Manifold geometry underlies a unified code for category and category-independent features

Lorenzo Tiberi[1] and Haim Sompolinsky[1,2]

[1] *Center for Brain Science, Harvard University, Cambridge, MA 02138*

[2] *Edmond and Lily Safra Center for Brain Sciences, Hebrew University, Jerusalem 9190401, Israel*

In everyday vision, animals routinely extract from the same visual stimulus both object identity and continuous identity-independent variables such as position and size. It has been shown that linear decoding performance of both kinds of information increases along the ventral stream, suggesting that inferior temporal cortex may be implementing a joint code for object category and category-independent features. A central open question is whether such a code can indeed exist within a single representation and, if so, what geometric properties enable it. Here, we show that convolutional neural networks can develop such a code. We then derive a theory of regression on category manifolds, identifying the key manifold-geometry measures that enable accurate readout of category-independent features, and showing how they can be optimized while preserving manifold properties known to support classification performance. We further characterize how common experimental constraints, such as subsampling neural units and using a limited number of categories, affect the empirical estimation of regression performance. Our findings thus provide a principled understanding of the geometry underlying joint codes and yield testable predictions for future neural recordings probing the joint-code hypothesis in the ventral stream.

I. INTRODUCTION

Natural intelligence often relies on extracting multiple kinds of information from the same sensory stimulus. In everyday vision, for example, animals routinely extract both object identity and continuous, identity-independent attributes such as position and size from the same visual stimulus. A central question in neuroscience is how the brain implements this ability.

Neural recordings in higher-order areas, such as macaque prefrontal cortex, anterior cingulate cortex, and hippocampus [1, 2], show that multiple task-relevant variables can be decoded from the same population activity using simple (often linear) readouts. This has led to the hypothesis that a single high-level population code can jointly encode multiple stimulus variables in a linearly accessible form, raising the question of which representational principles make such multi-variable encoding possible [3]. Existing theoretical accounts have mainly addressed highly controlled regimes with low-dimensional stimulus spaces and task variables [1, 2, 4–7], leaving open the question of how multi-variable decodability is achieved in regimes with many stimulus categories and high-dimensional within-category variability spanning both category-specific latent factors and multiple category-invariant features.

A natural setting to study this regime is vision, where single-object images within a category can vary widely across a range of category-preserving transformations. Indeed, it has long been suggested that neural representations in primate higher-order visual areas encode not only object category, but also category-independent features [8–10], including object size [11, 12], texture and material properties [13–15], shape properties [16, 17], and facial properties [18, 19]. A seminal study [20] systematically tested the hypothesis of a joint population code by showing that, along the macaque ventral stream (V1→V4→inferior temporal cortex, IT), the linear decodability of object category *and* the linear regressibility of category-independent features, such as object position, size, and pose within the image, both increase. Yet, despite these improvements along the visual hierarchy, our analysis shows that even in IT cortex the absolute regression performance as estimated from the macaque recordings remains limited, raising the question of whether this reflects experimental limitations—such as population subsampling and limited image sets—or instead a genuine constraint on how ventral-stream populations can jointly encode object category and category-independent features.

To address this question, convolutional neural networks (CNNs) provide a useful testbed because they align well with ventral-stream neural responses [21, 22] and can be systematically investigated without the experimental limitations of neural recordings. Recent work [23] suggested that CNNs can indeed be jointly optimized for object classification and regression of category-independent features, indicating that a joint population code may at least be possible in artificial systems. Here we address the following fundamental question: What properties of a population representation enable effective joint encoding of category and category-independent information?

Identifying these distinctive properties requires a theoretical framework that relates linear classification and linear regression performance to properties of neural representations. For object classification, such a framework already exists in terms of object manifolds, defined as the set of population responses elicited by images from the same category. Theoretical work relates aspects of manifold geometry, such as radius and dimensionality,

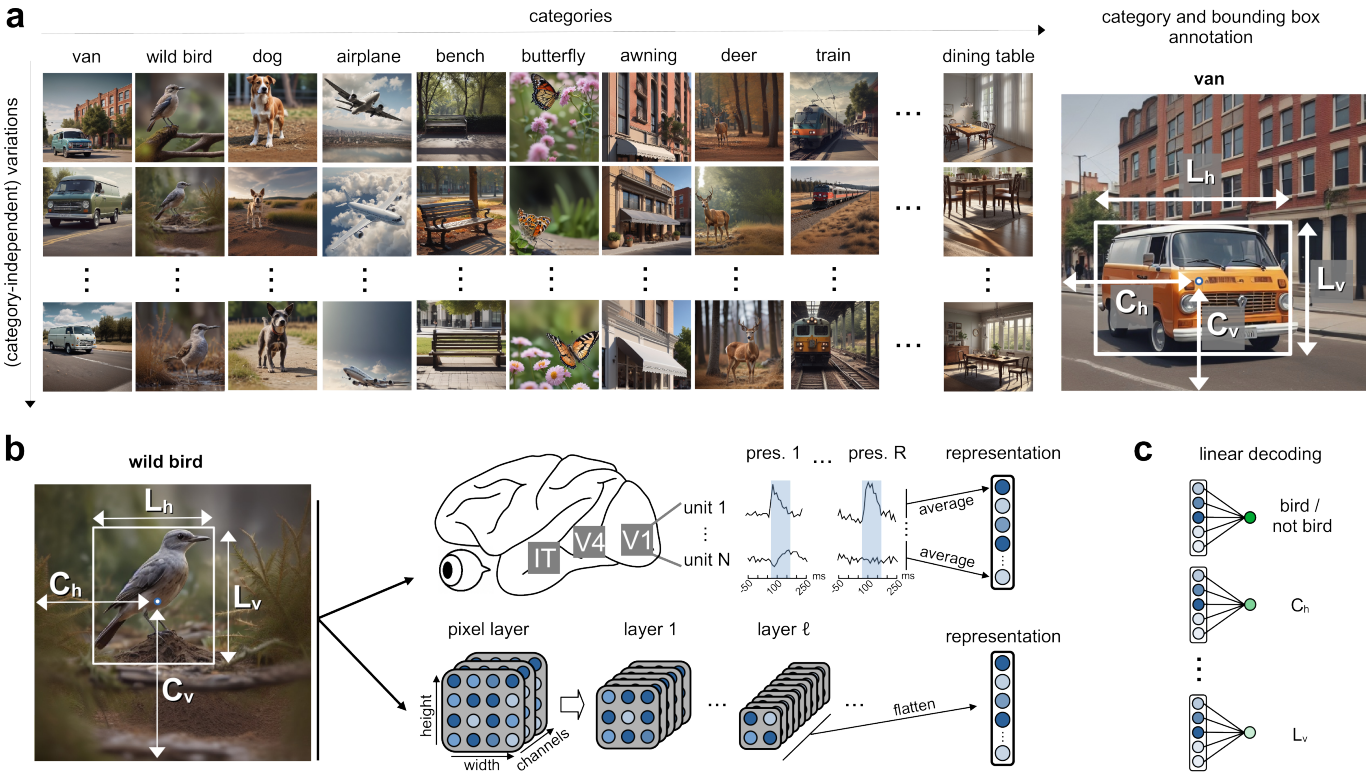

Figure 1. Framework to quantify how well neural representations encode category and category-independent information. (**a**) Overview of the main image dataset used in this work. The dataset contains single-object images annotated with object category and four bounding box coordinates C_h, C_v, L_h, L_v. (**b**) From image to neural representation. An image is presented to the ventral visual stream (top), modeled in this work by a convolutional neural network (CNN; bottom), and the resulting neural representations are analyzed. Top: Representations from neural recordings are obtained by averaging spike counts over a post-timulus time window (shaded region) and across stimulus repetitions. Bottom: Representations from a given CNN layer are obtained by flattening that layer's activations into a one-dimensional vector. (**c**) We evaluate the ability of linear decoders to extract category and bounding-box coordinates from these representations (see the Online Methods for details).

to proxy measures of linear classification performance, including storage capacity [24] and few-shot generalization [25]. These approaches have been used to interpret trends in classification performance along the hierarchy of both the macaque ventral stream and artificial CNNs [25, 26]. In contrast, an analogous theory linking manifold geometry to the regression performance of category-independent features is currently lacking. Crucially, because our goal is a theory of joint coding, it is not sufficient to develop a regression theory in a setting disconnected from categorization. Instead, the theory should be embedded within the category manifold framework, so that it can address how geometric properties relevant for regression of category-independent features relate to—and potentially trade off with—those supporting object classification. Here we develop such a theory and test it in CNNs trained under different objectives, by analyzing the resulting representations and task performance.

In this work, we first show that a ventral-stream–like CNN can be trained so that the feature layer representations support high-accuracy linear readout of both object category and category-independent features, such as object position and size. We then develop a manifold-geometry theory for linear regression of category-independent features that decomposes the global regression error into a "local" error, which quantifies how well a feature is linearly encoded within individual category manifolds, and a "local–global" error gap, which captures the additional error incurred by requiring a single category-independent linear readout shared across categories. We then identify the across-category differences in manifold geometry that contribute to this gap. We further investigate how this regression-relevant manifold geometry can coexist with the manifold geometry that supports linear object classification. Finally, we use this framework to identify signatures that distinguish a true joint code from a category-only code and to quantify how common experimental constraints (limited categories and neural-unit subsampling) can obscure those signatures.

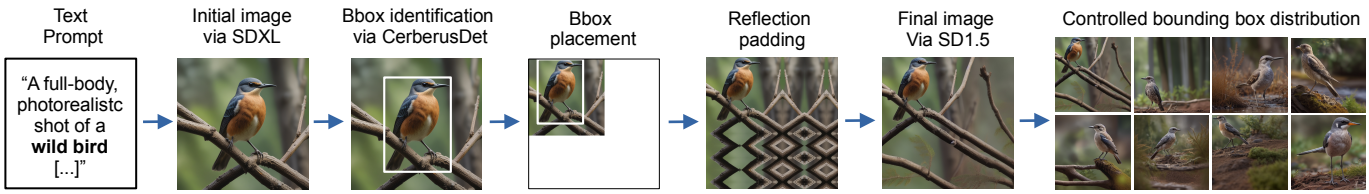

Figure 2. Generation process for an image in our dataset. From left to right: an initial image of an object in the desired category (e.g. "wild bird") is generated from a text prompt using Stable Diffusion XL [27, 28], with categories drawn from the taxonomy of Objects365 [29], an image dataset for object detection. The object's bounding box is then estimated by CerberusDet [30], an Objects365-trained detector. The image is then isotropically rescaled and pasted onto a white canvas so that the detected box matches a target box whose parameters (C_h, C_v, L_h, L_v) have been drawn uniformly (conditional on preserving aspect ratio and remaining inside the canvas; see Online Methods and SI). The remaining white area is filled by reflection-padding the border region around the pasted image, and the resulting canvas is used as the conditioning image for a second round of generation with a Stable Diffusion 1.5 inpainting model [31–33], yielding the final sample. The method allows generation of images with a controlled distribution of bounding boxes. For example images from other categories, see Fig. 1a and the SI.

II. RESULTS

1. *Linear decoding framework*

To assess how well neural representations encode both an image's object category and features that vary independently of category ("category-independent" features), we adopt the linear-decoding framework of [20] (**Fig. 1**).

We consider image datasets consisting of single-object images labeled by category and by four continuous bounding-box parameters describing object position and size: center coordinates C_h, C_v and side lengths L_h, L_v along the image's horizontal (h) and vertical (v) axes (**Fig. 1a**).

For each image, an N-dimensional neural representation vector is extracted from the activations of neurons in the brain visual hierarchy or from the activations of a CNN that mimics this hierarchy (**Fig. 1b**).

Finally, we evaluate the performance of linear decoders in extracting category and category-independent information from these representations, treating it as a proxy for the decoding performance achieved by hypothetical downstream neurons (**Fig. 1c**). For category, we train one-vs-rest classifiers and evaluate their balanced classification accuracy, averaged across category. For the bounding box parameters, we train one linear regressor per parameter and evaluate it by normalized mean squared error (nMSE), i.e., MSE divided by the parameter variance. All decoding metrics are cross-validated over multiple train–test splits.

2. *One code supports both classification and regression*

We have applied this decoding framework to the publicly available dataset of macaque recordings [34] studied in [20], reproducing their qualitative results (see SI): decoding performance of both category and category-independent features improves along the visual hierarchy. However, we also find that absolute regression performance remains limited: root nMSE remains above 0.5, meaning that the root-mean-square prediction error is still larger than half the across-image standard deviation of the decoded feature.

To investigate what it takes to accurately perform both categorization and regression tasks we apply the above decoding framework to CNNs trained on a newly constructed large-scale dataset of single-object images with category and bounding-box annotations (**Fig. 1a**). Images span many object categories (265) as well as many images per category (20,000 per category: 10,000 for training the CNN and 10,000 for generating neural representations and evaluating linear decoders). The large number of images and categories serves several purposes: it enables effective training, a more precise statistical characterization of neural representations, and it places us in a regime closer to the limit of infinite number of categories, which is valuable for testing theoretical predictions and serves as a clean baseline from which we can later introduce controlled category subsampling. Images were generated with a multi-stage pipeline that uses diffusion models [27, 31] to produce photorealistic single-object images with a controlled distribution of bounding-box parameters (**Fig. 2**; Online Methods).

