rows and columns:

$$\left\langle v_{ijlr}^{\alpha\beta} \right\rangle_{\text{both}} = \frac{1}{\#\text{ of summands}} \sum_{i,j,l,r\ \text{all distinct}} \sum_{\alpha\neq\beta} v_{ijlr}^{\alpha\beta}.$$

Then, our unbiased estimators of A and B are

$$A_{\text{both}} \coloneqq \left\langle v_{iijj}^{\alpha\beta} \right\rangle_{\text{both}} - 2\left\langle v_{iijl}^{\alpha\beta} \right\rangle_{\text{both}} + \left\langle v_{ijlr}^{\alpha\beta} \right\rangle_{\text{both}}, \quad \text{and}$$

$$B_{\text{both}} \coloneqq \left\langle v^{\alpha\beta}_{ijij} \right\rangle_{\text{both}} - 2\left\langle v^{\alpha\beta}_{ijjl} \right\rangle_{\text{both}} + \left\langle v^{\alpha\beta}_{ijlr} \right\rangle_{\text{both}}.$$

Finally, we define our estimator for the true dimensionality as simply the ratio of A_{both} and B_{both}:

$$\gamma_{\text{both}} \coloneqq \frac{A_{\text{both}}}{B_{\text{both}}}.$$

Note that even if $\hat{X}$ and $\hat{Y}$ are unbiased estimators of X and Y, $\hat{X}/\hat{Y}$ is a biased estimate of X/Y. The ratio operation introduces a small but inevitable bias, which cannot be reduced further in a straightforward manner (see Sec. D). In general, however, the bias contributed by the ratio operation is negligible compared to that contributed by biases in the numerator and denominator. We provide a detailed theoretical analysis of the bias and variance of these estimators in Sec. D.

If one desires, one can only correct the bias contributed from row sampling by summing over unequal row indices, but still summing over all column indices. This could be useful if one has full observation of neurons, but the inputs are sampled. In a similar manner, one can only correct the bias contributed by column sampling. We refer to these estimators as γ_{row} and γ_{col}, respectively.

4.1 Noise correction

In many scenarios, the sample matrix is corrupted by an additive or multiplicative noise. This is inevitable in neural recordings. We show that we can correct the bias from the additive or multiplicative noise (or both simultaneously), as long as two sample matrices are obtained over two trials for fixed sets of stimuli and neurons. An alternative naive approach would be to perform N trials and take an element-wise average of the N sample matrices, before passing it to the dimensionality estimator. However, performing multiple trials is typically expensive, and the bias contributed by the noise would be $\mathcal{O}\left(1/\sqrt{N}\right)$ with this naive method. In contrast, our method, inspired by Stringer et al. (2019), only requires $N = 2$ trials, and the bias contributed by the noise is $\mathcal{O}\left(1/P + 1/Q\right)$, much more efficient than the alternative method. There, we assume the sample matrix in t-th trial is generated by

$$\Phi^{(t)}_{i\alpha} = \phi(x_i, w_\alpha) + \eta(x_i, w_\alpha, \epsilon_t)$$

where ϵ_t is sampled independently across trial. If $\eta(x_i, w_\alpha, \epsilon_t) = \epsilon_t \phi(x_i, w_\alpha)$ and ϵ_t is zero-mean, then this models multiplicative noise: $\Phi^{(t)}_{i\alpha} = (1 + \epsilon_t)\,\phi(x_i, w_\alpha)$.

To correct for the bias due to noise, we can simply redefine $v^{\alpha\beta}_{ijkl}$ as $v^{\alpha\beta}_{ijkl} \leftarrow \Phi^{(1)}_{i\alpha}\Phi^{(2)}_{j\alpha}\Phi^{(1)}_{k\beta}\Phi^{(2)}_{l\beta}$ where $\Phi^{(1)}$ and $\Phi^{(2)}$ are the sample matrices from two trials. The rest of the calculation can be performed as explained in the previous section to obtain the dimensionality estimates.

4.2 Weighted dimensionality

Real-life datasets often have outliers that one might want to ignore or downweight when measuring dimensionality. Suppose $\mathcal{S} = \{s_i\}_{i=1}^{P}$ is a set of scaling factors, each of which determines how much we want to weight a given neural response. We now define a weighted sum notation:

$$\left\langle v^{\alpha\beta}_{ijlr} \right\rangle^{\mathcal{S}}_{\text{both}} \coloneqq \frac{\sum_{i\neq j\neq l\neq r} \sum_{\alpha\neq\beta} s_i s_j s_l s_r v^{\alpha\beta}_{ijlr}}{Q\,(Q-1) \sum_{i\neq j\neq l\neq r} s_i s_j s_l s_r}.$$

Then the weighted dimensionality is given by

$$\gamma^{\mathcal{S}}_{\text{both}} \coloneqq \frac{\left\langle v^{\alpha\beta}_{iijj} \right\rangle^{\mathcal{S}}_{\text{both}} - 2\left\langle v^{\alpha\beta}_{iijl} \right\rangle^{\mathcal{S}}_{\text{both}} + \left\langle v^{\alpha\beta}_{ijlr} \right\rangle^{\mathcal{S}}_{\text{both}}}{\left\langle v^{\alpha\beta}_{ijij} \right\rangle^{\mathcal{S}}_{\text{both}} - 2\left\langle v^{\alpha\beta}_{ijjl} \right\rangle^{\mathcal{S}}_{\text{both}} + \left\langle v^{\alpha\beta}_{ijlr} \right\rangle^{\mathcal{S}}_{\text{both}}}.$$

One can also estimate the local dimensionality, i.e. intrinsic dimensionality, around a given point in the representation space by assigning large weights to the neighboring points and small weights to distant points. In conjunction with the noise correction method explained above, our local dimensionality estimator is resistant to the additive/multiplicative noise, unlike popular local dimensionality estimators (see Sec. 5).

4.3 IMPLEMENTATION

The main challenge in implementation is in performing the sum over unequal indices. Vectorizing or parallelizing the sum is highly non-trivial. Consider, for example, a sum of the following form, $\sum_{i\neq j\neq k} u_{ijk}$ for some tensor u. To vectorize this sum, one needs to express it in terms of a regular sum that sums over all indices. After some algebra, the unequal sum can be expanded as:

$$\sum_{i\neq j\neq k} u_{ijk} \equiv \sum_{ijk} u_{ijk} - \sum_{ij} u_{iij} - \sum_{ij} u_{ijj} - \sum_{ij} u_{iji} + 2\sum_{i} u_{iii}$$

where now the vectorized calculation is possible for each sum on a computer. One can simply use the `einsum` subroutine for each sum. In our case, we have a set of 4 indices that need to be unequal, and another set of 2 indices that also need to be unequal, which makes the expanded forms too long to be presented in the main text. The implementable expressions of our estimators are provided in Sec. A.

4.4 SYNTHETIC DATA

We first verify our estimator on synthetic data by testing how quickly different estimators converge to their true dimensionality. Here, we considered the following noisy linear generative process:

$$\Phi_{i\alpha} = x_i \cdot w_\alpha + \epsilon_{i\alpha} \tag{4}$$

We sample Q feature variables $\{w_\alpha\}_{\alpha=1}^{Q}$ and P inputs $\{x_i\}_{i=1}^{P}$ independently from $\mathcal{N}(0, \mathbf{I}_d)$, and the noise term $\{\epsilon_{i\alpha}\}$ from $\mathcal{N}(0, \sigma_\epsilon^2)$ to form a $P \times Q$ sample matrix Φ. In the limit when $P, Q \to \infty$, the true, noise-free, PR should approach d, which we refer to as the true dimensionality γ for this setup. Note that a finite size Φ obtained by the above process can be seen as a random, noisy submatrix of an underlying infinite matrix.

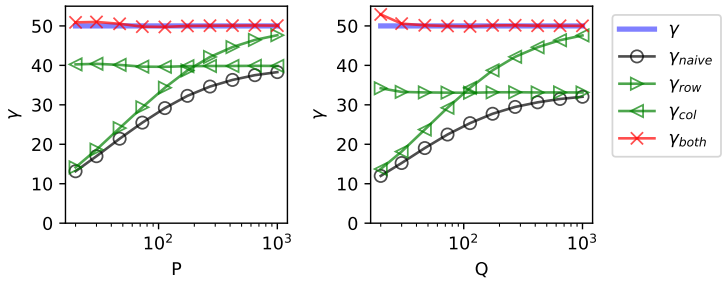

Figure 1: Different dimensionality estimates of the linear model with $d = 50$ and noise variance $\sigma_\epsilon^2 = 0.2$.

We find that our estimator γ_{both} is able to recover the true dimensionality across wide ranges of finite P and Q (Figure 1). Note that our estimator did not require any information about the distributions of x and w, or the map ϕ. We do observe very small bias in our estimator when P or Q is very small, due to the nonlinear effect of taking the ratio of unbiased estimators, as described earlier in Section 4. However, this bias is negligible compared to the bias in the other estimators.

4.5 BRAIN DATA

In real data scenarios, the population dimensionality γ is unknown, and we can only assess the estimators' performance based on how quickly they converge to the dimensionality calculated using the entire dataset. Here, we test our estimator on multiple real neural datasets, all using natural image stimuli:

1. Mouse V1 recorded with calcium imaging Stringer et al. (2019)
2. Macaque V4 recorded with microelectrode arrays (local field potential; LFP) Papale et al. (2025)
3. Macaque IT recorded with microelectrode arrays (spike-sorted) Majaj et al. (2015)

4. Human IT fMRI data Hebart et al. (2023)

By subsampling from each dataset, we vary the number of neuronal units[1] Q or the number of natural images P, and apply the dimensionality estimators. In Figure 2, we report our results as a function of the number of subsamples sampled from the full dataset.

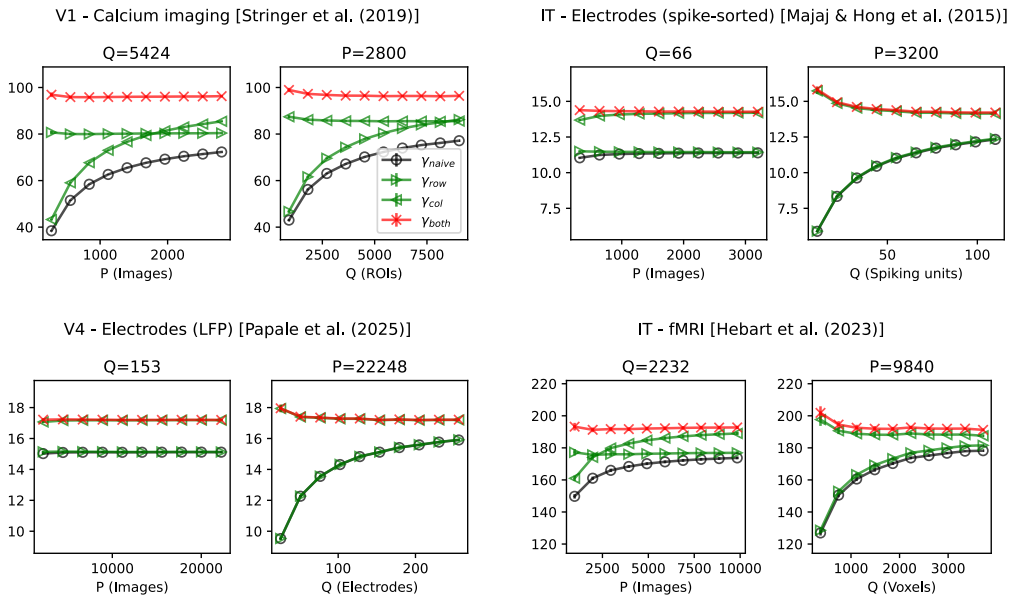

Figure 2: Dimensionality estimates on four different neural recording datasets for varying number of stimuli P, and neural activation units Q, by subsampling from the full dataset. Top left: Mouse V1 (Stringer et al., 2019); Top right: Macaque IT (Majaj et al., 2015); Bottom left: Macaque V4 (Papale et al., 2025); Bottom right: Human IT (Hebart et al., 2023).

Most notably, the empirical mean of γ_{both} is consistently least sensitive to the number of samples, compared to that of the other estimators (Figure 2). Note that when varying the number of rows P, the γ_{row} that corrects the bias due to row sampling is practically invariant, but it is sensitive to Q (Figure 2). The opposite is true for γ_{col}. Finally, γ_{naive} is sensitive to both P and Q and is the most biased. These results indicate that our estimator is useful regardless of neural recording modality and can capture the underlying dimensionality with a relatively much smaller number of samples.

4.6 Artificial Neural Networks

We also apply our estimator to artificial neural networks. In this case, one has access to the entire population of neurons; hence, correcting for overlapping column indices should not make any difference. However, there may be input-limited cases where only a few exemplars of a particular class can be accessed.

Here, we consider this case for evaluating the dimensionality of hidden layers of large language models (LLM). We use the `FLORES+` dataset NLLB Team et al. (2024) for multilingual machine translation and extract representations from the hidden layers of a pretrained `Llama3` base model Grattafiori et al. (2024). The `FLORES+` dataset contains 483 sentences translated to over 200 languages. Here, we picked 9 languages (see Sec. C for details) and extracted their hidden layer representations from the LLM. Since sentences have different lengths, we use the representation of the last token of each sentence. In Fig. 3, we report the average dimensionality across all languages against the input sampling ratio.

[1]They are either ROIs (calcium imaging), spiking units (Ephys), electrodes (Ephys), or voxels (fMRI).

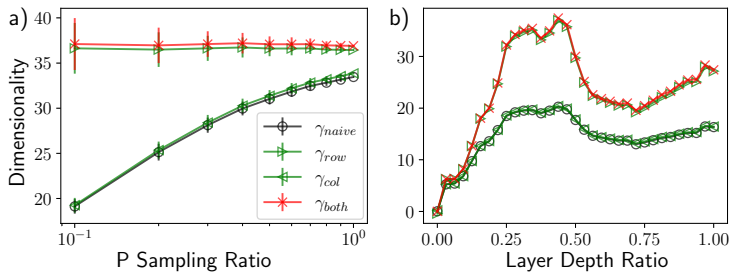

Figure 3: Estimating the task dimensionality of LLM features for different languages. **a)** We calculate the dimensionality of the last layer for each language separately and report its average as a function of the input sampling ratio. In this example, all layers have $Q = 4096$ dimensional representations, and each language has a total of $P = 483$ sentences. The error bars represent the standard deviation for 50 random draws. **b)** The dimensionality profile across layers when the sampling ratio is 0.1 ($P = 48$).

Since we fix the number of neurons and subsample the inputs only, we expect column correction to have a tiny effect compared to row correction. In Fig. 3a, we indeed observe that γ_{col} performs as poorly as the naive estimator and that γ_{row} performs as well as our estimator.

Next, we show how the average dimensionality changes from layer to layer in Fig. 3b when the sample size is small. While the naive estimator significantly underestimates the dimensionality, it preserves the overall profile of dimensionality across layers. However, our estimator reveals more fine-grained features of the layerwise dimensionality that are hidden otherwise. We provide additional details about our experiments in Sec. C.

Finally, we would like to comment on the interesting behavior in Fig. 3b, where the dimensionality increases towards the mid-layers and decreases again. This behavior was observed previously Valeriani et al. (2023) and was recently reported in Skean et al. (2025) by using matrix-based entropy measures Giraldo et al. (2014), which includes the logarithm of the dimensionality as a special case. It would be interesting to see if our estimator can be extended to these more general measures in future work.

5 Extension: Local Dimensionality Estimation

In the earlier section, we described the procedure of weighing samples based on importance. Here we demonstrate that one can extend this framework to measure local dimensionality of a manifold, adopting a method inspired by Recanatesi et al. (2022). From a given data point $\overrightarrow{\Phi}_0$ (an arbitrary row vector of Φ), one can measure the distance to the rest of the data points in a manifold dataset. One can then discard the data points that are further than some predefined distance r, by giving them zero weights and giving the points inside the local ball radius of r uniform weights. The resulting weighted dimensionality estimate is denoted $\gamma^{\text{local}}_{\text{both}}\left(\overrightarrow{\Phi}_0, r\right)$. One can obtain the average local dimensionality of a given manifold by taking the average of $\gamma^{\text{local}}_{\text{both}}\left(\overrightarrow{\Phi}_0, r\right)$ over all available $\overrightarrow{\Phi}_0$'s:

$$\gamma^{\text{local}}_{\text{both}}(r) := \frac{1}{P}\sum_{i=1}^{P} \gamma^{\text{local}}_{\text{both}}\left(\overrightarrow{\Phi_{i,:}}, r\right)$$

where $\overrightarrow{\Phi_{i,:}}$ denotes ith row vector of Φ. In our experiments, we use the Mahalanobis distance with a local metric. See Sec. E for a formal definition of the local dimensionality estimator and distance metric. One can similarly define $\gamma^{\text{local}}_{\text{naive}}(r)$, $\gamma^{\text{local}}_{\text{row}}(r)$, and $\gamma^{\text{local}}_{\text{col}}(r)$.

To confirm that our estimator can indeed capture the true local dimensionality, we test it on a synthetic dataset. The synthetic dataset is created using a random Fourier feature (RFF) model whose covariance kernel converges to the radial basis function (RBF) kernel in the limit where the number of features (neurons) approaches infinity. A noisy RFF model is defined as

$$\phi(x, \{w, b\}) = \sin(x \cdot w + b)$$

where $w \in \mathbb{R}^d$ is sampled independently from a normal distribution $\mathcal{N}(0, \mathbf{I}_d)$, b is sampled independently from a uniform distribution $\mathcal{U}(-\pi/2, \pi/2)$. The data matrix $\Phi \in \mathbb{R}^{P \times Q}$ is created by presenting P number of inputs (x's) sampled from $\mathcal{N}(0, \sigma_x^2 \mathbf{I}_d)$ to Q number of random features, such that for a trial t, $\Phi_{i\alpha}^{(t)} = \phi(x_i, \{w_\alpha, b_\alpha\}) + \epsilon_{i,\alpha,t}$ where $\epsilon_{i,\alpha,t}$ is the noise term. We emphasize that the noise term $\epsilon_{i,\alpha,t}$ is sampled independently across all i, α, and t from $\mathcal{N}(0, \sigma_\epsilon^2)$. In the limit of infinite P and Q, the true noise-free local dimensionality of the Q-dimensional representation manifold should approach d.

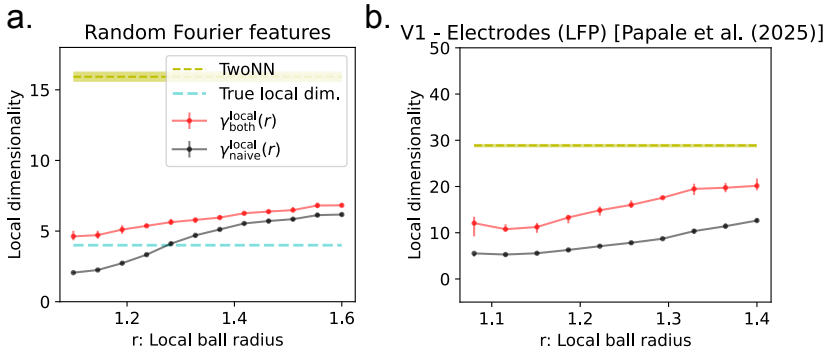

Figure 4: **a**. Estimating the local dimensionality of the random Fourier feature model using TwoNN, $\gamma^{\text{local}}_{\text{naive}}(r)$, and $\gamma^{\text{local}}_{\text{both}}(r)$, while varying the radius of the local ball for the latter two estimators. Signal-to-noise ratio is approximately 3.33 ($\sigma_\epsilon = 0.3$). **b.** Estimating the local dimensionality of the macaque V1 LFP measured with electrode arrays (Papale et al., 2025).