On this dataset, we fine-tuned an ImageNet-1k–pretrained ResNet-50 (**Fig. 3a**), a state-of-the-art CNN with good alignment to ventral stream responses [22]. We treated the final feature layer as the shared representation and attached two types of readout: (i) the standard linear classification head for predicting object category, and (ii) four additional linear regression heads, each predicting one bounding-box parameter (C_h, C_v, L_h, L_v). The network was optimized end-to-end using stochastic gradient descent on a combined loss: cross-entropy loss on the classification head and Smooth L1 loss on each regression head (Online Methods). In addition to this model, which we call network CR, we also trained the same architecture exclusively on the classification task (network C) and exclusively on the regression task (network R).

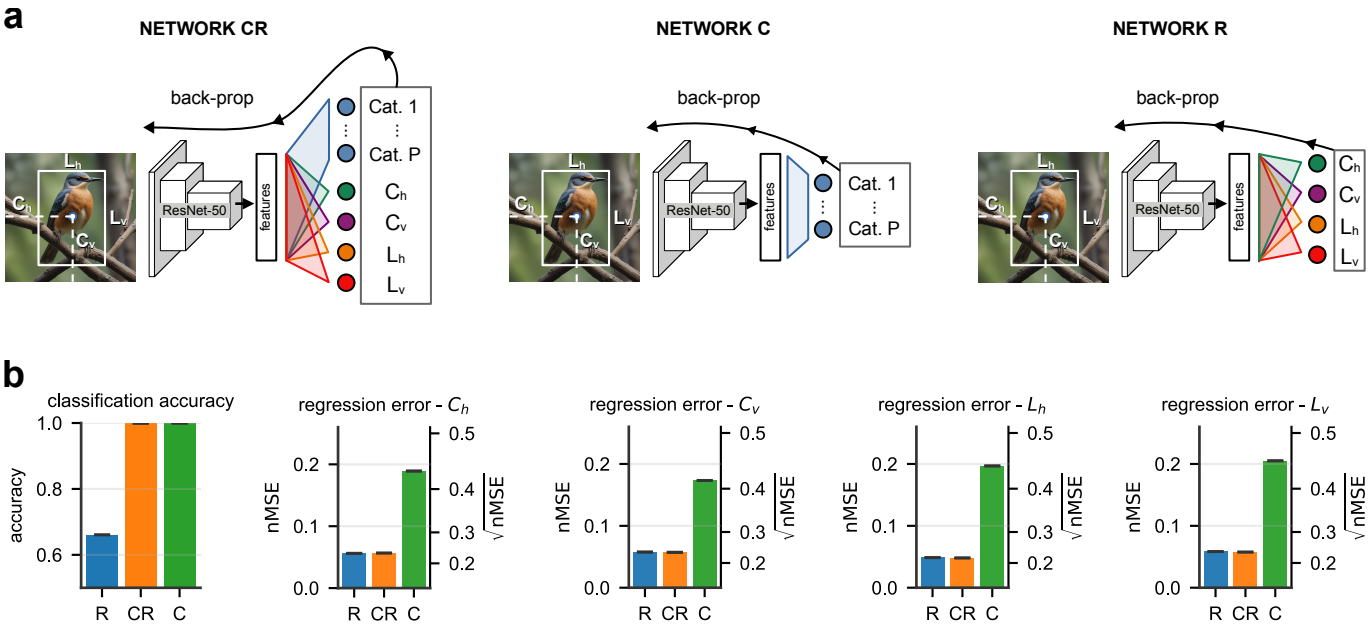

Figure 3. One code supports both object classification and regression of category-independent features. (**a**) The three CNN variants trained on our image dataset. Network CR consists of a ResNet-50 backbone whose feature layer feeds both the standard classification head and four regression heads, one for each bounding-box coordinate. It is trained jointly on the object classification and bounding-box regression tasks. The full network, including the shared backbone, is trained end-to-end by stochastic gradient descent with error back-propagation through all components. By contrast, network C is trained only for object classification and therefore includes only the classification head, while network R is trained only to regress the bounding-box coordinates and therefore includes only the four regression heads. (**b**) Results of linearly decoding object category and bounding-box coordinates from the feature layer of networks R, CR and C. Error bars indicate the SEM across cross-validation splits. For regression, we report for convenience both the nMSE on the left y-axis, and the corresponding root, $\sqrt{\text{nMSE}}$, on the right y-axis.

The C and R networks provide upper bounds on the performance achievable on this dataset by this architecture, when it is allowed to specialize in a single task.

Fig. 3b shows the performance of linear decoders trained on the feature-layer representations of these three networks. Network CR matches the performance of network C on classification and that of network R on regression. We thus conclude that the CR network implements a single population code that jointly supports optimal linear readout of object category and category-independent features.

3. *Theory*

What properties of the population code of network CR, or of any population code in general, allow this optimal performance in both classification and regression of category-independent features? For classification, theories already exist that relate neural representation geometry to proxy measures of linear classification performance, such as storage capacity [24], or few-shot learning accuracy [25]. Such theories are formulated within the framework of object manifolds, defined as the collection of all neural representations of images belonging to the same category (**Fig. 4a**). It has been shown that along the hierarchy of both classification-trained CNNs [26] and the ventral visual stream [25], such object manifolds change their geometric properties, such as their dimensionality, radius, and centroid separation, so as to become more and more disentangled, thereby improving linear classification performance (**Fig. 4a**). In a system optimized also for regression, one could imagine a similar scenario whereby, along the visual hierarchy, the encoding of category-independent features becomes both more linear within each manifold and more consistently aligned across manifolds, such that it can be regressed accurately by a single category-independent regressor (**Fig. 4a**). Unlike for classification, however, a theory identifying and linking the relevant manifold geometry to linear regression performance is currently lacking. We now develop such a theory.

We begin by distinguishing two additive contributions, of very different nature, that determine linear regression performance for category-independent features (**Fig. 4b-c**). Let E denote the test MSE obtained when linearly regressing a given category-independent feature with a single category-independent readout. We decompose it

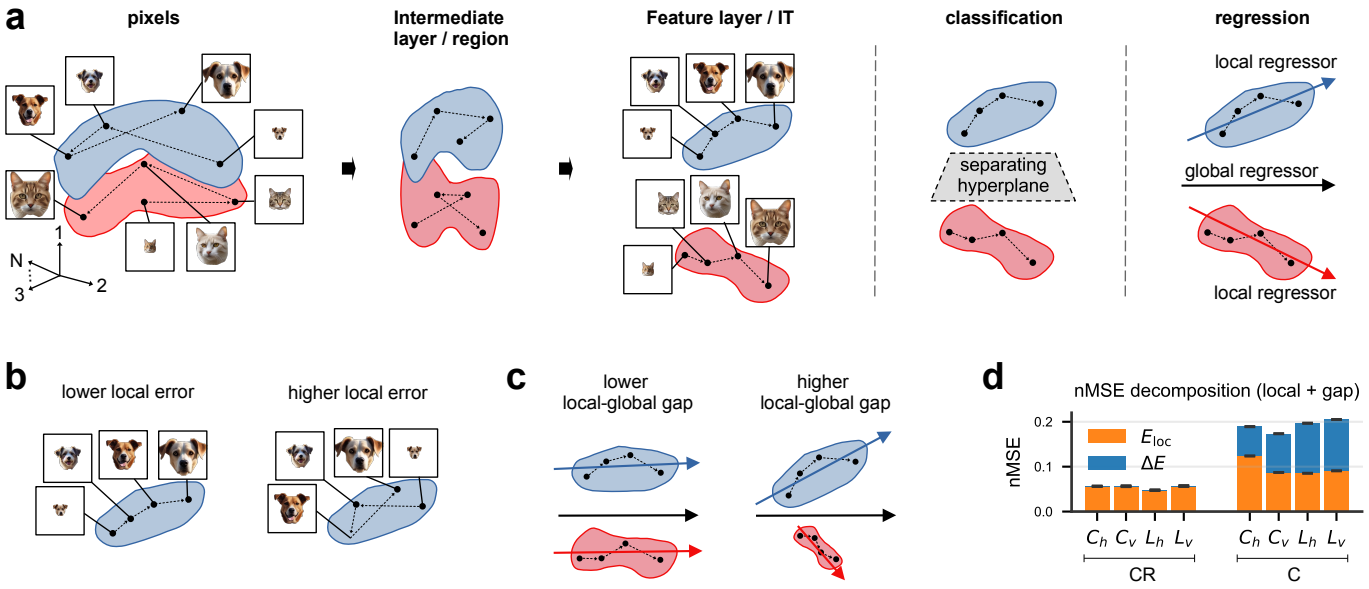

Figure 4. Theoretical framework. (**a**) An object manifold is defined as the collection of all neural representations of images belonging to the same category. We illustrate a "dog" manifold (blue) and a "cat" manifold (red). Each image is mapped to a point (black dot) in the N-dimensional space of neural units. Variability within a manifold arises from both category-dependent changes (e.g., different dog or cat exemplars) and category-independent changes, here schematically represented by variation in object size (arrows connect representations in order of increasing size). Along the hierarchy of a system optimized for both object classification and regression of category-independent features, manifolds become more geometrically disentangled (improving linear classification) and category-independent features become more linearly encoded (improving linear regression). (**b**) The local regression error $E_{\rm loc}$ depends on how linearly the feature is encoded within individual object manifolds. In the left manifold, object size is encoded more linearly than in the right manifold, leading to a lower $E_{\rm loc}$. (**c**) The local-global error gap ΔE depends on the geometric arrangement of object manifolds across categories. Along the regressed feature, the manifolds on the left are both more aligned and more similar in scale than the manifolds on the right, resulting in a lower ΔE. (**d**) Decomposition of the nMSE into local error $E_{\rm loc}$ (orange) and local–global gap ΔE (blue) for regression of the four bounding box coordinates from the feature layer of networks C and CR. Note that ΔE is very small and barely visible for network CR.

as

$$E = E_{\rm loc} + \Delta E \tag{1}$$

where $E_{\rm loc} = {\rm Avg}_\mu \left[E^\mu\right]$ and E^μ is the test MSE achieved by a regressor trained and tested exclusively on the μ-th category. The "local error" $E_{\rm loc}$ quantifies the quality of the feature's linear encoding at the *local* level of a single category, i.e. within a single object manifold (**Fig. 4b**). It provides a lower bound on the global error E, corresponding to the regression error of an oracle that is allowed to use separate regressors for each category. The residual ΔE, termed the "local-global error gap", captures the additional error incurred by requiring a single category-independent readout, and reflects the global geometric arrangement of object manifolds across categories in the N-dimensional space of neural units (**Fig. 4c**).

Fig. 4d shows the decomposition $E = E_{\rm loc} + \Delta E$ for the MSE of regressing object bounding box coordinates from the feature layer of networks C and CR. Relative to network C, network CR improves both the within-manifold linear encoding of the feature, captured by $E_{\rm loc}$, and the across-manifold organization of this encoding, reflected by ΔE. The dominant change is in ΔE, which drops by orders of magnitude in network CR, indicating a coordinated reorganization of manifold geometry across categories that makes regression by a single category-independent readout markedly more effective. Notably, in the SI we show that ΔE is largely reduced not only in network CR, which is optimized for *category-independent* regression and classification, but also in a network that we term CRloc, which we optimized for *category-specific* regression and classification. In CRloc, category-specific regression is implemented by attaching multiple regression heads to the feature layer—one for each category—rather than a single head shared across categories. This result indicates that optimization for regression alone is sufficient to substantially reduce the local-global gap, even in the absence of the additional constraint imposed by a shared readout across categories.