We find that our local dimensionality measure $\gamma^{\text{local}}_{\text{both}}(r)$ recovers the true local dimensionality d despite the presence of noise as we decrease the local ball radius to the smallest allowable length (Figure 4). The smallest radius is determined by requiring that at least four data points lie inside a local ball. Recall that our estimator requires at least four distinct indices, so four is the minimum allowed dataset size. In contrast, the widely adopted local dimensionality estimator TwoNN significantly overestimates the local dimensionality due to its susceptibility to noise. The naive local estimator $\gamma^{\text{local}}_{\text{naive}}(r)$ also fails to recover the true dimensionality, since it underestimates when there are only a small number of samples in a local ball. This highlights the need to correct for bias due to the small sample size, especially when measuring local dimensionality.

We also apply the local dimensionality estimators to the real brain dataset, V1 LFP measurements from macaque (Papale et al., 2025). We find that the estimate from our estimator $\gamma^{\text{local}}_{\text{both}}(r)$ in the small radius limit is much smaller than the TwoNN estimate (Figure 4).

6 Discussions

In this paper, we resolve the timely issue of the sensitivity of the global dimensionality estimator to the sample size and measurement noise. To this end, we correct the bias in the PR, a popular measure of global dimensionality. We then apply our estimators to a synthetic dataset, real neuronal datasets of various recoding modalities, and a large language model representation, to show that our estimator is indeed resistant to the sample size. As extensions, we also show how we can measure local dimensionality using our dimensionality estimators.

7 Limitations

For measuring local dimensionality, our estimator $\gamma^{\text{local}}_{\text{both}}(r)$ is computationally more expensive compared to TwoNN, since it needs to be swept across a range of local ball radius r so that one can inspect the convergence $\gamma^{\text{local}}_{\text{both}}(r)$ in the small r limit. However, the computation can be easily parallelized over multiple threads.

REFERENCES

Francis Bach. Sharp analysis of low-rank kernel matrix approximations. In *Conference on learning theory*, pages 185–209. PMLR, 2013.

Manuel Beiran, Nicolas Meirhaeghe, Hansem Sohn, Mehrdad Jazayeri, and Srdjan Ostojic. Parametric control of flexible timing through low-dimensional neural manifolds. *Neuron*, 111(5):739–753.e8, 2023. ISSN 0896-6273. doi: https://doi.org/10.1016/j.neuron.2022.12.016. URL https://www.sciencedirect.com/science/article/pii/S0896627322010893.

Andrea Caponnetto and Ernesto De Vito. Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3):331–368, 2007.

Chi-Ning Chou, Royoung Kim, Luke A. Arend, Yao-Yuan Yang, Brett D. Mensh, Won Mok Shim, Matthew G. Perich, and SueYeon Chung. Geometry linked to untangling efficiency reveals structure and computation in neural populations. *bioRxiv*, 2025. doi: 10.1101/2024.02.26.582157. URL https://www.biorxiv.org/content/early/2025/03/31/2024.02.26.582157.

SueYeon Chung, Daniel D Lee, and Haim Sompolinsky. Classification and geometry of general perceptual manifolds. *Physical Review X*, 8(3):031003, 2018.

Uri Cohen, SueYeon Chung, Daniel D Lee, and Haim Sompolinsky. Separability and geometry of object manifolds in deep neural networks. *Nature communications*, 11(1):746, 2020.

Francesco Denti, Diego Doimo, Alessandro Laio, and Antonietta Mira. The generalized ratios intrinsic dimension estimator. *Scientific Reports*, 12(1):20005, 2022.

Elena Facco, Maria d'Errico, Alex Rodriguez, and Alessandro Laio. Estimating the intrinsic dimension of datasets by a minimal neighborhood information. *Scientific reports*, 7(1):12140, 2017.

Stanislav Fort, Ekin Dogus Cubuk, Surya Ganguli, and Samuel S. Schoenholz. What does a deep neural network confidently perceive? the effective dimension of high certainty class manifolds and their low confidence boundaries. *ArXiv*, abs/2210.05546, 2022. URL https://api.semanticscholar.org/CorpusID:252815950.

Cátia Fortunato, Jorge Bennasar-Vázquez, Junchol Park, Joanna C. Chang, Lee E. Miller, Joshua T. Dudman, Matthew G. Perich, and Juan A. Gallego. Nonlinear manifolds underlie neural population activity during behaviour. *bioRxiv*, 2024. doi: 10.1101/2023.07.18.549575.

Peiran Gao, Eric Trautmann, Byron Yu, Gopal Santhanam, Stephen Ryu, Krishna Shenoy, and Surya Ganguli. A theory of multineuronal dimensionality, dynamics and measurement. *bioRxiv*, 2017. doi: 10.1101/214262.

Luis Gonzalo Sanchez Giraldo, Murali Rao, and Jose C Principe. Measures of entropy from data using infinitely divisible kernels. *IEEE Transactions on Information Theory*, 61(1):535–548, 2014.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Sarah E. Harvey, David Lipshutz, and Alex H. Williams. What representational similarity measures imply about decodable information, 2024. URL https://arxiv.org/abs/2411.08197.

Martin N Hebart, Oliver Contier, Lina Teichmann, Adam H Rockter, Charles Y Zheng, Alexis Kidder, Anna Corriveau, Maryam Vaziri-Pashkam, and Chris I Baker. Things-data, a multimodal collection of large-scale datasets for investigating object representations in human brain and behavior. *eLife*, 12:e82580, feb 2023. ISSN 2050-084X. doi: 10.7554/eLife.82580.

Subhash Kantamneni, Joshua Engels, Senthooran Rajamanoharan, Max Tegmark, and Neel Nanda. Are sparse autoencoders useful? a case study in sparse probing. *arXiv preprint arXiv:2502.16681*, 2025.

Artem Kirsanov, Chi-Ning Chou, Kyunghyun Cho, and SueYeon Chung. The geometry of prompting: Unveiling distinct mechanisms of task adaptation in language models. *arXiv preprint arXiv:2502.08009*, 2025.

Michael Kuoch, Chi-Ning Chou, Nikhil Parthasarathy, Joel Dapello, James J DiCarlo, Haim Sompolinsky, and SueYeon Chung. Probing biological and artificial neural networks with task-dependent neural manifolds. In *Conference on Parsimony and Learning*, pages 395–418. PMLR, 2024.

Sidney R Lehky, Roozbeh Kiani, Hossein Esteky, and Keiji Tanaka. Dimensionality of object representations in monkey inferotemporal cortex. *Neural computation*, 26(10):2135–2162, 2014.

Najib J Majaj, Ha Hong, Ethan A Solomon, and James J DiCarlo. Simple learned weighted sums of inferior temporal neuronal firing rates accurately predict human core object recognition performance. *Journal of Neuroscience*, 35(39):13402–13418, 2015.

Claire Meissner-Bernard, Friedemann Zenke, and Rainer W Friedrich. Geometry and dynamics of representations in a precisely balanced memory network related to olfactory cortex. December 2024. doi: 10.7554/elife.96303.2. URL http://dx.doi.org/10.7554/eLife.96303.2.

Gabriel Mel and Surya Ganguli. A theory of high dimensional regression with arbitrary correlations between input features and target functions: sample complexity, multiple descent curves and a hierarchy of phase transitions. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 7578–7587. PMLR, 18–24 Jul 2021.

Jorge Aurelio Menendez, Jay A Hennig, Matthew D Golub, Emily R Oby, Patrick T Sadtler, Aaron P Batista, Steven M Chase, Byron M Yu, and Peter E Latham. A theory of brain-computer interface learning via low-dimensional control. *bioRxiv*, pages 2024–04, 2025.

Patrick Mineault, Niccolò Zanichelli, Joanne Zichen Peng, Anton Arkhipov, Eli Bingham, Julian Jara-Ettinger, Emily Mackevicius, Adam Marblestone, Marcelo Mattar, Andrew Payne, et al. Neuroai for ai safety. *arXiv preprint arXiv:2411.18526*, 2024.

James E Niemeyer, Poornima Gadamsetty, Chanwoo Chun, Sherika Sylvester, Jacob P Lucas, Hongtao Ma, Theodore H Schwartz, and Emre RF Aksay. Seizures initiate in zones of relative hyperexcitation in a zebrafish epilepsy model. *Brain*, 145(7):2347–2360, 2022.

NLLB Team, Marta R. Costa-jussa, James Cross, Onur Celebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzman, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841–846, 2024. ISSN 1476-4687. URL https://doi.org/10.1038/s41586-024-07335-x.

Paolo Papale, Feng Wang, Matthew W Self, and Pieter R Roelfsema. An extensive dataset of spiking activity to reveal the syntax of the ventral stream. *Neuron*, 2025.

Ivan Pocrnic, Daniela AL Lourenco, Yutaka Masuda, Andres Legarra, and Ignacy Misztal. The dimensionality of genomic information and its effect on genomic prediction. *Genetics*, 203(1): 573–581, 2016.

Kanaka Rajan, L Abbott, and Haim Sompolinsky. Inferring stimulus selectivity from the spatial structure of neural network dynamics. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper_files/paper/2010/file/19b650660b253761af189682e03501dd-Paper.pdf.

Stefano Recanatesi, Serena Bradde, Vijay Balasubramanian, Nicholas A. Steinmetz, and Eric Shea-Brown. A scale-dependent measure of system dimensionality. *Patterns*, 3(8):100555, 2022. ISSN 2666-3899. doi: https://doi.org/10.1016/j.patter.2022.100555. URL https://www.sciencedirect.com/science/article/pii/S266638992200160X.