To gain insights into the manifold-geometry reorganization reflected by ΔE, we develop a theory that predicts ΔE from readily interpretable geometric measures. Under fairly general assumptions on manifold statistics, in the limit of infinitely many categories, and neglecting higher-order corrections in the number N of neurons, we

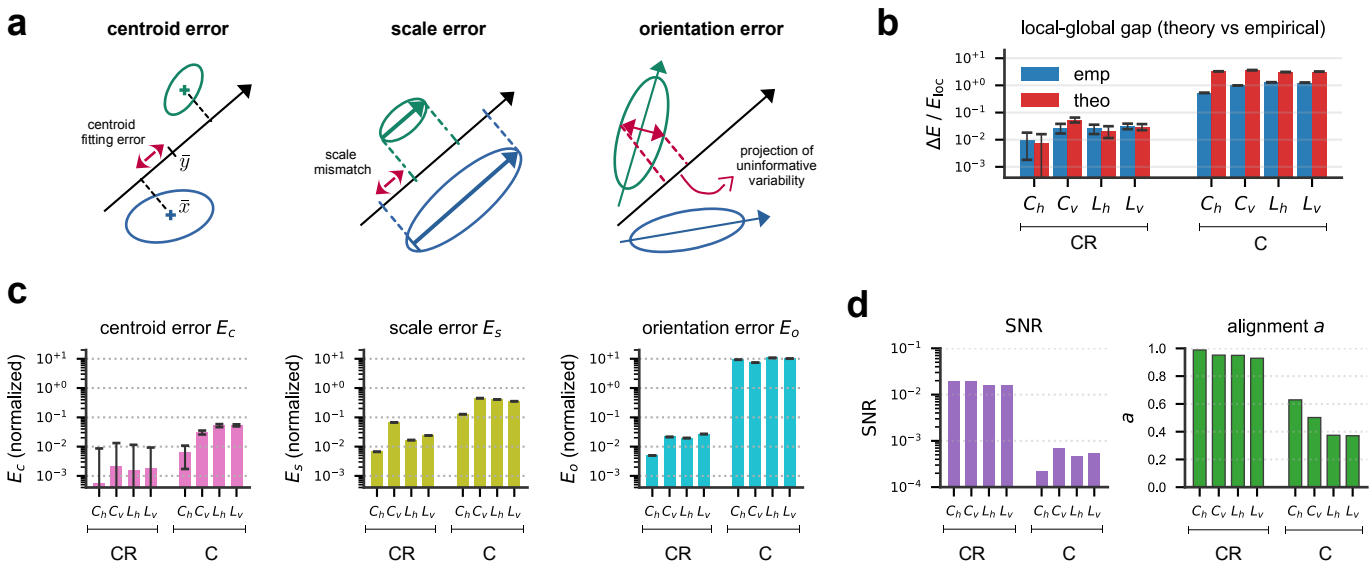

Figure 5. Theory for the local-global error gap. (**a**) Schematic illustration of the three sources of error (E_c, E_s, E_o) contributing to ΔE. Left: The centroid error E_c reflects the error in fitting the manifold centroids. Center: The scale error E_s arises from variability in the scale of the local feature encoding across categories. Right: The orientation error E_o arises from misalignment of the local feature-encoding direction across categories, causing the projection of uninformative manifold variability onto the global regressor. (**b**) Measured relative local-global gap $\Delta E/E_\mathrm{loc}$ (blue) compared to our theory prediction (red). (**c**) Decomposition of the relative gap $\Delta E/E_\mathrm{loc}$ into the contributions from the centroid error E_c (pink) scale error E_s (olive) and orientation error E_o (cyan). All three errors are normalized by dividing by E_loc, and scale and orientation error are further multiplied by $\hat{\sigma}^2$. This normalization allows for a direct comparison of the three errors, since, to linear order, $\Delta E \sim E_\mathrm{c} + \hat{\sigma}^2 (E_\mathrm{s} + E_\mathrm{o})$. (**d**) The two elements contributing to the orientation error: SNR, and alignment a of the local feature-encoding directions.

show (see Online Methods and SI) that

$$\Delta E = E_\mathrm{c} + \hat{\sigma}^2 \bar{\Delta E}, \qquad \bar{\Delta E} = 1 - \frac{1}{(1+E_\mathrm{s})(1+E_\mathrm{o})} \quad (2)$$

where E_s, E_o and E_c represent three distinct sources of error:

1. **Centroid error (E_c).** The term E_c reflects the error in fitting the manifold centroids (**Fig. 5a**, left). Each category μ has a centroid (mean representation) $\bar{x}^\mu \in \mathbb{R}^N$ and a mean feature value $\bar{y}^\mu \in \mathbb{R}$. A single category-independent regressor must be able to fit the mapping $\bar{x}^\mu \to \bar{y}^\mu$. Operationally, E_c is the part of ΔE that would be eliminated by centering each category manifold (and the corresponding feature labels) around its category mean before fitting and evaluating the category-independent regressor (Online Methods). The remaining term $\bar{\Delta E}$ represents the error due to fitting manifold variability around the centroids.

2. **Scale error (E_s).** The scale error E_s reflects an across-category mismatch in the scale at which the feature is encoded within each manifold (**Fig. 5a**, center). Let $W^\mu \in \mathbb{R}^N$ denote the optimal local readout vector regressing the feature in category μ, and define the scale of the local encoding as $\|W^\mu\|$. We then define

$$E_\mathrm{s} = \frac{\mathrm{Var}_\mu\left[\|W^\mu\|\right]}{\mathrm{Avg}_\mu\left[\|W^\mu\|\right]^2}, \quad (3)$$

which is the squared coefficient of variation of $\|W^\mu\|$ across categories (and vanishes only when all categories encode the feature with the same scale).

3. **Orientation error (E_o).** The orientation error E_o reflects across-category misalignment of the local encoding directions $\hat{w}^\mu = W^\mu / \|W^\mu\|$. Misalignment causes within-manifold variability that is uninformative for the feature to project onto the global category-independent regressor, increasing error (**Fig. 5a**, right). We obtain

$$E_\mathrm{o} = \frac{1-a}{N \cdot a \cdot \mathrm{SNR}} \quad (4)$$

where $a = \left\|\mathrm{Avg}_\mu\left[\hat{w}^\mu\right]\right\|^2$ quantifies the alignment of the local encoding directions ($a = 0$: no net alignment; $a = 1$: perfect alignment) and $\mathrm{SNR} = \mathrm{Avg}_\mu\left[\hat{w}^{\mu\top}\Sigma^\mu\hat{w}^\mu\right] / \mathrm{Avg}_\mu\left[\mathrm{Tr}\Sigma^\mu\right]$ measures the signal-to-noise ratio between the (informative) variance along the local encoding direction and the total (uninformative) manifold variance. Here $\Sigma^\mu \in \mathbb{R}^{N\times N}$ is the centered covariance matrix of representations belonging to the μ-th manifold.

The N^{-1} scaling reflects a high-dimensional advantage: in larger representational spaces, uninformative within-manifold variability is increasingly likely to be nearly orthogonal to the global regressor and thus to contribute less readout noise, allowing a fixed E_{o} to be attained with smaller alignment a or SNR.

Finally, $\hat{\sigma}^2$ is the variance of the feature explained by the optimal local linear readout, averaged across categories, and converts the dimensionless gap $\bar{\Delta E}$ into MSE units.

Fig. 5b compares our theoretical prediction with the measured local–global error gap ΔE. The theory captures well the orders-of-magnitude reduction in ΔE achieved by network CR compared to network C. **Fig. 5c** further decomposes the theoretical prediction of ΔE into contributions from the centroid error E_{c}, scale error E_s, and orientation error E_{o}. In both networks, the centroid error is always negligible compared to the other errors. In network CR, both E_{s} and E_{o} are substantially reduced to comparably small values, with neither term dominating. In contrast, the poor regression performance of network C is primarily driven by a large orientation error E_{o}. Finally, **Fig. 5d** shows the two factors contributing to the orientation error E_{o}: the SNR and the alignment index a. Both are improved in network CR.

4. Manifold geometry reveals optimization strategy

In comparing network C to CR, we have shown how the latter optimizes the manifold geometry relevant for regression. But how is this optimization achieved while preserving the same classification accuracy achieved by network C? Our theory illuminates how this is possible: because regression-relevant geometry is mostly related to a specific direction within the manifold—the feature-encoding direction—rather than the manifold as a whole, it can be optimized with only marginal effects on classification-relevant geometry such as manifold radius, dimensionality and centroid separation.

We begin illustrating this by considering manifold shapes, which are directly related to classification-relevant geometry such as manifold radius and dimensionality [24, 25]. One example of a regression-relevant measure whose optimization could affect manifold shape is SNR. However, note that SNR can be increased via two complementary strategies (**Fig. 6a**): (i) increasing variance along the feature-encoding direction (and/or reducing variance in directions orthogonal to it), thereby reshaping the manifold; or (ii) re-encoding the feature along more dominant principal-component (PC) directions, thereby increasing variance along the encoding direction without affecting manifold shape. We find that network CR implements strategy (ii) almost exclusively, thereby preserving the manifold shapes found in the classification-optimized network C: manifold PC spectra—used here as an estimate of manifold shapes—are essentially unchanged relative to network C (**Fig. 6b**), while the overlap of the local feature-encoding direction with the top PCs is substantially increased (**Fig. 6c**).

Similarly to manifold shapes, we also find that manifold centroids—the other key geometric property relevant for classification performance—show only modest changes in norm and relative distance compared to network C (see SI).

Beyond shape and centroid position, manifolds also have an orientation. We therefore consider alignment between manifolds, which we quantify by an alignment matrix A whose entry $A_{\mu\nu} \in [0,1]$ is large when manifolds μ and ν are aligned, i.e., their axes of variability (PCs) point in similar directions and carry comparable variance (see **Fig. 6** caption for the formal definition). We have discussed how regression optimization increases a form of direction-specific alignment, namely the alignment a of the local feature-encoding directions. Its effect on the overall manifold alignment A is, however, nontrivial, because a could be increased via two complementary strategies (**Fig. 6d**): (1) rotating (in the sense of re-encoding) the feature-encoding direction *within* each manifold, without changing manifold orientation and thus global manifold alignment; or (2) rotating the manifolds themselves while keeping the feature-encoding direction fixed relative to their associated manifold, thereby increasing global manifold alignment. While we have observed a mechanism compatible with option (1) in our analysis of manifold shapes (**Fig. 6c**), we find that network CR partly implements also option (2), as reflected by a clear, though modest, increase in the average entry of A compared to network C (**Fig. 6e**). Interestingly, while in network C the alignment matrix A exhibits a block structure that mirrors the semantic organization of categories, in network CR this structure is largely obscured by the overall increase of alignment across all categories (**Fig. 6f**).

More generally, these results show that regression optimization can leave only a subtle signature on coarse measures of manifold geometry, such as manifold shape, centroid position, and mutual alignment. This helps explain why networks C and CR exhibit highly similar representations under widely used similarity measures such as centered kernel alignment [35] (CKA; ~ 0.95, see SI), as well as similar alignment scores with neural data (see SI), and may likewise account for analogous results reported previously [23]. Our results therefore highlight the importance of finer geometric measures—such as those identified by our theory—that can distinguish a true joint code even when coarser representational-similarity measures cannot.

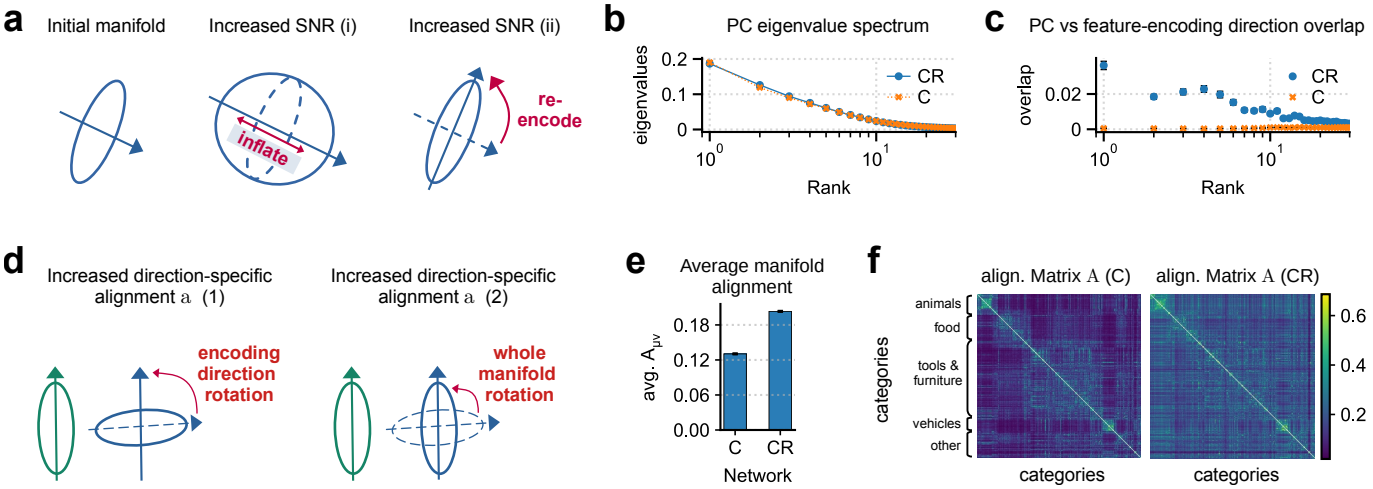

Figure 6. Geometry optimization strategy. (**a**) Two complementary strategies to increase SNR: (i) Inflate manifold variance along the feature-encoding direction; or (ii) re-encoding the feature-encoding direction towards more dominant manifold PCs. (**b**) PC eigenvalue spectrum (eigenvalue vs rank) for category manifolds in network C (orange) and CR (blue), averaged across categories (error bars are SEM). For each manifold μ, the eigenvalues were normalized by the manifold total variance $\mathrm{Tr}\Sigma^{\mu}$, averaged across networks C and CR. Analogous per-category comparisons of the PC spectra are provided in the SI. (**c**) Squared cosine angle between the feature-encoding direction $\hat{w}^{\mu}$ and the PC directions of manifold μ (ordered by rank) for networks C (orange) and CR (blue), averaged across categories (error bars are SEM). We show the case of feature L_{h} (see SI for other features). (**d**) Two complementary ways of increasing the alignment a between feature-encoding directions: (1) rotating (in the sense of re-encoding) the feature-encoding direction *within* the manifold, without rotating the manifold, or (2) rotating the manifold itself, thereby rotating the feature-encoding direction with it. (**e**) Average of the off-diagonal entries of A, for networks C and CR. Matrix entries are defined as $A_{\mu\nu} = \mathcal{N}^{-\frac{1}{2}}\mathrm{Tr}\left(\Sigma^{\mu}\Sigma^{\nu}\right)$, with Σ^{μ} the covariance of manifold μ, and $\mathcal{N} = \mathrm{Tr}\left(\Sigma^{\mu 2}\right)\mathrm{Tr}\left(\Sigma^{\nu 2}\right)$. (**f**) Manifold alignment matrix A, for network C (left) and network CR (right). Categories are ordered (and grouped) by semantic similarity; macro-groups are indicated on the left.