Linden Schrage, Kazuki Irie, and Haim Sompolinsky. Neural representational geometry of concepts in large language models. In *NeurIPS 2024 Workshop on Symmetry and Geometry in Neural Representations*, 2024.

McNair Shah, Saleena Angeline, Adhitya Rajendra Kumar, Naitik Chheda, Kevin Zhu, Vasu Sharma, Sean O'Brien, and Will Cai. The geometry of harmfulness in llms through subconcept probing. *arXiv preprint arXiv:2507.21141*, 2025.

Oscar Skean, Md Rifat Arefin, Dan Zhao, Niket Patel, Jalal Naghiyev, Yann LeCun, and Ravid Shwartz-Ziv. Layer by layer: Uncovering hidden representations in language models. *arXiv preprint arXiv:2502.02013*, 2025.

Lewis Smith, Senthooran Rajamanoharan, Arthur Conmy, Callum McDougall, János Kramár, Tom Lieberum, Rohin Shah, and Neel Nanda. Negative results for saes on downstream tasks and deprioritising sae research (gdm mech interp team progress update# 2), march 2025. *URL https://www. lesswrong. com/posts/4uXCAJNuPKtKBsi28/negative-results-for-saes-on-downstream-tasks. LessWrong post*, 2025.

Ben Sorscher, Surya Ganguli, and Haim Sompolinsky. Neural representational geometry underlies few-shot concept learning. *Proceedings of the National Academy of Sciences*, 119(43): e2200800119, 2022.

Carsen Stringer, Marius Pachitariu, Nicholas Steinmetz, Matteo Carandini, and Kenneth D Harris. High-dimensional geometry of population responses in visual cortex. *Nature*, 571(7765):361–365, 2019.

Lee Susman, Francesca Mastrogiuseppe, Naama Brenner, and Omri Barak. Quality of internal representation shapes learning performance in feedback neural networks. *Phys. Rev. Res.*, 3: 013176, Feb 2021. doi: 10.1103/PhysRevResearch.3.013176. URL https://link.aps.org/doi/10.1103/PhysRevResearch.3.013176.

Lucrezia Valeriani, Diego Doimo, Francesca Cuturello, Alessandro Laio, Alessio Ansuini, and Alberto Cazzaniga. The geometry of hidden representations of large transformer models. *Advances in Neural Information Processing Systems*, 36:51234–51252, 2023.

Francis R Willett, Donald T Avansino, Leigh R Hochberg, Jaimie M Henderson, and Krishna V Shenoy. High-performance brain-to-text communication via handwriting. *Nature*, 593(7858): 249–254, 2021.

Matthew S Willsey, Nishal P Shah, Donald T Avansino, Nick V Hahn, Ryan M Jamiolkowski, Foram B Kamdar, Leigh R Hochberg, Francis R Willett, and Jaimie M Henderson. A high-performance brain–computer interface for finger decoding and quadcopter game control in an individual with paralysis. *Nature Medicine*, 31(1):96–104, 2025.

Theodosia Woo, Xitong Liang, Dominic A Evans, Olivier Fernandez, Friedrich Kretschmer, Sam Reiter, and Gilles Laurent. The dynamics of pattern matching in camouflaging cuttlefish. *Nature*, 619(7968):122–128, 2023.

Thomas Yerxa, Yilun Kuang, Eero Simoncelli, and SueYeon Chung. Learning efficient coding of natural images with maximum manifold capacity representations. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Tong Zhang. Effective dimension and generalization of kernel learning. *Advances in Neural Information Processing Systems*, 15, 2002.

A Derivation of Bias Corrected Estimators of Dimensionality

A.1 Definitions

Kernel integral operator

We assume there are latent variables, x and w, associated with stimulus and neuron, respectively. To model the sampling of stimulus, we assume there is a distribution $\rho_{\mathcal{X}}$ over the set $\mathcal{X}$ of all stimulus latent variables. Similarly, to model the sampling of neurons, we assume there is a distribution $\rho_{\mathcal{W}}$ over the set $\mathcal{W}$ of all neuron latent variables. Then, we assume the entry of the sample matrix is given by a map $\phi : \mathcal{X} \times \mathcal{W} \to \mathbb{R}$

$$\Phi_{i\alpha} = \phi(x_i, w_\alpha). \tag{S1}$$

We let P and Q denote the number of sampled stimuli and neurons in this paper, i.e. $\Phi \in \mathbb{R}^{P\times Q}$. Assume ϕ is square-integrable with respect to $\rho_{\mathcal{X}}$ and $\rho_{\mathcal{W}}$. Now, define the associated kernel functions

$$k(x, x') \coloneqq \int d\rho_{\mathcal{W}}(w)\, \phi(x, w)\phi(x', w), \quad \text{and} \tag{S2}$$

$$\tilde{k}(w, w') \coloneqq \int d\rho_{\mathcal{X}}(x)\, \phi(x, w)\phi(x, w'). \tag{S3}$$

There is a kernel integral operator that is associated to k, $T_k : \mathcal{L}^2(\rho_{\mathcal{X}}, \mathcal{X}) \to \mathcal{L}^2(\rho_{\mathcal{X}}, \mathcal{X})$, defined as

$$T_k f = \int d\rho_{\mathcal{X}}(x)\, k(\cdot, x) f(x). \tag{S4}$$

T_k is a trace operator and is analogous to the population covariance matrix K_{all} introduced earlier. The tuple $(\phi, \rho_{\mathcal{X}}, \rho_{\mathcal{W}})$ uniquely defines a generative process of data.

Participation ratio

The eigenvalues $\{\lambda_i\}_{i=1}^{\infty}$ of T_k are implicitly defined by

$$T_k e_i = \lambda_i e_i \quad \forall i \tag{S5}$$

where $e_i \in \mathcal{L}^2(\rho_{\mathcal{X}}, \mathcal{X})$, and $\{e_i\}_{i=1}^{\infty}$ forms an orthonormal set. There are a countably infinite number of eigenvalues. Then, the participation ratio of these eigenvalues is defined as

$$\gamma \coloneqq \frac{\left(\sum_{i=1}^{\infty} \lambda_i\right)^2}{\sum_{i=1}^{\infty} \lambda_i^2} \equiv \frac{\left(\int d\rho_{\mathcal{X}}(x)\, k(x, x)\right)^2}{\int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\, k(x, y)^2}. \tag{S6}$$

We refer to γ as the true effective dimensionality of T_k. If n number of λ_i's take a constant value c and the rest are 0, then the participation ratio is n, matching the definition of rank. However, if we decrease one of the positive eigenvalues to be smaller than c (but still positive), then the participation ratio is between $n-1$ and n, reflecting the fact that the dimensionality is effectively slightly less than n. Therefore, the participation ratio forms a lower bound on the rank, and is "softer" than the rank.

Typically, the participation ratio is computed on the eigenvalues $\{\eta_i\}$ of a sample covariance matrix $K \coloneqq \frac{1}{Q}\Phi\Phi^\top$:

$$\hat{\gamma}_0 \coloneqq \frac{\left(\sum_{i=1}^{\infty} \eta_i\right)^2}{\sum_{i=1}^{\infty} \eta_i^2} \equiv \frac{\text{tr}(K)^2}{\text{tr}\,(K^2)}. \tag{S7}$$

A.2 Centered kernel

Suppose $k_c(x, y)$ is a centered kernel, where activations for each neuron is centered:

$$k_c(x_i, x_j) = \int d_{\mathcal{W}}\rho(w)\, \phi_s(x, w)\phi_s(y, w)$$

where

$$\phi_s(x, w) = \phi(x, w) - \int d\rho_{\mathcal{X}}(z)\, \phi(z, w).$$

Similarly, $\tilde{k}_c(x, y)$ is another centered kernel, where activations for each stimulus is centered:

$$\tilde{k}_c(x_i, x_j) = \int d_{\mathcal{W}}\rho(w)\, \phi_f(x, w)\phi_f(y, w)$$

where

$$\phi_f(x, w) = \phi(x, w) - \int d\rho_{\mathcal{W}}(z)\, \phi(x, z).$$

Before centering, it did not matter whether we computed the dimensionality with k or $\tilde{k}$:

$$\hat{\gamma} \coloneqq \frac{\left(\int d\rho_{\mathcal{X}}(x)\, k(x, x)\right)^2}{\int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\, k(x, y)^2} \equiv \frac{\left(\int d\rho_{\mathcal{W}}(w)\, \tilde{k}(w, w)\right)^2}{\int d\rho_{\mathcal{W}}(w) d\rho_{\mathcal{W}}(u)\, \tilde{k}(w, u)^2}.$$

Not only the ratio, but also the numerators and denominators of the two expressions match respectively. However, the centered dimensionalities are different

$$\hat{\gamma}^{\text{task}} \neq \hat{\gamma}^{\text{neuron}}, \quad \text{where}$$

$$\hat{\gamma}^{\text{task}} = \frac{\left(\int d\rho_{\mathcal{X}}(x)\, k_c(x, x)\right)^2}{\int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\, k_c(x, y)^2}, \quad \text{and} \quad \hat{\gamma}^{\text{neuron}} = \frac{\left(\int d\rho_{\mathcal{W}}(w)\, \tilde{k}_c(w, w)\right)^2}{\int d\rho_{\mathcal{W}}(w) d\rho_{\mathcal{W}}(u)\, \tilde{k}_c(w, u)^2}.$$

For now, let us focus on the numerator and the denominator of $\hat{\gamma}^{\text{task}}$. Note that the numerator can be expressed in terms of the original kernel k as:

$$\hat{A} \coloneqq \left(\int d\rho_{\mathcal{X}}(x)\, k_c(x, x)\right)^2$$

$$= \left(\int d\rho_{\mathcal{X}}(x)\; k(x, x)\right)^2 - 2\int d\rho_{\mathcal{X}}(x)\; k(x, x) \int d\rho_{\mathcal{X}}(x)\rho_{\mathcal{X}}\rho(y)\; k(x, y) + \left(\int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\; k(x, y)\right)^2$$

The denominator can be expanded as:

$$\hat{B} = \int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\, k_c(x, y)^2$$

$$= \int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\; k(x, y)^2 - 2\int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y) d\rho_{\mathcal{X}}(z)\; k(x, z)k(z, y) + \left(\int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\; k(x, y)\right)^2$$

A.3 Derivation of the estimators

The goal is to derive an unbiased estimator of each one of the six terms (five unique terms) above. The five unique terms are

$$\hat{t}^1 \coloneqq \left(\int d\rho_{\mathcal{X}}(x)\; k(x, x)\right)^2, \quad \hat{t}^2 \coloneqq \int d\rho_{\mathcal{X}}(x)\; k(x, x) \int d\rho_{\mathcal{X}}(x)\rho_{\mathcal{X}}\rho(y)\; k(x, y),$$

$$\hat{t}^3 \coloneqq \int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\; k(x, y)^2, \quad \hat{t}^4 \coloneqq \int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y) d\rho_{\mathcal{X}}(z)\; k(x, z)k(z, y),$$

$$\text{and} \quad \hat{t}^5 \coloneqq \left(\int d\rho_{\mathcal{X}}(x) d\rho_{\mathcal{X}}(y)\, k(x, y)\right)^2.$$

With these terms, the numerator of $\hat{\gamma}^{\text{task}}$ is

$$\hat{A} = \hat{t}^1 - 2\hat{t}^2 + \hat{t}^5,$$

and the denominator is

$$\hat{B} = \hat{t}^3 - 2\hat{t}^4 + \hat{t}^5.$$

Consider $\hat{t}^5$ as an example. We start with deriving a naive biased estimator, since it is easiest.

$$t^1_{\text{naive}} \coloneqq \frac{1}{P^2Q^2} \sum_{ij} \sum_{\alpha\beta} \phi(x_i w_\alpha)^2 \phi(x_j w_\beta)^2$$

We arrived at the above by simple derivation:

$$P^2Q^2 t^1_{\text{naive}} \coloneqq \left(\text{Tr}(K)\right)^2 = \left(\sum_i \sum_\alpha \Phi^2_{i\alpha} \right)^2 \tag{S8}$$

$$= \sum_{ij} \sum_{\alpha\beta} \Phi^2_{i\alpha} \Phi^2_{j\beta} = \sum_{ij} \sum_{\alpha\beta} \Phi^2_{i\alpha} \Phi^2_{j\beta} = \sum_{ij} \sum_{\alpha\beta} \phi(x_i w_\alpha)^2 \phi(x_j w_\beta)^2. \tag{S9}$$

The reason t^1_{naive} is biased, i.e. $\hat{t}^1 \neq \left\langle t^1_{\text{naive}} \right\rangle_\Phi$ where $\langle \cdot \rangle_\Phi$ is the average over all submatrices, is the following:

$$\begin{aligned}
\left\langle t^1_{\text{naive}} \right\rangle_\Phi =& \left\langle \frac{1}{P^2Q^2} \sum_{ij} \sum_{\alpha\beta} \phi(x_i w_\alpha)^2 \phi(x_j w_\beta)^2 \right\rangle_{\{x_i\},\{w_\alpha\}} \\
=& \frac{1}{P^2Q^2} \Bigg(\sum_{i\neq j} \sum_{\alpha\neq\beta} \left\langle \phi(x_i w_\alpha)^2 \phi(x_j w_\beta)^2 \right\rangle + \sum_i \sum_{\alpha\neq\beta} \left\langle \phi(x_i w_\alpha)^2 \phi(x_i w_\beta)^2 \right\rangle \\
& + \sum_{i\neq j} \sum_\alpha \left\langle \phi(x_i w_\alpha)^2 \phi(x_j w_\alpha)^2 \right\rangle + \sum_i \sum_\alpha \left\langle \phi(x_i w_\alpha)^4 \right\rangle \Bigg) \\
=& \frac{(P-1)(Q-1)}{PQ} \left\langle k(x,x) \right\rangle^2_{x\sim\rho_\mathcal{X}} + \frac{Q-1}{PQ} \left\langle k(x,x)^2 \right\rangle_{x\sim\rho_\mathcal{X}} \\
& + \frac{P-1}{PQ} \left\langle \tilde{k}(w,w)^2 \right\rangle_{w\sim\rho_\mathcal{W}} + \frac{1}{PQ} \left\langle \phi(x,w)^4 \right\rangle_{x\sim\rho_\mathcal{X}, w\sim\rho_\mathcal{W}}
\end{aligned} \tag{S10}$$

Note that in the first term, $\left\langle k(x,x) \right\rangle^2_{x\sim\rho_\mathcal{X}} = \hat{t}^1$, so the leading order term in $\left\langle t^1_{\text{naive}} \right\rangle_\Phi$ is unbiased. However, the rest of the terms contribute to $\mathcal{O}\left(\frac{1}{P} + \frac{1}{Q}\right)$ bias. This shows that we can derive the unbiased estimator by simply summing over disjoint indices, and normalizing by the number of summands:

$$t^1_{\text{both}} = \frac{1}{P(P-1)Q(Q-1)} \sum_{i\neq j} \sum_{\alpha\neq\beta} \phi(x_i w_\alpha)^2 \phi(x_j w_\beta)^2.$$

Then we have $\left\langle t^1_{\text{both}} \right\rangle_\Phi = \hat{t}^1$. Applying the same logic to all five terms $\hat{t}^1$, $\hat{t}^2$, $\hat{t}^3$, $\hat{t}^4$, and $\hat{t}^5$, we arrive at the following unbiased estimators:

$$t^1_{\text{both}} = \frac{1}{P(P-1)Q(Q-1)} \sum_{i\neq j} \sum_{\alpha\neq\beta} v^{\alpha\beta}_{iijj},$$

$$t^2_{\text{both}} = \frac{1}{P(P-1)(P-2)Q(Q-1)} \sum_{i\neq j\neq l} \sum_{\alpha\neq\beta} v^{\alpha\beta}_{iijl},$$

$$t^3_{\text{both}} = \frac{1}{P(P-1)Q(Q-1)} \sum_{i\neq j} \sum_{\alpha\neq\beta} v^{\alpha\beta}_{ijij},$$

$$t^4_{\text{both}} = \frac{1}{P\,(P-1)\,(P-2)\,Q\,(Q-1)} \sum_{i\neq j\neq l} \sum_{\alpha\neq\beta} v^{\alpha\beta}_{ijjl}, \quad \text{and}$$

$$t^5_{\text{both}} = \frac{1}{P\,(P-1)\,(P-2)\,(P-3)\,Q\,(Q-1)} \sum_{i\neq j\neq l\neq r} \sum_{\alpha\neq\beta} v^{\alpha\beta}_{ijlr},$$

where

$$v^{\alpha\beta}_{ijlr} \coloneqq \Phi_{i\alpha}\Phi_{j\alpha}\Phi_{k\beta}\Phi_{l\beta}.$$

The remaining challenge is computing the sums over disjoint indices. The implementation is challenging, since computing this sum over a loop can be slow on a computer. Therefore, one needs to re-express a disjoint sum into a linear combination of regular sums. For example:

$$\sum_{i\neq j} r_{ij} = \sum_{i,j} r_{ij} - \sum_i r_i.$$

This becomes non-trivial as the number of indices increases. Consider An example with three indices:

$$\begin{aligned}\sum_{i\neq j\neq k} r_{ijk} &= \sum_{i,j,k} r_{ijk}\left(1-\delta_{ij}\right)\left(1-\delta_{ik}\right)\left(1-\delta_{jk}\right)\\ &= \sum_{i,j,k} r_{ijk}\left(1-\delta_{jk}-\delta_{ik}-\delta_{ij}+2\delta_{ijk}\right)\\ &= \sum_{i,j,k} r_{ijk} - \sum_{i,j} r_{ijj} - \sum_{i,j} r_{iji} - \sum_{i,j} r_{iij} + 2\sum_{i,j,k} r_{iii}.\end{aligned}$$

Using this technique, we find that the terms can be expanded into the following. Let

$$r_{ijkl} \coloneqq \frac{1}{Q\,(Q-1)} \sum_{\alpha\neq\beta} v^{\alpha\beta}_{ijlr} = \frac{1}{Q\,(Q-1)} \left(\sum_{\alpha,\beta} v^{\alpha\beta}_{ijlr} - \sum_{\alpha} v^{\alpha\alpha}_{ijlr} \right)$$

then

$$t^1_{\text{both}} = \frac{\sum_{ij} r_{iijj} - \sum_i r_{iiii}}{P\,(P-1)},$$

$$t^2_{\text{both}} = \frac{\sum_{ijl} r_{iijl} - 2\sum_{ij} r_{ijjj} - \sum_{ij} r_{iijj} + 2\sum_i r_{iiii}}{P\,(P-1)\,(P-2)},$$

$$t^3_{\text{both}} = \frac{\sum_{ij} r_{ijij} - \sum_i r_{iiii}}{P\,(P-1)},$$

$$t^4_{\text{both}} = \frac{\sum_{ijl} r_{ijjl} - 2\sum_{ij} r_{iiij} - \sum_{ij} r_{ijij} + \sum_i 2r_{iiii}}{P\,(P-1)\,(P-3)}, \quad \text{and}$$

$$t^5_{\text{both}} = \frac{\sum_{ijlm} r_{ijlm} - 2\sum_{ijl}\left(r_{iijl}+2r_{ijjl}\right) + \sum_{ij}\left(r_{iijj}+8r_{ijjj}+2r_{ijij}\right) - 6\sum_i r_{iiii}}{P\,(P-1)\,(P-2)\,(P-3)}.$$