5. *Decoding performance along the visual hierarchy*

To understand how the linear encoding of category and category-independent features emerges along the visual hierarchy, we next examine decoding performance as a function of layer depth (**Fig. 7a**). Experimental studies probing the joint-code hypothesis in the brain have focused on improvements in global regression performance across the visual hierarchy [20], although this performance, as estimated from current recordings, remains limited even in IT (see SI). Our empirical results on CNNs show, however, that improvement in regression performance along the hierarchy is not by itself diagnostic of a joint population code, since it occurs even in a network that performs well only on classification (network C). In network C, moreover, this improvement is confined to the local error, while the local-global gap remains high across layers, resulting in poor global regression performance. By contrast, changes in the local-global gap across layers appear more discriminative of a joint code: in network CR, the gap is reduced by orders of magnitude in the final layers.

Another interesting observation is that, in network CR, regression performance improves sharply only in the final layers, concurrently with a sharp rise in classification accuracy. One possible explanation is that estimating an object's bounding box depends on first recognizing the object, tightly linking regression performance to classification accuracy. This may also help explain the across-layer improvements in regression performance observed in network C despite the absence of explicit regression optimization, although other explanations are possible [36], and a thorough investigation of them lies beyond the scope of the present work.

We also report how the manifold-geometry measures contributing to the local–global gap evolve across layers—namely the centroid, scale, and orientation errors (**Fig. 7b**). In both networks C and CR, the centroid error is only relevant in early layers, while in later layers it becomes negligible compared to the other sources of error. The scale error remains approximately constant in network C, whereas network CR tends to reduce it in the final layers. In contrast, the orientation error increases sharply in network C. This behavior can be understood by inspecting the two components of the orientation error: SNR and alignment a (**Fig. 7c**). In network C, alignment a remains approximately constant, while SNR decreases drastically. This is consistent with the classification-only objective of network C, which incentivizes suppression of class-irrelevant variability such as bounding-box coordinates, thereby reducing their SNR. By contrast, network CR appears to counteract this increase in orientation er-

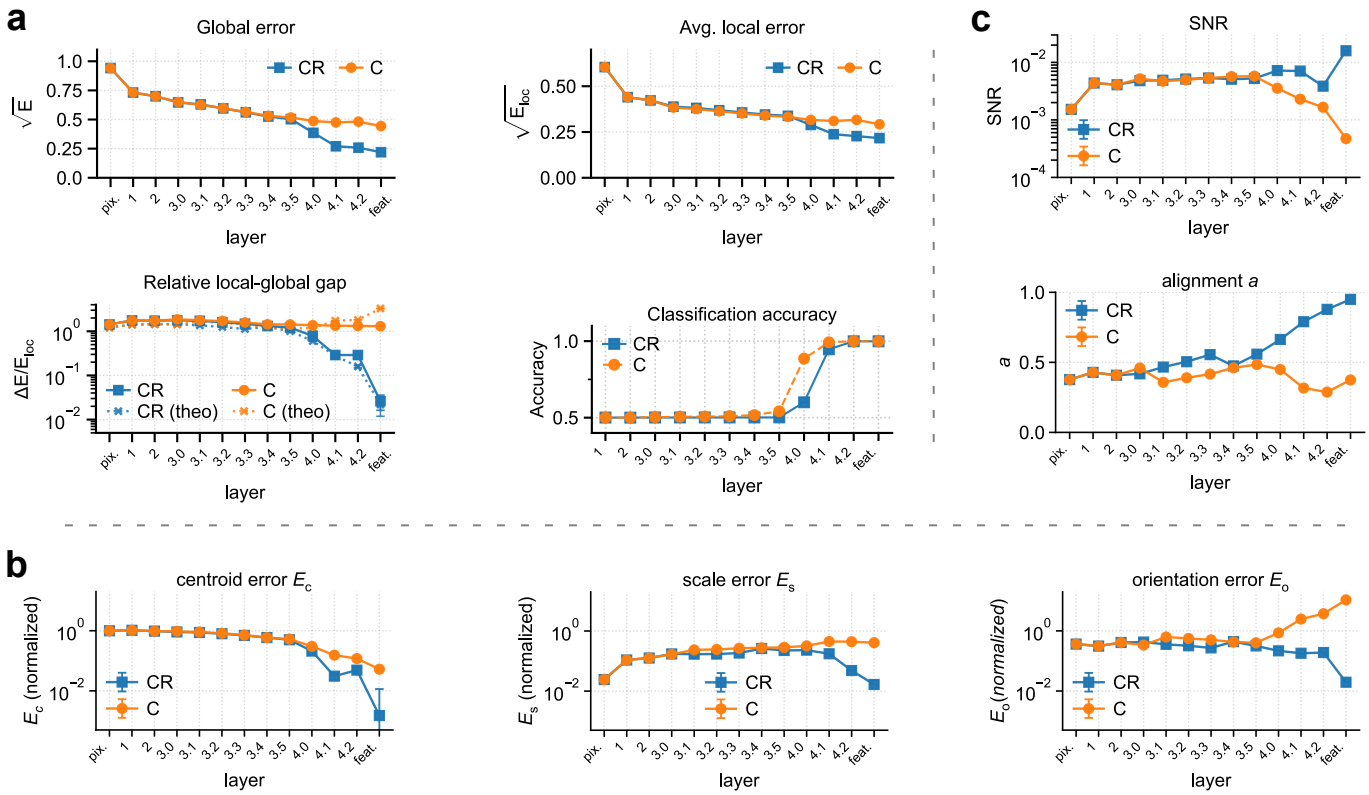

Figure 7. Decoding performance along the visual hierarchy. (**a**) Regression performance for the bounding-box parameter L_{h} and classification performance, using linear readouts from progressively deeper layers of networks C (orange circles, full line) and CR (blue squares, full line). At each layer, linear readouts were trained on activations randomly projected to 2048 dimensions, to match the dimensionality of the final feature layer. For regression, we report the global regression error, as well as its decomposition into local error E_{loc} and relative local-global gap $\Delta E/E_{\mathrm{loc}}$. For the gap, we also report our theory prediction (x markers, dotted lines). Note that global and local error are reported as root nMSE. (**b**) The three contributions to the relative local-global gap, as identified by our theory: centroid error E_{c}, orientation error E_{o}, scale error E_{s}. These quantities are normalized in the same way as explained in **Fig. 5**. (**c**) The two elements contributing to the orientation error: SNR, and alignment a of the local feature-encoding directions.

ror by improving both SNR and alignment a across layers.

6. *Subsampling effects*

We next characterize how typical experimental constraints—namely the subsampling of neural units, and a limited number of categories in the image dataset—affect the estimation of regression performance. We have shown that the most distinctive feature of network CR compared to C is a small relative local–global gap ($\Delta E/E_{\mathrm{loc}} \ll 1$), we therefore focus on this quantity here, leaving an analysis of the local error E_{loc} to the SI.

We first examine the effect of subsampling the number P of categories in our image dataset (**Fig. 8a**). In network C, the relative gap $\Delta E/E_{\mathrm{loc}}$ is strongly underestimated when regressing from few categories (blue curve), potentially making a classification-only population code appear consistent with a joint code. We extended our theory (see Online Methods and SI) to capture this finite-P behavior (green curve), which can be intuitively understood as a form of overfitting. Crucially, our theory allows one to avoid mistaken conclusions due to this finite-P overfitting, as it allows to reliably extrapolate the local-global gap to the $P \to \infty$ limit (red curve). In particular, even when the theory is fit using manifold statistics computed from as few as $P = 8$ categories (the same number of categories used, for example, to obtain ventral stream recordings [34] probing the joint-code hypothesis [20]) it predicts a local–global gap consistent with the value predicted using all of the $P = 265$ categories (i.e., the red curve is relatively flat).

Second, we examine the effect of neural-unit subsampling (**Fig. 8b**). For sufficiently small number of neural units, subsampling becomes particularly destructive: in network CR, $\Delta E/E_{\mathrm{loc}}$ increases to the same level as in network C, rendering the two networks indistinguishable by this key metric. All three theoretical error sources con-

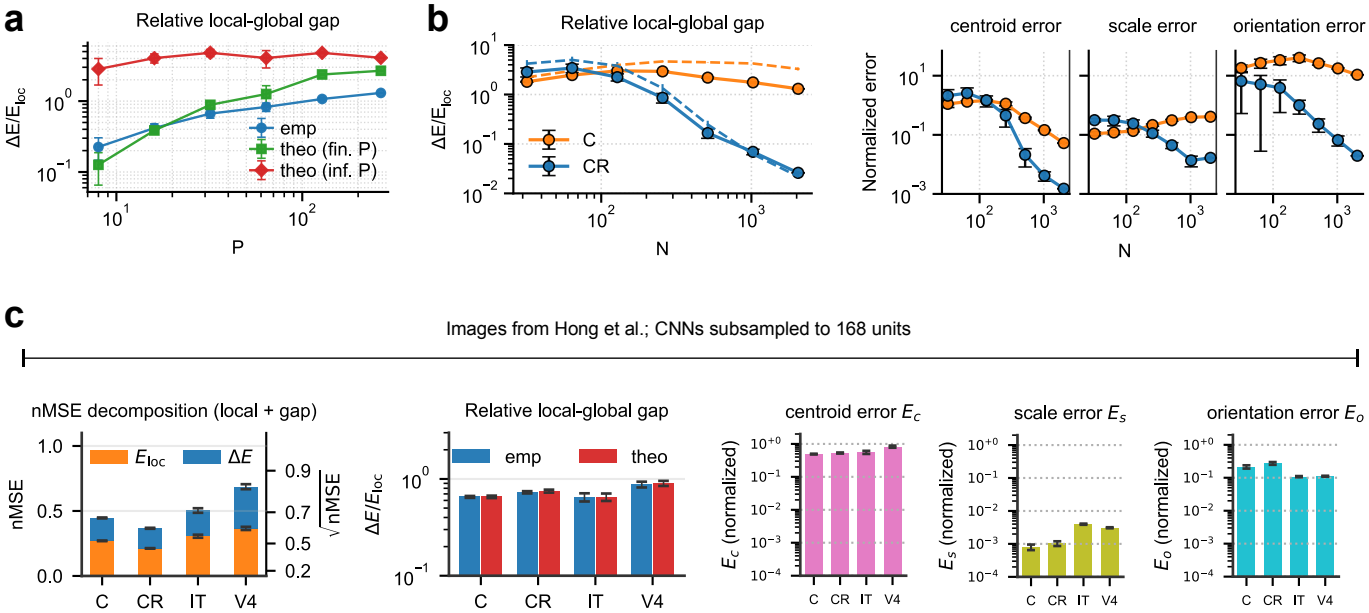

Figure 8. Subsampling effects. (**a**) relative gap $\Delta E/E_{\text{loc}}$ for regressing the bounding-box feature L_h from network C, as a function of the number P of randomly subsampled categories (out of the 265 in our dataset). Blue dots: measured gap; red diamonds: theoretical prediction in the $P \to \infty$ limit; Green squares: theoretical prediction for finite P. Error bars denote standard deviation across 4 independent category subsamples. (**b**) $\Delta E/E_{\text{loc}}$ (left) and associated error sources (right) for regressing the bounding-box feature L_h from networks C (orange) and CR (blue) on our dataset, as a function of the number N of randomly subsampled units (out of 2048 feature-layer units). (**c**) Analysis of the regression performance of feature L_h, using the images and corresponding neural recordings found in [34]. We report the analysis of regression from the IT and V4 units recorded in [34], as well as 168 randomly projected units from the feature layer of networks C and CR. Error bars denote SEM across cross-validation splits and, where applicable, random unit subsamples.

tributing to ΔE converge to similar values in networks C and CR, with the orientation error E_{o} and centroid error E_{c} becoming the dominant error contributions. We find a similar phenomenology when analyzing the images and corresponding neural recordings in [34] (**Fig. 8c**): IT, V4, and networks C and CR subsampled to 168 units—the number recorded in IT—all show comparable values of $\Delta E/E_{\text{loc}}$, as well as comparable contributions from each of the three error sources, with E_{o} and E_{c} emerging as the dominant terms. The similar phenomenology observed in the neural recordings (**Fig. 8c**) and in our controlled CNN analysis (**Fig. 8b**) suggests that this collapse of the local-global gap may, also in the neural data, be largely driven by the limited number of recorded units, and that future experiments may benefit from larger-scale recordings.