Finally, the unbiased estimators of the numerator and the denominator of $\hat{\gamma}_{\text{task}}$ are

$$A_{\text{both}} = t^1_{\text{both}} - 2t^2_{\text{both}} + t^5_{\text{both}}, \quad \text{and}$$

$$B_{\text{both}} = t^3_{\text{both}} - 2t^4_{\text{both}} + t^5_{\text{both}}.$$

By simply taking the ratio of these unbiased estimators, we obtain our dimensionality estimator

$$\gamma_{\text{both}} = \frac{A_{\text{both}}}{B_{\text{both}}}.$$

Deriving A_{row} and B_{row} can be achieved by simply summing over the columns with the regular sum, but the rows with disjoint indices: Let

$$r'_{ijkl} \coloneqq \frac{1}{Q^2} \sum_{\alpha\beta} v^{\alpha\beta}_{ijlr}$$

Then,

$$t^1_{\text{row}} = \frac{\sum_{ij} r'_{iijj} - \sum_i r'_{iiii}}{P\,(P-1)},$$

$$t^2_{\text{row}} = \frac{\sum_{ijl} r'_{iijl} - 2\sum_{ij} r'_{ijjj} - \sum_{ij} r'_{iijj} + 2\sum_i r'_{iiii}}{P\,(P-1)\,(P-2)},$$

$$t^3_{\text{row}} = \frac{\sum_{ij} r'_{ijij} - \sum_i r'_{iiii}}{P\,(P-1)},$$

$$t^4_{\text{row}} = \frac{\sum_{ijl} r'_{ijjl} - 2\sum_{ij} r'_{iiij} - \sum_{ij} r'_{ijij} + \sum_i 2r'_{iiii}}{P\,(P-1)\,(P-3)}, \quad \text{and}$$

$$t^5_{\text{row}} = \frac{\sum_{ijlm} r'_{ijlm} - 2\sum_{ijl}\left(r'_{iijl} + 2r'_{ijjl}\right) + \sum_{ij}\left(r'_{iijj} + 8r'_{ijjj} + 2r'_{ijij}\right) - 6\sum_i r'_{iiii}}{P\,(P-1)\,(P-2)\,(P-3)}.$$

Then the row-corrected estimators are given by

$$A_{\text{row}} = t^1_{\text{row}} - 2t^2_{\text{row}} + t^5_{\text{row}},$$

$$B_{\text{row}} = t^3_{\text{row}} - 2t^4_{\text{row}} + t^5_{\text{row}}, \quad \text{and}$$

$$\gamma_{\text{row}} = \frac{A_{\text{row}}}{B_{\text{row}}}.$$

Simiarly, A_{col} and B_{col} can be achieved by simply summing over the rows with the regular sum, but the columns with disjoint indices:

$$t^1_{\text{col}} = \frac{1}{P^2} \sum_{i,j} r_{iijj},$$

$$t^2_{\text{col}} = \frac{1}{P^3} \sum_{i,j,l} r_{iijl},$$

$$t^3_{\text{col}} = \frac{1}{P^2} \sum_{i,j} r_{ijij},$$

$$t^4_{\text{col}} = \frac{1}{P^3}\sum_{i,j,l} r_{ijjl}, \quad \text{and}$$

$$t^5_{\text{col}} = \frac{1}{P^4}\sum_{i,j,l,r} r_{ijlr}.$$

Note that the unbiased estimators for $\hat{\gamma}_{\text{neuron}}$ are given by redefining $v^{\alpha\beta}_{ijlr}$ as $\Phi^\top_{i\alpha}\Phi^\top_{j\alpha}\Phi^\top_{k\beta}\Phi^\top_{l\beta}$, i.e. simply compute everything with the transposed data matrix.

All the estimators can be straightforwardly implemented using the einsum operation. We use the einsum function provided in JAX.

B Reconciling Representational Similarity with Dimensionality

Consider a set of data matrices $\left\{\Phi^{(i)}\right\}_{i=1}^n$ of the same number of rows $\Phi^{(i)} \in \mathbb{R}^{P\times Q_i}\forall i$. Concatenate the matrices as:

$$\Phi_c = \begin{bmatrix}\Phi^{(1)} & \Phi^{(2)} & \dots & \Phi^{(n)}\end{bmatrix} \in \mathbb{R}^{P\times\sum_{i=1}^n Q_i}$$

Then the covariance matrix of the concatenation is given by

$$K_c = \frac{1}{\sum_{i=1}^n Q_i}\Phi_c\Phi_c^\top = \sum_{i=1}^n r_i K^{(i)},$$

$$\text{where} \quad K^{(i)} = \frac{1}{Q_i}\Phi^{(i)}\Phi^{(i)\top} \quad \text{and} \quad r_i = \frac{Q_i}{\sum_{k=1}^n Q_k}.$$

The numerator of the dimensionality of the concatenated representation is

$$\left(\frac{1}{P}\text{Tr}\left(\sum_{i=1}^n r_i K^{(i)}\right)\right)^2 = \left(\sum_{i=1}^n r_i \frac{1}{P}\text{Tr}\left(K^{(i)}\right)\right)^2,$$

and the denominator is

$$\frac{1}{P^2}\text{Tr}\left(\left(\sum_{i=1}^n r_i K^{(i)}\right)^2\right) = \frac{1}{P^2}\sum_{ij} r_i r_j \text{Tr}\left(K^{(i)}K^{(j)}\right)$$

Consider the inverse participation ratio of the eigenvalues of the covariance of the concatenated matrix:

$$\frac{1}{\gamma_{\text{joint}}} = \frac{1}{\left(\sum_{l=1}^n r_l \frac{1}{P}\text{Tr}\left(K^{(l)}\right)\right)^2}\left[\sum_{i=1}^n r_i^2 \frac{1}{P^2}\text{Tr}\left(K^{(i)2}\right) + \sum_{i\neq j} r_i r_j \frac{1}{P^2}\text{Tr}\left(K^{(i)}K^{(j)}\right)\right]$$

Note that

$$\frac{1}{P^2}\text{Tr}\left(K^{(i)2}\right) = \frac{\frac{1}{P^2}\text{Tr}\left(K^{(i)}\right)^2}{\gamma_i}, \quad \text{and} \quad \frac{1}{P^2}\text{Tr}\left(K^{(i)}K^{(j)}\right) = \frac{1}{P^2}\sqrt{\text{Tr}\left(K^{(i)2}\right)\text{Tr}\left(K^{(j)2}\right)}\text{CKA}^{(i,j)}$$

$$\frac{1}{\gamma_{\text{joint}}} = \left[\sum_{i=1}^{n}\left(\frac{r_i \frac{1}{P}\text{Tr}\left(K^{(i)}\right)}{\sum_{l=1}^{n} r_l \frac{1}{P}\text{Tr}\left(K^{(l)}\right)}\right)^2 \frac{1}{\gamma_i} + \sum_{i\neq j}\frac{r_i r_j \frac{1}{P^2}\sqrt{\text{Tr}\left(K^{(i)2}\right)\text{Tr}\left(K^{(j)2}\right)}}{\left(\sum_{l=1}^{n} r_l \frac{1}{P}\text{Tr}\left(K^{(l)}\right)\right)^2}\text{CKA}^{(i,j)}\right]$$

Let us define

$$\kappa_i = \left(\frac{r_i \frac{1}{P}\text{Tr}\left(K^{(i)}\right)}{\sum_{l=1}^{n} r_i \frac{1}{P}\text{Tr}\left(K^{(l)}\right)}\right)^2$$

Then,

$$\frac{1}{\gamma_{\text{joint}}} = \sum_{i=1}^{n}\frac{\kappa_i}{\gamma_i} + \sum_{i\neq j}\sqrt{\frac{\kappa_i\kappa_j}{\gamma_i\gamma_j}}\text{CKA}^{(i,j)}.$$

To interpret the resulting expression, κ_i simply encodes how much variance is in one representation relative to the overall variance, γ_i encodes the dimensionality, and CKA is the only term that is sensitive to the alignments of the representations.

To build intuition, consider the following example. Suppose all representations have unit variance $\frac{1}{PQ}\|\Phi\|_F^2 = \frac{1}{P}\text{Tr}\left(K^{(i)}\right) = 1$ and there are equal number of data for all representations, i.e. $Q_1 = Q_2 = \cdots = Q_n$, which are reasonable assumptions. In that case, $r_i = \frac{1}{n}$, and $\kappa_i = \frac{1}{n^2}$ for all i and therefore,

$$\frac{1}{\gamma_{\text{joint}}} = \frac{1}{n^2}\sum_{i=1}^{n}\frac{1}{\gamma_i} + \frac{1}{n^2}\sum_{i\neq j}\frac{1}{\sqrt{\gamma_i\gamma_j}}\text{CKA}^{(i,j)}.$$

Now, in addition, suppose all representations have the same dimensionality. Then,

$$\frac{1}{\gamma_{\text{joint}}} = \frac{1}{n}\frac{1}{\gamma_i} + \frac{1}{n^2}\frac{1}{\gamma_i}\sum_{i\neq j}\text{CKA}^{(i,j)}.$$

Note that the first term is of $\mathcal{O}\left(\frac{1}{n}\right)$, whereas the second term is of $\mathcal{O}\left(1\right)$ if $\text{CKA}^{(i,j)}$ is positive and finite. Therefore, if there is significant alignment amongst the presentations, i.e. $\text{CKA}^{(i,j)} = \mathcal{O}(1)\ \forall i, j$, then γ_{joint} is of $\mathcal{O}\left(1\right)$. This means that adding more representations does not blow up the joint dimensionality if there are order 1 alignments between all pairs. However, if all representations are othogonal, i.e. $\text{CKA}^{(i,j)} = 0$, then $\gamma_{\text{joint}} = n\gamma_i = \mathcal{O}(n)$.

B.1 Excess dimensionality

If all representations are aligned such that $\text{CKA}^{(i,j)} = 1$, then

$$\frac{1}{\gamma_{\text{align}}} = \left(\sum_{i=1}^{n}\sqrt{\frac{\kappa_i}{\gamma_i}}\right)^2.$$

If all representations are orthogonal such that $\text{CKA}^{(i,j)} = 0$, then

$$\frac{1}{\gamma_{\text{ortho}}} = \sum_{i=1}^{n}\frac{\kappa_i}{\gamma_i}.$$

Then the difference between γ_{joint} and γ_{align}, normalized by γ_{joint}, is given by

$$\text{ExD} = \frac{\gamma_{\text{joint}} - \gamma_{\text{align}}}{\gamma_{\text{joint}}}$$

$$= 1 - \frac{\gamma_{\text{align}}}{\gamma_{\text{joint}}}$$

where

$$\frac{\gamma_{\text{align}}}{\gamma_{\text{joint}}} = \frac{1}{\left(\sum_{i=1}^{n} \sqrt{\frac{\kappa_i}{\gamma_i}}\right)^2} \left(\sum_{i=1}^{n} \frac{\kappa_i}{\gamma_i} + \sum_{i \neq j} \sqrt{\frac{\kappa_i \kappa_j}{\gamma_i \gamma_j}} \text{CKA}^{(i,j)} \right).$$

Consider the limit where $n \to \infty$. Then, $1 - \text{ExD}$ converges to a weighted average of CKAs:

$$1 - \text{ExD} = \frac{\gamma_{\text{align}}}{\gamma_{\text{joint}}} \to \frac{\sum_{i \neq j} \sqrt{\frac{\kappa_i \kappa_j}{\gamma_i \gamma_j}} \text{CKA}^{(i,j)}}{\sum_{i \neq j} \sqrt{\frac{\kappa_i \kappa_j}{\gamma_i \gamma_j}}}.$$

There exists an expression that recovers the weighted average exactly for finite n. This expression can be written in terms of the inverse participation ratio:

$$\frac{\gamma_{\text{joint}}^{-1} - \gamma_{\text{ortho}}^{-1}}{\gamma_{\text{align}}^{-1} - \gamma_{\text{ortho}}^{-1}} = \frac{\sum_{i \neq j} \sqrt{\frac{\kappa_i \kappa_j}{\gamma_i \gamma_j}} \text{CKA}^{(i,j)}}{\sum_{i \neq j} \sqrt{\frac{\kappa_i \kappa_j}{\gamma_i \gamma_j}}}.$$

C Experimental Details

We provide all our code and necessary data to produce the figures with our supplementary material. Here, we provide additional results from our experiments.

C.1 Neural data experiments

In all our experiments, we computed the dimensionality of subsampled activations over many iterations (`Stringer`: 50, `MajajHong`: 500, `TVSD`: 250, and `ThingsFMRI`: 50).

Here, we show that the bias in our estimator arises from the nonlinear nature of the division operation used in computing dimensionality. Note that the dimensionality is defined as the ratio between A and B as in Eq. (2). In Fig. S1, the first row shows our estimator when dimensionality is calculated by averaging the ratio, $\left\langle \frac{A}{B} \right\rangle$, while the second row shows the ratio of the averaged numerator and denominator, $\frac{\langle A \rangle}{\langle B \rangle}$. While our estimator gives unbiased estimators of A and B separately, it remains biased for dimensionality.

C.2 LLM experiments

For our experiments, we use the pretrained Llama 3 base model from Grattafiori et al. (2024). We extract its hidden representations on nine different languages from the Flores dataset NLLB Team et al. (2024): `English`, `French`, `Japanese`, `Korean`, `Russian`, `Ukranian`, `Turkish`, `Kazakh`, and `Greek`.

D Bias Variance Analysis

Here we quantify the bias and variance of the γ_{naive} and γ_{both}, in the context where both the rows and columns are sampled. P and Q are the number of sampled rows and columns, respectively. We will

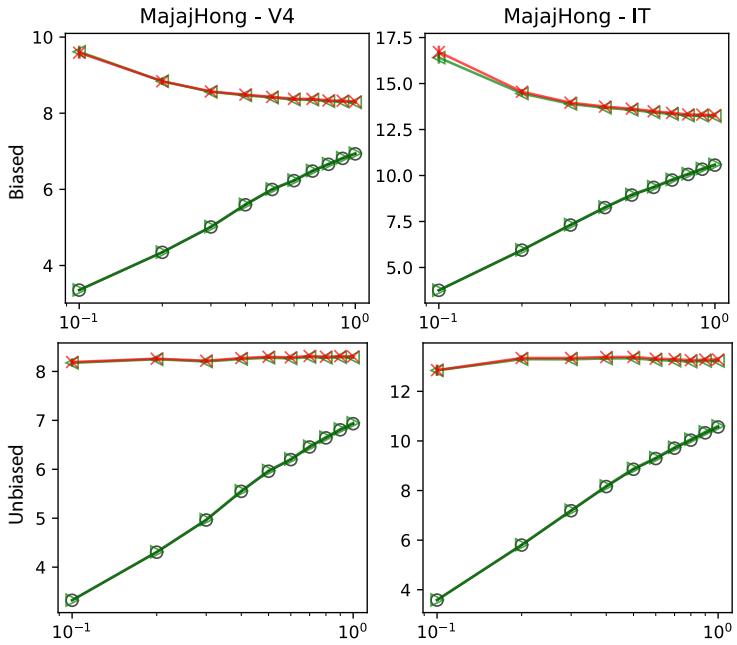

Figure S1: Bias due to nonlinearity in the definition of dimensionality.

ignore the centering operation for simplicity. We start by considering the general behavior of the ratio of estimators.

Suppose A is an unbiased estimate of a and B is an unbiased estimate of b. Suppose δ_A and δ_B are the biases of A and B, respectively, e.g. $\delta_A = \langle A \rangle - a$, σ_A^2 and σ_B^2 are the variances of A and B, respectively, and σ_{AB} is the covariance of A and B. Then the following are the bias and variance of A/B as an estimator of a/b, up to the first-order approximations:

$$\text{bias}\left(\frac{A}{B}\right) \approx \frac{a}{b}\left(\frac{1}{a}\delta_A - \frac{1}{b}\delta_B - \frac{1}{ab}\left(\sigma_{AB} + \delta_A\delta_B\right) + \frac{1}{b^2}\left(\sigma_B^2 + \delta_B^2\right)\right)$$

$$\text{var}\left(\frac{A}{B}\right) \approx \frac{a^2}{b^2}\left(\frac{1}{a^2}\sigma_A^2 + \frac{1}{b^2}\sigma_B^2 - 2\frac{1}{ab}\sigma_{AB}\right)$$

In our case, $a = \left\langle k(x,x)\right\rangle^2$ and $b = \left\langle k(x,y)^2\right\rangle$

$$\text{bias}\left(\frac{A}{B}\right) \approx \gamma\left(\frac{\delta_A}{\left\langle k(x,x)\right\rangle^2} - \frac{\delta_B}{\left\langle k(x,y)^2\right\rangle} - \frac{\sigma_{AB} + \delta_A\delta_B}{\left\langle k(x,x)\right\rangle^2\left\langle k(x,y)^2\right\rangle} + \frac{\sigma_B^2 + \delta_B^2}{\left\langle k(x,y)^2\right\rangle^2}\right)$$

$$\text{var}\left(\frac{A}{B}\right) \approx \gamma^2\left(\frac{\sigma_A^2}{\left\langle k(x,x)\right\rangle^4} + \frac{\sigma_B^2}{\left\langle k(x,y)^2\right\rangle^2} - 2\frac{\sigma_{AB}}{\left\langle k(x,x)\right\rangle^2\left\langle k(x,y)^2\right\rangle}\right)$$

D.1 Naive estimator

We have the following for the naive estimator γ_{naive}:

$$\delta_A = \frac{1}{P}\left(\left\langle k(x,x)^2\right\rangle - \left\langle k(x,x)\right\rangle^2\right) + \frac{1}{Q}\left(\left\langle \tilde{k}\left(w,w\right)^2\right\rangle - \left\langle k(x,x)\right\rangle^2\right)$$

$$\delta_B = \frac{1}{P}\left(\left\langle k(x,x)^2\right\rangle - \left\langle k(x,y)^2\right\rangle\right) + \frac{1}{Q}\left(\left\langle \tilde{k}\left(w,w\right)^2\right\rangle - \left\langle k(x,y)^2\right\rangle\right)$$

$$\sigma_{AB} = \frac{4}{P}\left(\left\langle k(x,y)^2 k(x,x)\right\rangle \left\langle k(x,x)\right\rangle - \left\langle k(x,y)^2\right\rangle \left\langle k(x,x)\right\rangle^2\right) + \frac{4}{Q}\left(\left\langle \tilde{k}(w,u)^2 \tilde{k}(w,w)\right\rangle \left\langle \tilde{k}(w,w)\right\rangle - \left\langle k(x,y)^2\right\rangle \left\langle k(x,x)\right\rangle^2\right)$$

$$\sigma_A^2 = -\frac{6}{P}\left\langle k(x,x)\right\rangle^4 + \frac{4}{P}\left\langle k(x,x)^2\right\rangle \left\langle k(x,x)\right\rangle^2 + \frac{2}{P}\left\langle k(x,x)\right\rangle^2 \left\langle k(x,y)^2\right\rangle - \frac{6}{Q}\left\langle \tilde{k}(w,w)\right\rangle^4 + \frac{4}{Q}\left\langle \tilde{k}(w,w)^2\right\rangle \left\langle \tilde{k}(w,w)\right\rangle^2 + \frac{2}{Q}\left\langle \tilde{k}(w,w)\right\rangle^2 \left\langle \tilde{k}(w,u)^2\right\rangle$$

$$\sigma_B^2 = \frac{4}{P}\left(\left\langle k(x,y)^2 k(y,z)^2\right\rangle - \left\langle k(x,y)^2\right\rangle^2\right) + \frac{4}{Q}\left(\left\langle \tilde{k}(w,u)^2 \tilde{k}(u,v)^2\right\rangle - \left\langle \tilde{k}(w,u)^2\right\rangle^2\right)$$