III. DISCUSSION

In this work, we addressed the question of what properties of a population code enable an effective joint encoding of object category and category-independent features. Using ventral-stream–like CNNs [22], we first showed that such a joint code is possible: a CNN can be trained so that the same feature-layer representation supports accurate linear readout of both category and bounding-box parameters. We established this on a large-scale single-object image dataset developed for this study, featuring photo-realistic images with controlled bounding-box distributions. We then developed a theory within the framework of category manifolds [24, 25] to understand which manifold properties enable not only linear separability of categories, but also regression of category-independent features.

While previous manifold-based theories linked category-manifold geometry to classification performance [24, 25, 37], here we extended that framework to provide an analogous account for regression of category-independent features. Central to our theoretical framework is a decomposition of global regression error into a local error, which quantifies how well a feature is linearly encoded within individual category manifolds, and a local-global gap, which captures whether this encoding is organized consistently enough across manifolds to support accurate readout by a single category-independent regressor. This decomposition proves essential for characterizing a joint code. In our empirical results, joint-code representations exhibit a strongly suppressed local-global gap, reduced by orders of magnitude relative to a population code that performs well only on object classification. By contrast, differences

in local regression error between the two codes are more modest. Notably, a small local-global gap can emerge even when a network is trained only for classification and *category-specific* regression, without the additional constraint of a shared *category-independent* regressor.

Experimental studies probing the joint-code hypothesis in the brain have focused on improvements in global regression performance across the visual hierarchy [20], although this performance, as estimated from current recordings, remains limited even in IT. Here we show that a similar improvement across layers can be seen even in networks optimized only for classification. However, this improvement is limited to the local error, while the local-global gap remains high, resulting in poor overall performance. Whether the limited regression performance currently estimated in IT reflects experimental constraints, or instead the absence of joint-code properties such as a suppressed local-global gap, is hard to judge from the present experimental data. To elucidate this, we have characterized how experimental constraints affect the estimation of regression performance. In particular, we showed that subsampling neural units causes the estimated local-global gap to increase sharply and, below ~ 200 units, to reach similar values in networks with and without a suppressed local-global gap, making the two difficult to distinguish. This suggests that future experiments may require larger-scale recordings to estimate the local-global gap reliably.

We also considered the issue of finite number of categories, denoted here by P, in experimental setups. Previous manifold theories of classification [24, 37] focused on the regime of a finite number of categories per neuron, namely a fixed ratio of $\alpha = P/N$ where N is the number of neurons used for the readout, or on tasks involving a small number of categories, such as few-shot generalization of object categories [25]. In contrast, for a representation to support global regression of category-independent features, it must be able to do so for images associated with any category from the same statistical ensemble, hence the relevant limit is $P \to \infty$ for large but fixed N. Indeed our theory predicts that the local-global gap can be strongly underestimated in the finite-P regime. Remarkably, the theory shows how to extrapolate results from finite P to the large P limit (cf. **Fig.** 8**a**). In this sense, our work goes beyond previous empirical analyses of the local-global gap in pairs of manifolds [38], by providing the first theoretical account of this gap in terms of interpretable manifold-geometric measures, and by formulating it in the $P \to \infty$ regime most relevant for cross-category generalization, while at the same time yielding accurate finite-P corrections.

By idetifying the manifold-geometry measures predictive of the local-global error gap, our theory allows to explain how these measures can be optimized alongside the manifold geometry relevant to classification performance, thereby implementing an effective joint code. This question can be addressed here because our theory of regression is formulated in the language of category manifolds, and therefore in the same framework used to study classification [24, 25, 37], unlike a recent theory of manifold regression [39], in which manifolds are defined by discretized values of the regressed feature rather than by category. Specifically, we find that regression-relevant geometry is mostly related to a specific direction within the manifold—the feature-encoding direction—rather than the manifold as a whole, and can therefore be optimized with marginal effects on classification-relevant geometry such as manifold radius, dimensionality, and centroid separation.

More generally, this implies that regression optimization can leave only subtle signatures on coarse manifold-geometry measures. This helps explain why CNNs trained only for classification and CNNs trained jointly for classification and regression exhibit highly similar representations under widely used similarity measures such as CKA, as well as similar alignment scores with neural data, and may likewise account for analogous results reported previously [23]. Our results therefore highlight the importance of finer geometric measures—such as those identified by our theory—that can distinguish a true joint code even when coarser representational-similarity measures cannot.

An open question is whether this picture persists when representations must encode many more category-independent variables. At present, large-scale datasets for studying the joint-code hypothesis [23], including ours, contain annotations for only a few such variables, making the development of richer datasets an important challenge for future work. Constructing such datasets may draw on prior work that parameterized additional visual features, including texture [14], material properties [13], shape [16], and facial properties [19, 40]. Alternatively, relevant features could be defined as stimulus dimensions that maximize neural responses [41–43].

Several directions remain open for characterizing joint codes in artificial networks. Here we focused on CNNs as ventral-stream models [21–23], and on supervised training to compare representations explicitly optimized for a joint code with those optimized only for classification. An important next step is to ask whether joint codes also emerge under self-supervised training [44–50], and how different training augmentations may preserve or degrade linear access to category-independent variables [51–53]. We also did not consider state-of-the-art object detection networks [54–59], because our goal was to study direct linear readout from a single shared representation rather than the more elaborate decoding pipelines used in these systems. Among current detectors, DETR [59] is arguably closest to implementing a joint population code, since both category and bounding-box information are read out from a common representation space, albeit nonlinearly for bounding-box features, making it a natural target for future study.

ACKNOWLEDGMENTS

We thank Xu Pan, and David Klindt for helpful discussions. This research is supported by the Swartz Foundation, the Gatsby Charitable Foundation, the Kempner Institute for the Study of Natural and Artificial Intelligence at Harvard University, the grant NSF PHY-2309135 to the Kavli Institute for Theoretical Physics and Office of Naval Research grant No. N0014-23-1-2051.

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

[2] S. Bernardi, M. K. Benna, M. Rigotti, J. Munuera, S. Fusi, and C. D. Salzman, The geometry of abstraction in the hippocampus and prefrontal cortex, Cell **183**, 954 (2020).

[3] S. Fusi, E. K. Miller, and M. Rigotti, Why neurons mix: high dimensionality for higher cognition, Current opinion in neurobiology **37**, 66 (2016).

[4] A. Dubreuil, A. Valente, M. Beiran, F. Mastrogiuseppe, and S. Ostojic, The role of population structure in computations through neural dynamics, Nature neuroscience **25**, 783 (2022).

[5] A. Saxe, S. Sodhani, and S. J. Lewallen, The neural race reduction: Dynamics of abstraction in gated networks, in *International Conference on Machine Learning* (PMLR, 2022) pp. 19287–19309.

[6] T. Flesch, K. Juechems, T. Dumbalska, A. Saxe, and C. Summerfield, Orthogonal representations for robust context-dependent task performance in brains and neural networks, Neuron **110**, 1258 (2022).

[7] W. J. Johnston and S. Fusi, Abstract representations emerge naturally in neural networks trained to perform multiple tasks, Nature Communications **14**, 1040 (2023).

[8] M. Ito, H. Tamura, I. Fujita, and K. Tanaka, Size and position invariance of neuronal responses in monkey inferotemporal cortex, Journal of neurophysiology **73**, 218 (1995).

[9] C. P. Hung, G. Kreiman, T. Poggio, and J. J. DiCarlo, Fast readout of object identity from macaque inferior temporal cortex, Science **310**, 863 (2005).

[10] N. Kriegeskorte, M. Mur, D. A. Ruff, R. Kiani, J. Bodurka, H. Esteky, K. Tanaka, and P. A. Bandettini, Matching categorical object representations in inferior temporal cortex of man and monkey, Neuron **60**, 1126 (2008).

[11] T. Konkle and A. Oliva, A real-world size organization of object responses in occipitotemporal cortex, Neuron **74**, 1114 (2012).

[12] T. Konkle and A. Caramazza, Tripartite organization of the ventral stream by animacy and object size, Journal of Neuroscience **33**, 10235 (2013).

[13] N. Goda, A. Tachibana, G. Okazawa, and H. Komatsu, Representation of the material properties of objects in the visual cortex of nonhuman primates, Journal of Neuroscience **34**, 2660 (2014).

[14] G. Okazawa, S. Tajima, and H. Komatsu, Image statistics underlying natural texture selectivity of neurons in macaque v4, Proceedings of the National Academy of Sciences **112**, E351 (2015).

[15] T. Kim, W. Bair, and A. Pasupathy, Neural coding for shape and texture in macaque area v4, Journal of Neuroscience **39**, 4760 (2019).

[16] S. L. Brincat and C. E. Connor, Underlying principles of visual shape selectivity in posterior inferotemporal cortex, Nature neuroscience **7**, 880 (2004).

[17] Y. Yamane, E. T. Carlson, K. C. Bowman, Z. Wang, and C. E. Connor, A neural code for three-dimensional object shape in macaque inferotemporal cortex, Nature neuroscience **11**, 1352 (2008).

[18] W. A. Freiwald, D. Y. Tsao, and M. S. Livingstone, A face feature space in the macaque temporal lobe, Nature neuroscience **12**, 1187 (2009).

[19] L. Chang and D. Y. Tsao, The code for facial identity in the primate brain, Cell **169**, 1013 (2017).

[20] H. Hong, D. L. Yamins, N. J. Majaj, and J. J. DiCarlo, Explicit information for category-orthogonal object properties increases along the ventral stream, Nature neuroscience **19**, 613 (2016).

[21] D. L. Yamins, H. Hong, C. F. Cadieu, E. A. Solomon, D. Seibert, and J. J. DiCarlo, Performance-optimized hierarchical models predict neural responses in higher visual cortex, Proceedings of the national academy of sciences **111**, 8619 (2014).

[22] M. Schrimpf, J. Kubilius, H. Hong, N. J. Majaj, R. Rajalingham, E. B. Issa, K. Kar, P. Bashivan, J. Prescott-Roy, F. Geiger, *et al.*, Brain-score: Which artificial neural network for object recognition is most brain-like?, BioRxiv , 407007 (2018).

[23] Y. Xie, W. Huang, E. Alter, J. Schwartz, J. B. Tenenbaum, and J. J. DiCarlo, Vision cnns trained to estimate spatial latents learned similar ventral-stream-aligned representations, in *The Thirteenth International Conference on Learning Representations*.

[24] S. Chung, D. D. Lee, and H. Sompolinsky, Classification and geometry of general perceptual manifolds, Physical Review X **8**, 031003 (2018).

[25] B. Sorscher, S. Ganguli, and H. Sompolinsky, Neural representational geometry underlies few-shot concept learning, Proceedings of the National Academy of Sciences **119**, e2200800119 (2022).

[26] U. Cohen, S. Chung, D. D. Lee, and H. Sompolinsky, Separability and geometry of object manifolds in deep neural networks, Nature communications **11**, 746 (2020).

[27] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, Sdxl: Improving latent diffusion models for high-resolution image synthesis, arXiv preprint arXiv:2307.01952 (2023).

[28] S. Lin, A. Wang, and X. Yang, Sdxl-lightning: Progressive adversarial diffusion distillation, arXiv preprint arXiv:2402.13929 (2024).

[29] S. Shao, Z. Li, T. Zhang, C. Peng, G. Yu, X. Zhang, J. Li, and J. Sun, Objects365: A large-scale, high-quality dataset for object detection, in *Proceedings of the IEEE/CVF international conference on computer vision* (2019) pp. 8430–8439.

[30] I. Tolstykh, M. Chernyshov, and M. Kuprashevich, Cerberusdet: Unified multi-dataset object detection, in *International Conference on Image Processing and Vision Engineering* (Springer, 2025) pp. 95–110.

[31] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, High-resolution image synthesis with latent diffusion models, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2022) pp. 10684–10695.

[32] S. Luo, Y. Tan, L. Huang, J. Li, and H. Zhao, Latent consistency models: Synthesizing high-resolution images with few-step inference, arXiv preprint arXiv:2310.04378 (2023).

[33] S. Luo, Y. Tan, S. Patil, D. Gu, P. Von Platen, A. Passos, L. Huang, J. Li, and H. Zhao, Lcm-lora: A universal stable-diffusion acceleration module, arXiv preprint arXiv:2311.05556 (2023).

[34] N. J. Majaj, H. Hong, E. A. Solomon, and J. J. DiCarlo, Simple learned weighted sums of inferior temporal neuronal firing rates accurately predict human core object recognition performance, Journal of Neuroscience **35**, 13402 (2015).

[35] S. Kornblith, M. Norouzi, H. Lee, and G. Hinton, Similarity of neural network representations revisited, in *International conference on machine learning* (PMlR, 2019) pp. 3519–3529.

[36] P. Reizinger, A. Bizeul, A. Juhos, J. E. Vogt, R. Balestriero, W. Brendel, and D. Klindt, Cross-entropy is all you need to invert the data generating process, arXiv preprint arXiv:2410.21869 (2024).

[37] A. J. Wakhloo, T. J. Sussman, and S. Chung, Linear classification of neural manifolds with correlated variability, Physical Review Letters **131**, 027301 (2023).

[38] M. Kuoch, C.-N. Chou, N. Parthasarathy, J. Dapello, J. J. DiCarlo, H. Sompolinsky, and S. Chung, Probing biological and artificial neural networks with task-dependent neural manifolds, in *Conference on Parsimony and Learning* (PMLR, 2024) pp. 395–418.

[39] W. Slatton, C.-N. Chou, and S. Chung, Linear readout of neural manifolds with continuous variables, arXiv preprint arXiv:2603.10956 (2026).

[40] I. Yildirim, M. Belledonne, W. Freiwald, and J. Tenenbaum, Efficient inverse graphics in biological face processing, Science advances **6**, eaax5979 (2020).

[41] C. R. Ponce, W. Xiao, P. F. Schade, T. S. Hartmann, G. Kreiman, and M. S. Livingstone, Evolving images for visual neurons using a deep generative network reveals coding principles and neuronal preferences, Cell **177**, 999 (2019).

[42] W. Xiao and G. Kreiman, Xdream: Finding preferred stimuli for visual neurons using generative networks and gradient-free optimization, PLoS computational biology **16**, e1007973 (2020).

[43] B. Wang and C. R. Ponce, Tuning landscapes of the ventral stream, Cell reports **41** (2022).

[44] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, A simple framework for contrastive learning of visual representations, in *International conference on machine learning* (PmLR, 2020) pp. 1597–1607.

[45] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, Momentum contrast for unsupervised visual representation learning, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020) pp. 9729–9738.

[46] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. Richemond, E. Buchatskaya, C. Doersch, B. Avila Pires, Z. Guo, M. Gheshlaghi Azar, *et al.*, Bootstrap your own latent-a new approach to self-supervised learning, Advances in neural information processing systems **33**, 21271 (2020).

[47] A. Bardes, J. Ponce, and Y. LeCun, Vicreg: Variance-invariance-covariance regularization for self-supervised learning, arXiv preprint arXiv:2105.04906 (2021).

[48] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, Emerging properties in self-supervised vision transformers, in *Proceedings of the IEEE/CVF international conference on computer vision* (2021) pp. 9650–9660.

[49] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, Masked autoencoders are scalable vision learners, in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2022) pp. 16000–16009.

[50] C. Zhuang, S. Yan, A. Nayebi, M. Schrimpf, M. C. Frank, J. J. DiCarlo, and D. L. Yamins, Unsupervised neural network models of the ventral visual stream, Proceedings of the National Academy of Sciences **118**, e2014196118 (2021).

[51] F. Locatello, S. Bauer, M. Lucic, G. Raetsch, S. Gelly, B. Schölkopf, and O. Bachem, Challenging common assumptions in the unsupervised learning of disentangled representations, in *international conference on machine learning* (PMLR, 2019) pp. 4114–4124.

[52] T. Konkle and G. A. Alvarez, A self-supervised domain-general learning framework for human ventral stream representation, Nature communications **13**, 491 (2022).

[53] A. Nayebi, N. C. Kong, C. Zhuang, J. L. Gardner, A. M. Norcia, and D. L. Yamins, Mouse visual cortex as a limited resource system that self-learns an ecologically-general representation, PLOS Computational Biology **19**, e1011506 (2023).

[54] S. Ren, K. He, R. Girshick, and J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in neural information processing systems **28** (2015).

[55] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, Ssd: Single shot multibox detector, in *European conference on computer vision* (Springer, 2016) pp. 21–37.

[56] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, Focal loss for dense object detection, in *Proceedings of the IEEE international conference on computer vision* (2017) pp. 2980–2988.

[57] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, You only look once: Unified, real-time object detection, in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016) pp. 779–788.

[58] Z. Tian, C. Shen, H. Chen, and T. He, Fcos: Fully convolutional one-stage object detection, in *Proceedings of the IEEE/CVF international conference on computer vision* (2019) pp. 9627–9636.

[59] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, End-to-end object detection with transformers, in *European conference on computer vision* (Springer, 2020) pp. 213–229.

[60] L. Song, A. Smola, A. Gretton, J. Bedo, and K. Borgwardt, Feature selection via dependence maximization., Journal of Machine Learning Research **13** (2012).

[61] J. Freeman, C. M. Ziemba, D. J. Heeger, E. P. Simoncelli, and J. A. Movshon, A functional and perceptual signature of the second visual area in primates, Nature neuroscience **16**, 974 (2013).

Online Methods

A. Code availability

The code used to reproduce the main results of this study, including the generation of the image dataset, the training of the CNNs, the linear decoding analyses, and the evaluation of the theory, is publicly available at https://github.com/tiberilor/manifolds-joint-code.

B. Linear decoding

To measure the performance of linear readouts in classifying object category and regressing category-independent features, we used the following methods.

Classification. We decoded object category using linear one-vs-rest classifiers trained with ridge regression. For each category, we fit a separate linear readout with binary label ($y = 1$ for images in the category, $y = 0$ otherwise) and converted readout scores to class predictions using a fixed threshold of 0.5. The ridge regularization strength was chosen by nested cross-validation, and reported performance was computed on held-out data across outer splits. Classification performance is reported as balanced accuracy, defined for each one-vs-rest task as

$$\mathrm{BalAcc} = \frac{1}{2}\left(\frac{\mathrm{TP}}{\mathrm{Pos}} + \frac{\mathrm{TN}}{\mathrm{Neg}}\right) \tag{5}$$

where TP and TN are the numbers of true positives and true negatives on the test set, and Pos and Neg are the total numbers of positive and negative test samples, respectively. We report the final performance as balanced accuracy averaged across categories.

Global regression. To decode bounding-box features at the global, across-category level, we fit linear ridge regressors from the neural representation to each scalar bounding-box feature separately. The ridge regularization strength was selected by nested cross-validation, and reported performance was computed on held-out data across outer splits. Outer and inner splits used a k-fold procedure. Regression performance is reported as nMSE, i.e., MSE averaged across outer splits, and then normalized by dividing by the variance of the regressed feature.

Local regression. To estimate the local linear encoding of bounding-box features within each category (used to compute E_{loc} and the geometry statistics entering our theory), we fit category-specific linear regressors ($W^\mu \in \mathbb{R}^N, b^\mu \in \mathbb{R}$) by joint optimization across all categories. Specifically, we minimize the average within-category MSE plus a shared regularizer that penalizes dispersion of the regressors around their mean:

$$E_{\mathrm{common}} = \frac{1}{P}\sum_{\mu=1}^{P}\left\langle \left(W^{\mu\top}x^\mu + b^\mu - y^\mu\right)^2\right\rangle_\mu + \gamma L\left(\{W^\mu\}\right) \tag{6}$$

with

$$L\left(\{W^\mu\}\right) = \frac{1}{P}\sum_{\mu=1}^{P}\left\| W^\mu - \frac{1}{P}\sum_{\nu=1}^{P} W^\nu \right\|^2 \tag{7}$$

Here $x^\mu \in \mathbb{R}^N$ is a representation in category μ and $y^\mu \in \mathbb{R}$ is the associated value of the bounding-box feature being regressed, P is the number of categories, and $\langle\rangle_\mu$ denotes averaging over samples of category μ in the training split, and γ controls the strength of the shared regularizer. The regularization strength γ was selected by nested cross-validation. Performance was evaluated on held-out data across outer splits by computing E_{loc} as the average test nMSE across categories. In practice, this procedure yielded E_{loc} values statistically compatible with those obtained by fitting separate ridge regressors per category. Compared to fitting independent per-category ridge regressors, the joint objective above has the advantage of yielding more stable estimates of the local regressors $\{W^\mu\}$, which enter our theory through their across-category variability in norm and direction (i.e., the scale error E_{s} and orientation error E_{o}). Indeed, when fitting each category independently with its own ridge penalty selected by cross-validation, we found that a broad range of ridge penalties yielded statistically indistinguishable performance. This leads to high variability in the selected penalties across categories and, consequently, unnecessary variability in the estimated $\{W^\mu\}$, which can inflate the estimates of norm and direction variability. The coupled objective above reduces this estimation variance by shrinking each W^μ toward the mean regressor $\bar{W} = \frac{1}{P}\sum_{\nu=1}^{P} W^\nu$.

Data splits. When applied to artificial networks, all decoding analyses explained above were performed on images not used to train the networks. Specifically, our image dataset consists of 10,000 images per category, which we used for training networks C, R, and CR, and a separate set of another 10,000 images per category, unseen during training, which we used to evaluate linear decoding performance.

C. Dataset generation

Our dataset consists of photorealistic images representing a single main object, placed in a contextually relevant scene. Each image is labeled with both the object identity (out of 265 categories) and the parameters of the bounding-box surrounding the main object, namely center coordinates C_h, C_v and side lengths L_h, L_v along the image's horizontal (h) and vertical (v) axes. The 265

categories are a subset of the 365 categories in the taxonomy of the Object365 image dataset, chosen because they could be generated reliably with our pipeline (described below). Each category consists of 10,000 training images (used to train the CNNs) and a disjoint set of 10,000 test images, unseen during training, used to evaluate linear decoding performance.

Another recently introduced large-scale dataset for testing the joint-code hypothesis consists of images of 3D object models rendered in randomized 3D environments and annotated with object category together with six category-independent features: three object-position coordinates and three pose parameters [23]. Relative to that dataset, ours provides four category-independent variables, namely the bounding-box coordinates. In exchange for giving up explicit 3D control, however, our dataset gains images with photorealistic appearance and contextually relevant backgrounds, as they are generated by diffusion models trained on natural-image statistics. The two datasets therefore occupy complementary points along the trade-off between controllability and realism.

Images in our dataset were generated using a multi-stage pipeline that combines text-to-image generation (Stable Diffusion XL, SDXL [27, 28]), object detection (CerberusDet [30]), and image outpainting (Stable Diffusion v1.5, SD1.5 [31–33]) to produce single-object images with controlled object placement and controlled distribution of bounding-box coordinates. The pipeline to generate one image in our dataset consists of the following three steps (cf. **Fig. 2a**).

Step 1: Object generation. We first generate a "seed" image of a single object in the desired category using Stable Diffusion XL. The goal of this step is simply to obtain a clear, unobstructed, single-instance depiction of the target object in a contextually appropriate scene, without yet enforcing any control over object position or scale (i.e., control over the object's bounding-box parameters). To make large-scale generation feasible, we use SDXL-Lightning weights (model release identifier: ByteDance/SDXL-Lightning) with the SDXL base checkpoint (stabilityai/stable-diffusion-xl-base-1.0), enabling generation in only 4 denoising steps. This allowed us to synthesize the full dataset within a few days on a 10×A100 GPU cluster. For a given category named <category>, we use the following prompt: "A full-body, photorealistic shot of a single <category>. The <category> is the clear main subject, fully in the scene, and unobstructed. The background is detailed, contextually appropriate, but does not distract.". We also use the negative prompt: "clutter, multiple objects, occlusion, low quality, blurry, distorted, unrealistic, extra limbs, cropped, artifacts, persons, people, humans, mannequin".

Step 2: Object placement. The goal of this step is to enforce controlled object location and size by mapping the seed image from step 1 onto a new canvas so that the main object occupies a newly sampled bounding box. We first obtain the object's bounding box in the seed image using CerberusDet, and treat this detected box as the object's initial bounding box. If CerberusDet does not detect any instance of the object, we discard the seed image and return to step 1. We then sample a target bounding box on a fixed 512×512 canvas as follows. First, we sample a target maximum side length $L_{\max} \sim \mathrm{Unif}\,[200, 500]$ (in pixel units) and isotropically rescale the seed image such that the detected bounding-box's maximum side length matches the newly sampled one, that is we ensure $\max(L_h, L_v) = L_{\max}$. Next, conditioned on the resulting (L_h, L_v), we sample the target bounding-box center uniformly over all centers that would keep the box inside the canvas: $C_h \sim \mathrm{Unif}\,[L_h/2, 512 - L_h/2]$ and $C_v \sim \mathrm{Unif}\,[L_v/2, 512 - L_v/2]$. In sum, object size (via $L_{\max}$) is sampled uniformly over a prescribed range, and object position is sampled uniformly conditional on size, subject to the newly sampled bounding-box staying fully within the image. To realize this target bounding-box, we paste the rescaled seed image onto a blank 512×512 canvas so that the object bounding-box matches the target one. Some regions of the canvas will remain uncovered by the rescaled images, and are filled in Step 3 via outpainting.

Step 3: Outpainting. The goal of this step is to fill in the uncovered regions introduced in Step 2, producing a natural-looking full 512×512 image while preserving the placed object and its newly sampled bounding box. We achieve this by outpainting the uncovered regions using an outpainting variant of SD1.5. Starting from the 512×512 canvas produced in Step 2, we construct an outpainting mask that marks the uncovered regions. To reduce visible seams at the boundary between original and synthesized content, we blur the mask edges with a Gaussian blur. Before outpainting, we initialize the undefined regions with reflection padding to provide the generative model with plausible color and texture statistics and reduce sharp discontinuities at the boundary between the pasted seed image and the area to be synthesized. Technically, we fill each blank side by repeatedly tiling a mirror-reflected strip of pixels taken only from the portion of the seed image between the pasted-image border and the object's bounding-box edge (never from inside the bounding box), so that the object itself is not reflected—avoiding to encourage the outpainting model to generate additional instances of the object. We finally outpaint the masked regions using SD1.5 inpainting (stable-diffusion-v1-5/stable-diffusion-inpainting) with an LCM scheduler and the latent-consistency/lcm-lora-sdv1-5 LoRA, using 4 denoising steps, guidance scale 0, and strength 0.95. The outpainting prompt is the same category prompt from Step 1 with the suffix: 'The background extends naturally from the existing scene. Seamless transition between regions, no visible boundaries. Consistent lighting and focus throughout the entire image." The negative prompt is identical to Step 1. Finally, we rerun CerberusDet on the outpainted 512×512 image and accept the sample only if the detector returns exactly one instance

of the desired category, so that images with multiple instances of the object are not accepted.