Plugging the above into the earlier expressions, the bias and variance of the naive estimator are given by

$$\text{bias}\left(\gamma_{\text{naive}}\right) \approx 4\gamma\left(\frac{1}{P}\left(c - c'\right) + \frac{1}{Q}\left(\tilde{c} - \tilde{c}'\right)\right) - \gamma\left(\gamma - 1\right)\left(\frac{1}{P\psi} + \frac{1}{Q\tilde{\psi}}\right), \quad \text{and}$$

$$\text{var}\left(\gamma_{\text{naive}}\right) \approx 4\frac{\gamma^2}{P}\left(\frac{1}{\psi} + c - 2c'\right) + 4\frac{\gamma^2}{Q}\left(\frac{1}{\tilde{\psi}} + \tilde{c} - 2\tilde{c}'\right) - 2\gamma\left(\gamma - 1\right)\left(\frac{1}{P} + \frac{1}{Q}\right)$$

where

$$c \coloneqq \frac{\left\langle \left\langle k(x,y)^2\right\rangle_x^2\right\rangle_y}{\left\langle k(x,y)^2\right\rangle_{x,y}^2}, \quad c' \coloneqq \frac{\left\langle k(x,y)^2 k(x,x)\right\rangle}{\left\langle k(x,y)^2\right\rangle \left\langle k(x,x)\right\rangle},$$

$$\tilde{c} \coloneqq \frac{\left\langle \left\langle \tilde{k}(w,u)^2\right\rangle_w^2\right\rangle_u}{\left\langle \tilde{k}(w,u)^2\right\rangle_{w,u}^2}, \quad \tilde{c}' \coloneqq \frac{\left\langle \tilde{k}(w,u)^2 \tilde{k}(w,w)\right\rangle}{\left\langle \tilde{k}(w,u)^2\right\rangle \left\langle \tilde{k}(w,w)\right\rangle},$$

$$\psi \coloneqq \frac{\left\langle k(x,x)\right\rangle^2}{\left\langle k(x,x)^2\right\rangle}, \quad \text{and} \quad \tilde{\psi} \coloneqq \frac{\left\langle \tilde{k}(w,w)\right\rangle^2}{\left\langle \tilde{k}(w,w)^2\right\rangle}.$$

D.2 Our estimator

We have the following for the naive estimator γ_{both}:

$$\delta_A = 0$$

$$\delta_B = 0$$

$$\sigma_{AB} = \frac{4}{P}\left(\left\langle k(x,y)^2 k(x,x)\right\rangle \left\langle k(x,x)\right\rangle - \left\langle k(x,y)^2\right\rangle \left\langle k(x,x)\right\rangle^2\right) + \frac{4}{Q}\left(\left\langle \tilde{k}(w,u)^2 \tilde{k}(w,w)\right\rangle \left\langle \tilde{k}(w,w)\right\rangle - \left\langle k(x,y)^2\right\rangle \left\langle k(x,x)\right\rangle^2\right)$$

$$\sigma_A^2 = \frac{4}{P}\left(\left\langle k(x,x)^2\right\rangle \left\langle k(x,x)\right\rangle^2 - \left\langle k(x,x)\right\rangle^4\right) + \frac{4}{Q}\left(\left\langle \tilde{k}(w,w)^2\right\rangle \left\langle \tilde{k}(w,w)\right\rangle^2 - \left\langle k(x,x)\right\rangle^4\right)$$

$$\sigma_B^2 = \frac{4}{P}\left(\left\langle k(x,y)^2 k(y,z)^2\right\rangle - \left\langle k(x,y)^2\right\rangle^2\right) + \frac{4}{Q}\left(\left\langle \tilde{k}(w,u)^2 \tilde{k}(u,v)^2\right\rangle - \left\langle k(x,y)^2\right\rangle^2\right)$$

Plugging the above into the earlier expressions, the bias and variance of our estimator are given by

$$\text{bias}\left(\gamma_{\text{both}}\right) \approx 4\gamma\left(\frac{1}{P}\left(c - c'\right) + \frac{1}{Q}\left(\tilde{c} - \tilde{c}'\right)\right), \quad \text{and}$$

$$\text{var}\left(\gamma_{\text{both}}\right) \approx 4\frac{\gamma^2}{P}\left(\frac{1}{\psi} + c - 2c'\right) + 4\frac{\gamma^2}{Q}\left(\frac{1}{\tilde{\psi}} + \tilde{c} - 2\tilde{c}'\right).$$

D.3 Comparison of the bias of γ_{naive} and γ_{both}

In summary, we have the following for the biases:

$$\text{bias}\left(\gamma_{\text{naive}}\right) = 4\gamma\left(\frac{1}{P}\left(c - c'\right) + \frac{1}{Q}\left(\tilde{c} - \tilde{c}'\right)\right) - \gamma\left(\gamma - 1\right)\left(\frac{1}{P\psi} + \frac{1}{Q\tilde{\psi}}\right) + \mathcal{O}\left(\left(\frac{1}{P} + \frac{1}{Q}\right)^2\right),$$

$$\text{and} \quad \text{bias}\left(\gamma_{\text{both}}\right) = 4\gamma\left(\frac{1}{P}\left(c - c'\right) + \frac{1}{Q}\left(\tilde{c} - \tilde{c}'\right)\right) + \mathcal{O}\left(\left(\frac{1}{P} + \frac{1}{Q}\right)^2\right)$$

Note that the first terms $4\gamma\left(\frac{1}{P}\left(c - c'\right) + \frac{1}{Q}\left(\tilde{c} - \tilde{c}'\right)\right)$ in $\text{bias}\left(\gamma_{\text{naive}}\right)$ is entirely the leading order term of $\text{bias}\left(\gamma_{\text{both}}\right)$. This means that the second term $\gamma\left(\gamma - 1\right)\left(\frac{1}{P\psi} + \frac{1}{Q\tilde{\psi}}\right)$ is the additional bias in γ_{naive}. Here we show that the first term is often small, and the second term tends to be large, which is removed in $\text{bias}\left(\gamma_{\text{both}}\right)$. Let us inspect the term $c - c'$ that contributed to the bias in γ_{both}:

$$c - c' = \frac{\left\langle \left\langle k(x,y)^2\right\rangle_x^2\right\rangle_y}{\left\langle k(x,y)^2\right\rangle_{x,y}^2} - \frac{\left\langle k(x,y)^2 k(x,x)\right\rangle_{x,y}}{\left\langle k(x,y)^2\right\rangle_{x,y}\left\langle k(x,x)\right\rangle_x},$$

If $\left\langle k(x,y)^2\right\rangle_x$ is constant for all y, then $c = 1$. This is also when $c' = 1$. Therefore, when $\left\langle k(x,y)^2\right\rangle_x$ is constant,

$$c - c' = 0.$$

Note that $\left\langle k(x,y)^2\right\rangle_{x\sim\rho_{\mathcal{X}}}$ is constant for all y, if the kernel and data distribution are matched by a global symmetry (rotational or translational). For example, consider any dot-product kernel of the form

$$k(x,y) = h\left(x \cdot y\right)$$

where $h : \mathbb{R} \to \mathbb{R}$, with x and y sampled uniformly from a unit sphere. Then $c - c' = 0$, so the bias in our estimator is dramatically smaller than that of the naive estimator. Similarly, consider any translation-invariant kernel of the form

$$k(x, y) = h(x - y)$$

with x and y sampled from a translation-invariant distribution, e.g. uniform measure on a torus. Then $c - c' = 0$.

E LOCAL DIMENSIONALITY ESTIMATION

The sample neural manifold is simply the set of row vectors of the sample matrix Φ:

$$\mathcal{M} \coloneqq \{\bar{\phi}_i\}_{i=1}^{P}$$

where $\bar{\phi}_i \in \mathbb{R}^Q$ is the ith row vector of Φ. Suppose the distance metric defined in the sample representation space $\mathbb{R}^Q$ is denoted as $d(\cdot, \cdot)$. Define the ball in the representation space centered around $\bar{\phi}_0 \in \mathbb{R}^Q$ with radius r:

$$\mathcal{B}(\bar{\phi}_0, r) \coloneqq \left\{\bar{\phi} \in \mathcal{M} \mid d(\bar{\phi}, \bar{\phi}_0) \leq r\right\}.$$

Let $\gamma_{\text{both}}(\mathcal{S})$ denote a dimensionality measured on a finite set $\mathcal{S}$ of sample representations in $\mathbb{R}^Q$. Then, the local dimensionality estimate for a radius r is defined as

$$\gamma_{\text{both}}^{\text{local}}(r) = \frac{1}{|\mathcal{M}|} \sum_{\bar{\phi}_0 \in \mathcal{M}} \gamma_{\text{both}}\left(\mathcal{B}(\bar{\phi}_0, r)\right).$$

Now, let us define the distance metric. For two representation vectors $\bar{\phi}, \bar{\phi}_0 \in \mathbb{R}^Q$, the squared Mahalanobis distance is defined as:

$$d^2(\bar{\phi}, \bar{\phi}_0) = \left(\bar{\phi} - \bar{\phi}_0\right)^\top M \left(\bar{\phi} - \bar{\phi}_0\right)$$

where M is a positive-definite metric. We want to preferably select representation along the tangent directions of $\bar{\phi}_0$ so that we faithfully capture the local dimensionality. We estimate a local representation metric by $\Sigma\left(\bar{\phi}_0\right)$, the covariance of the k-nearest neighbors of $\bar{\phi}_0$. Then we define M as $\Sigma\left(\bar{\phi}_0\right)^\dagger$, the pseudoinverse of $\Sigma\left(\bar{\phi}_0\right)$.