Categories selection. As mentioned above, we generated images in 265 categories out of the 365 categories in the Objects365 taxonomy, selected among those that could be generated reliably with our pipeline. To select these categories, we ran a diagnostic pass generating 50 images per category and measured how often the pipeline failed due to CerberusDet either not detecting the target object or detecting multiple instances. We excluded categories with high failure rates, since their low yield would have substantially increased the runtime and compute required to produce 20,000 accepted images per category. Furthermore, we visually inspected the 50 images in each category and excluded categories that systematically produced implausible or low-quality images, such as ambiguous or inconsistent object identity, severe artifacts, or the presence of multiple salient objects. For a few categories, SDXL introduced characteristic artifacts that were *consistent* across images within that category (e.g., "knife" images often contained forked or unusually curved blades). Since our goal is a dataset in which images within each category share a consistent semantic label, rather than strict realism, we did not exclude such categories. During this inspection, we also identified categories for which CerberusDet consistently localized the intended object but systematically assigned a different class label. Because we used CerberusDet only to extract bounding boxes, we defined for each category a small set of accepted detector labels and accepted an image if the detector returned exactly one instance with a label in that set.

D. CNN architecture and training

Architecture. All CNN models were based on an ImageNet-pretrained ResNet-50. We removed the original classification head and used the final 2048-dimensional feature layer as the shared representation. On top of this feature layer we attached either one or a combination of the following heads: a linear classification head for object category (265 classes) and four linear regression heads predicting the bounding-box parameters (C_h, C_v, L_h, L_v). These heads were randomly initialized with Xavier-uniform weights and zero biases. We trained four variants of this architecture. Network C included only the classification head. Network R included only the four regression heads. Network CR included both the classification head and the four regression heads. Network CRloc had the same backbone and classification head as CR, but replaced each category-independent regression head with a category-specific one: each bounding-box head produced one output per class, and during training the output corresponding to the ground-truth class was selected.

Training. Images were resized to 224×224 and normalized with the standard ImageNet channel statistics. Bounding-box coordinates were standardized using the training-set mean and standard deviation before regression. Training used shuffled mini-batches from the training split (8000 images per category) and performance was monitored on a held-out validation split (2000 images per category). Models were trained with batch size 208 using stochastic gradient descent (learning rate: 0.02; momentum: 0.9; weight decay: 10^{-4}; learning-rate schedule: cosine annealing over 40 epochs), Smooth L1 loss for each regression coordinate, and cross-entropy loss for classification. The total loss was the sum of the regression and classification losses. In models without the classification head or without regression heads, the corresponding loss terms were simply omitted. Models were trained until performance on the validation set plateaued.

E. Theory definitions and results

Regression errors definition. Our theory is derived for the training regression error (when applied to real data, however, we test its predictions against the corresponding test error; see 'Fitting theory to data'). Here we define the training regression errors entering the theory. Consider a set C consisting of P categories. Call $x^\mu \in \mathbb{R}^N$ the neural representation of an image (or, more generally, an input stimulus) in the μ-th category, where N is the number of neural units. Let y^μ be a continuous label associated to the image (e.g. one of the bounding-box parameters). Without loss of generality, we consider globally centered data and labels, that is $\left\langle \langle x^\mu \rangle_\mu \right\rangle_C = 0$ and $\left\langle \langle y^\mu \rangle_\mu \right\rangle_C = 0$, where $\langle \ldots \rangle_\mu$ denotes averaging over all images within the μ-th category, and $\langle \ldots \rangle_C = \mathrm{Avg}_\mu [\ldots]$ denotes averaging over categories.

Define the global regression MSE as

$$E = \min_W \left[\left\langle \left\langle \left(W^\top x^\mu - y^\mu \right)^2 \right\rangle_\mu \right\rangle_C \right] \tag{8}$$

where $W \in \mathbb{R}^N$ is the global regression vector. We decompose the global regression MSE as $E = E_{\mathrm{loc}} + \Delta E$. Here, $E_{\mathrm{loc}} = \langle E^\mu \rangle_C$ and E^μ is the local regression MSE for the μ-th category, defined as

$$E^\mu = \min_{(W^\mu, b^\mu)} \left[\left\langle \left(W^{\mu\top} x^\mu + b^\mu - y^\mu \right)^2 \right\rangle_\mu \right] \tag{9}$$

where $W^\mu \in \mathbb{R}^{\mathrm{N}}$, $b^\mu \in \mathbb{R}$ are the local regression vector and bias for category μ. It can be shown (see SI) that $\Delta E = E - E_{\mathrm{loc}}$ is given by

$$\Delta E = \min_W \left[\left\langle \left\langle \left(W^\top x^\mu - \hat{y}^\mu \right)^2 \right\rangle_\mu \right\rangle_C \right] \tag{10}$$

where $\hat{y}^\mu = W_*^{\mu\top} x^\mu + b_*^\mu$ is the label predicted by the local regressor for category μ, with the local regressor defined as $(W_*^\mu, b_*) = \mathrm{argmin}_{(W^\mu, b^\mu)} \left[\left\langle \left(W^{\mu\top} x^\mu + b^\mu - y^\mu \right)^2 \right\rangle_\mu \right]$. In other words, the local-global error gap is itself a global regression error in which the target is the locally linearized label $\hat{y}^\mu$ rather than the original label y^μ.

Generative model. We derive ΔE for the following generative model of manifolds. We first consider the case in which each manifold and associated labels are locally centered, that is $\langle x^\mu \rangle_\mu = 0$ and $\langle y^\mu \rangle_\mu = 0$, $\forall \mu$. Without loss of generality, we parameterize a point in the μ-th manifold as

$$x^\mu = \sum_{d=1}^{N} s_d^\mu u_d^\mu \tag{11}$$

where $\left\{ u_d^\mu \in \mathbb{R}^N \right\}_{d=1}^N$ is an orthonormal basis and $s_d^\mu \in \mathbb{R}$ is the coordinate of the point along direction d. We choose u_1^μ to correspond to the feature-encoding direction, that is $u_1^\mu = W_*^\mu / \|W_*^\mu\|$. We choose the remaining directions $u_{d>1}^\mu$ to be the principal-component directions of the manifold in the subspace orthogonal to u_1^μ, so that the covariance matrix of the coordinates has the form

$$\left\langle s^\mu s^{\mu\top} \right\rangle_\mu = \begin{matrix} \sigma_1^{\mu 2} & \Sigma_{12}^\mu & \dots & \Sigma_{1N} \\ \Sigma_{21}^\mu & \sigma_2^{\mu 2} & 0 & 0 \\ \vdots & 0 & \ddots & 0 \\ \Sigma_{N1} & 0 & 0 & \sigma_N^{\mu 2} \end{matrix} \tag{12}$$

where $s \in \mathbb{R}$ is the vector of coordinates whose d-th component is s_d^μ. Note that local centering of the manifolds implies $\langle s_d^\mu \rangle = 0$. This choice of parameterization can always be made without loss of generality. We then make the following assumptions on the across-manifold statistics:

1. All manifold directions with $d > 1$ are mutually uncorrelated and isotropically distributed in neuronal space, conditioned on being orthogonal to u_1^μ (as by their definition as the elements of an orthonormal basis). Concretely,

$$\langle u_d^\mu \rangle_C = 0$$

$$\left\langle u_d^\mu u_{d'}^{\mu\top} \right\rangle_C = \frac{1}{N} \delta_{d,d'} \left(\mathbb{I} - \left\langle u_1^\mu u_1^{\mu\top} \right\rangle_C \right). \tag{13}$$

2. The feature-encoding directions u_1^μ are isotropically distributed around a common mean direction

$$u_1^\mu = \sqrt{a} u_1 + \sqrt{1-a} \eta^\mu \,, \tag{14}$$

where $a \in [0,1]$ is the alignment strength, $u_1 \in \mathbb{R}^N$ is a unit-norm vector representing the common mean direction, and $\eta^\mu \in \mathbb{R}^N$ is a random i.i.d. Gaussian vector with $\langle \eta^\mu \rangle_C = 0$ and $\left\langle \eta^\mu \eta^{\mu\top} \right\rangle_C = \frac{1}{N}\mathbb{I}$. Under this definition, u_1^μ has unit norm on average, i.e. $\left\langle u_1^{\mu\top} u_1^\mu \right\rangle_C = 1$, with fluctuations that vanish as powers of $1/N$, which is sufficient for our derivations in the limit $N \to \infty$.

3. Aside from the above correlations, all manifold parameters are otherwise uncorrelated.

Theory results. Our theoretical results are derived in the asymptotic regime $P \to \infty$ with N large but finite. As explained in the Discussion, this is the relevant regime for the problem of regression of category independent features. Under the generative model described above, and neglecting higher order corrections in N, we obtain (see SI for the derivation).

$$\Delta E = \hat{\sigma}^2 \left[1 - \frac{1}{(1+E_\mathrm{s})(1+E_\mathrm{o})} \right] \tag{15}$$

$E_\mathrm{s} = \frac{\mathrm{Var}_\mu[\|W_*^\mu\|]}{\mathrm{Avg}_\mu[\|W_*^\mu\|]^2}$, $E_\mathrm{o} = \frac{1-a}{N \cdot a \cdot \mathrm{SNR}}$, $\mathrm{SNR} = \frac{\langle \sigma_1^{\mu 2} \rangle_C}{\sum_{d=1}^N \langle \sigma_d^{\mu 2} \rangle_C}$, and $\hat{\sigma}^2 = \left\langle \left\langle \hat{y}^{\mu 2} \right\rangle_\mu \right\rangle_C$. In the case in which manifolds and labels are not locally centered, we quantify the contribution of category centroids through the centroid error, defined as

$$E_\mathrm{c} = E - E_\mathrm{fl}, \tag{16}$$

where

$$E_\mathrm{fl} = \min_W \left[\left\langle \left\langle \left(W^\top \delta x^\mu - \delta y^\mu \right)^2 \right\rangle_\mu \right\rangle_C \right], \tag{17}$$

with $\delta x^\mu = x^\mu - \langle x^\mu \rangle_\mu$ and $\delta y^\mu = y^\mu - \langle y^\mu \rangle_\mu$ denoting the locally centered representations and labels. That is, E_c measures the additional regression error induced by the presence of category-dependent centroids. The remainder of the theory is then derived exactly as above, with E_fl taking the role of E.

Theory for finite number of categories. We also derive the local-global gap ΔE for finite number of categories. Formally, the result is derived in the regime of finite category density, by studying the thermodynamic limit $P, N \to \infty$ at fixed $\alpha = \frac{P}{N} \in (0, \infty)$. The result for infinitely many categories presented above ($P \to \infty$ with N large but finite) is recovered as the limiting case $\alpha \to \infty$. The expression for generic α is lengthy and is provided in the SI, alongside its derivation.

Fitting theory to data. The theory derived above applies to the *training* regression error. For a validation of the theory in this setting, using the generative model described above, see the SI. In the main text, instead, we test its predictions against the *test* regression error measured from the empirical data, and find good agreement in this case as well. This agreement assumes that the

number of images is sufficiently large and that the regularization of the fitted regressor is appropriate, so that the train-test generalization gap is small and the theoretical prediction remains informative for the test error. For the purpose of estimating the theory parameters from data, the local regressors are taken to be the regularized empirical solutions used in practice, as determined by the procedure described in the 'Linear Decoding' section of the Methods, rather than the unregularized minimizers of the local MSE. Accordingly, all quantities entering the theory that depend on the local regressors are evaluated using these regularized estimates.

Supplementary information

IV. SUPPLEMENTARY RESULTS

A. Linear decoding performance in macaque recordings

We have applied the linear decoding framework presented in the main text to the publicly available dataset of macaque recordings [34] studied in [20], as well as to an ImageNet-pretrained ResNet-50 CNN. We reproduce their qualitative results (**Fig. 9a**): decoding performance of both category and category-independent features improves along the visual hierarchy. However, we also find that absolute regression performance remains limited (**Fig. 9b**): in macaque IT, as well as in the CNN for most features, root nMSE remains above 0.5, meaning that the root-mean-square prediction error is still larger than half the across-image standard deviation of the decoded feature.

We note that this last point may be less apparent in the original presentation of [20], where regression performance was reported using the Pearson correlation coefficient. Here we instead report $\sqrt{\text{nMSE}}$, which we find more directly interpretable, as it measures the typical prediction error (root-mean-squared error) in units of the standard deviation of the decoded feature. In particular, because Pearson correlation is a nonlinear function of nMSE, it tends to compress even appreciable differences in nMSE into a narrow interval near 1. Indeed, for ordinary least-squares (no ridge) evaluated on the training set, Pearson correlation is related to nMSE by $\text{Pears} = \sqrt{1 - \text{nMSE}}$. For ridge regression on held-out data, this relation is only approximate but still useful as an intuition. We can see that Pearson correlation compresses a broad range of $\sqrt{\text{nMSE}}$ values into a narrow range near 1. For example, $\sqrt{\text{nMSE}} \in [0, 0.5]$—from perfect prediction to errors on the order of half a standard deviation—maps to $\text{Pears} \in [0.87, 1]$, making differences in regression accuracy harder to visualize and interpret.

Fig. 9b additionally reports regression performance using the Pearson correlation coefficient, alongside nMSE, to facilitate a more direct comparison with [20]. The Pearson values we obtain are comparable to those reported in that work (cf. Supplementary Fig. 3 and Supplementary Fig. 9 of [20]). This comparison is not exact, because the original analysis was performed on the full private dataset, whereas we here use the publicly available subset. Relative to the public dataset, the full dataset contained 320 additional images per category, as well as 98 additional IT units recorded from a third macaque. Even so, the Pearson-based regression performance reported here remains of similar magnitude to that reported in [20]. In particular, the IT Pearson values in [20] are still sufficiently below 1 to be consistent with a root nMSE well above 0.5, matching the conclusions we draw here from the public data.

B. Regression performance for network CRloc

Here we show that optimization for regression alone is sufficient to substantially reduce the local-global gap, even in the absence of the additional constraint imposed by a shared readout across categories. Indeed, we show that ΔE is largely reduced not only in network CR, which is optimized for *category-independent* regression and classification, but also in a network that we term CRloc, which we optimized for *category-specific* regression and classification (**Fig. 10a-b**). In CRloc, category-specific regression is implemented by attaching multiple regression heads to the feature layer—one for each category—rather than a single head shared across categories. We observed that, in network CRloc, ΔE is largely reduced compared to network C, down to values only slightly higher than network CR. The same observation holds for the three sources of error E_c, E_s, E_o (**Fig. 10c**), and the two factors contributing to E_o, namely alignment a of the feature-encoding directions, and SNR (**Fig. 10d**).

C. Changes in manifold geometry due to regression optimization

1. *Changes in manifold shapes (CR vs C)*

In the main text, we showed that, on average, manifold shapes are only weakly altered in network CR relative to network C (cf. **Fig. 6b**). Here we examine this effect on a category-by-category basis. For a given category μ, let $\{\lambda_d^\mu\}_{d=1}^N$ and $\{\gamma_d^\mu\}_{d=1}^N$ denote the PC eigenvalues of the μ-th category manifold in networks CR and C, respectively. We quantify the spectral difference between these two manifolds as

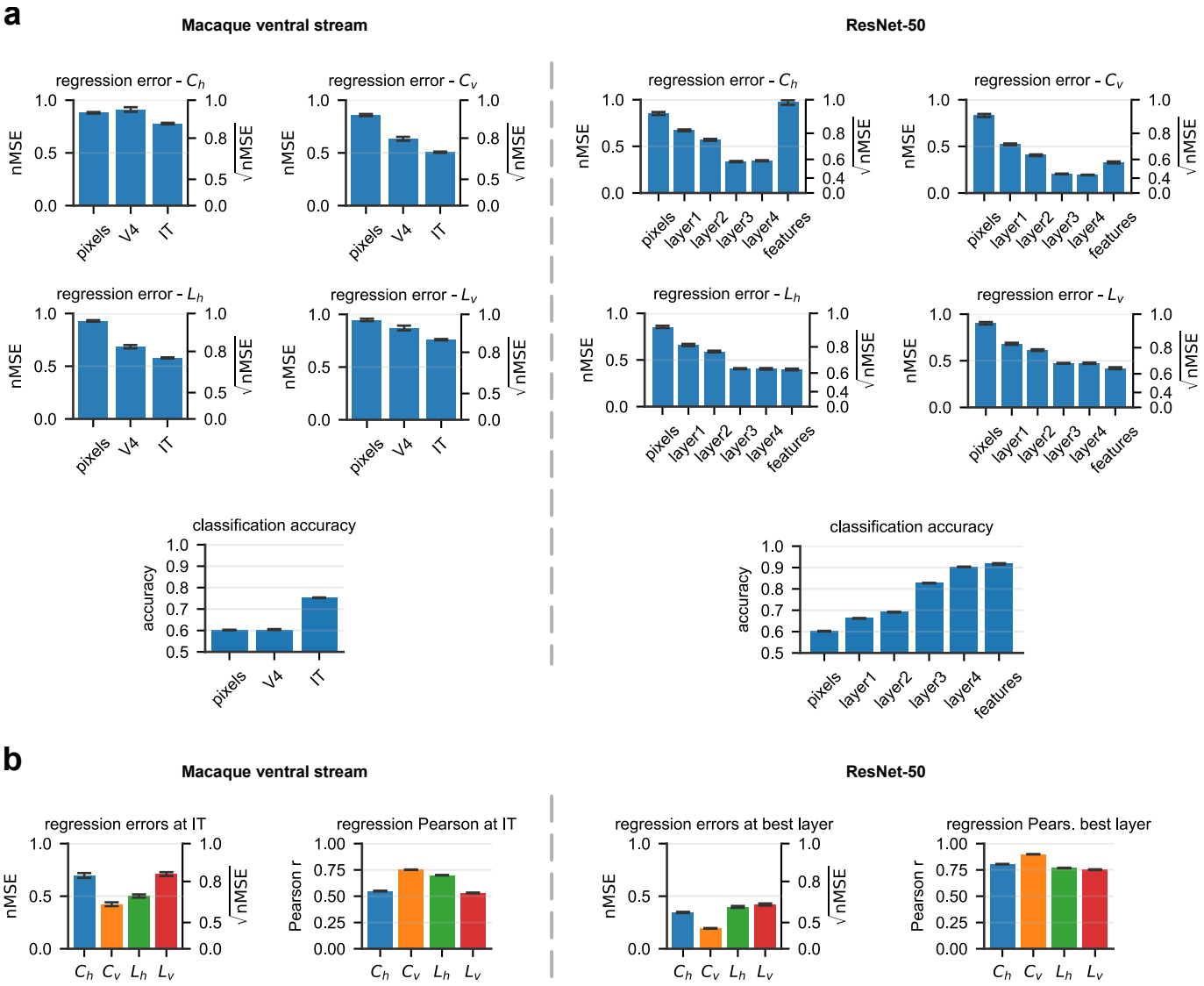

Figure 9. Linear decoding performance for macaque neural recordings and ResNet-50 activations. (**a**) Linear decoding performance along the visual hierarchy, for macaque neural recordings (left column) and ResNet-50 activations (right column). For a fair comparison across regions, the 168 IT units are randomly subsampled to match the 88 units in V4. Similarly, units in layers 1–4 of ResNet-50, as well as the pixels layer, are randomly projected to match the 2048 units in the final feature layer. (**b**) The regression performance at IT using all 168 units (left column), and error at the best performing layer of ResNet-50 (right column). We report performance both in terms of the nMSE (left) and the Pearson coefficient (right). Error bars indicate the standard error of the mean (SEM) across cross-validation splits and, where applicable, across random subsamples of units.

$$d^{\mu} = \mathcal{N}^{-1}\left[\sum_{d=1}^{N}\left(\sqrt{\lambda_d^{\mu}} - \sqrt{\gamma_d^{\mu}}\right)^2\right]^{1/2}, \qquad \mathcal{N} = \left[\sum_{d=1}^{N}\left(\lambda_d^{\mu} + \gamma_d^{\mu}\right)\right]^{1/2}. \tag{18}$$

Without the normalization factor $\mathcal{N}$, this quantity corresponds to the Wasserstein-2 distance between the covariance matrices of category μ in networks C and CR, after aligning their principal axes, which is appropriate here since our goal is to compare manifold shapes independent of their orientations. Intuitively, the Wasserstein-2 distance measures how much one distribution must be rearranged to match the other. Here we normalize this distance by the combined root variance of the two manifolds, $\mathcal{N}$, so that the resulting quantity is dimensionless, bounded in the range $d^{\mu} \in [0, 1]$, and comparable across categories. **Fig. 11** reports the histogram of d^{μ} across categories, together with example PC

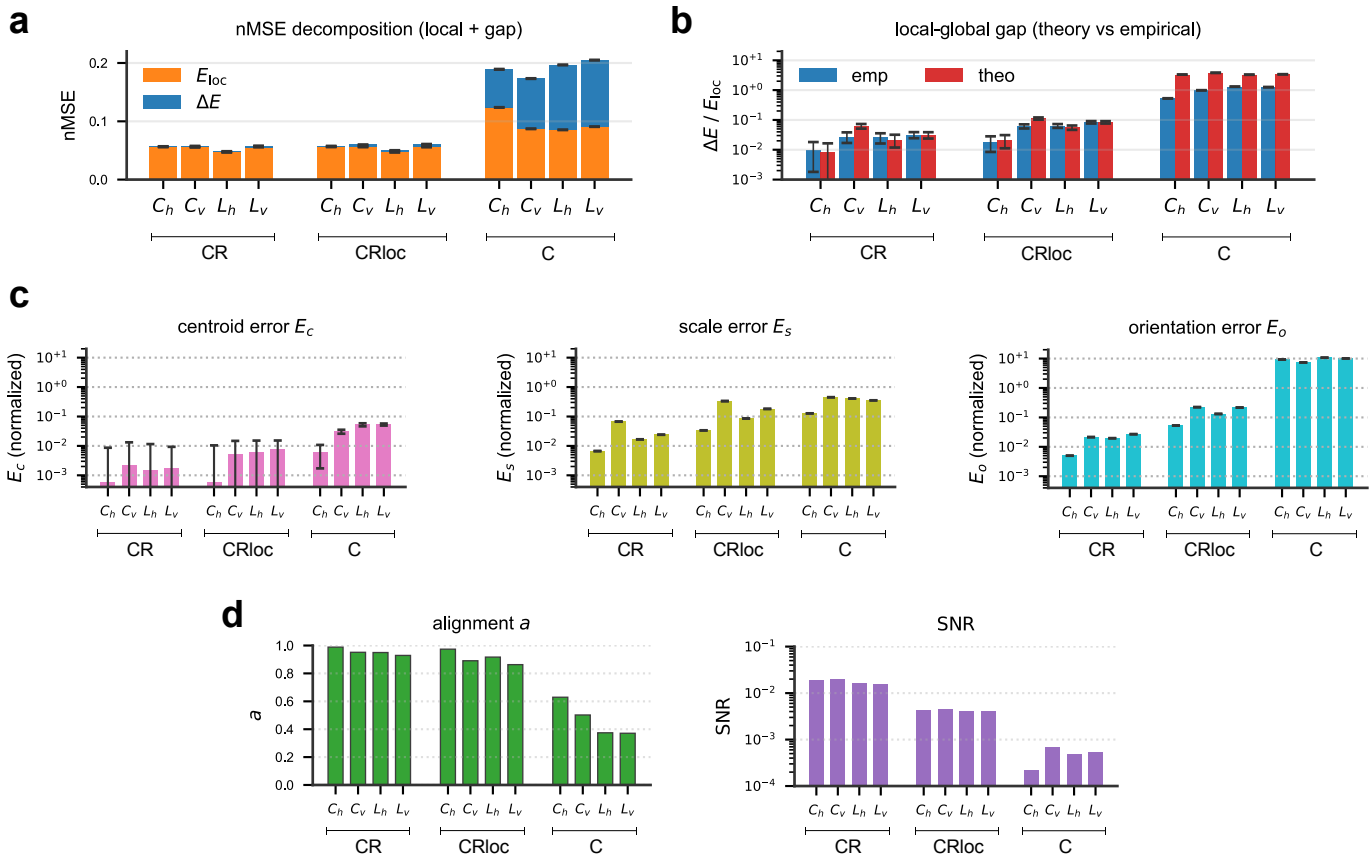

Figure 10. Regression performance in networks CR, CRloc and C. (**a**) Decomposition of the nMSE into local error E_{loc} (orange) and local–global gap ΔE (blue) for regression of the four bounding box coordinates. Note that ΔE is very small and barely visible for networks CR and CRloc. (**b**) Measured relative local-global gap $\Delta E/E_{\text{loc}}$ (blue) compared to our theory prediction (red). (**c**) Decomposition of the relative gap $\Delta E/E_{\text{loc}}$ into the contributions from the centroid error E_{c} (pink) scale error E_{s} (olive) and orientation error E_{o} (cyan). The three errors are normalized as explained in the caption of **Fig. 5**. (**d**) The two elements contributing to the orientation error: SNR, and alignment a of the local feature-encoding directions.

spectra for selected categories in networks C and CR. These results show that manifold shapes undergo only small relative changes between networks C and CR, with $\text{Avg}_\mu[d^\mu] \sim 0.083$.

2. *Changes in centroid norm and separation (CR vs C)*

For each category manifold μ, we denote by $\bar{x}_\mu \in \mathbb{R}^N$ its centroid. We compare between networks C and CR two centroid measures: the centroid norm $\|\bar{x}_\mu\|$, and the pairwise relative centroid distance $\|\bar{x}_\mu - \bar{x}_\nu\|/\langle\|\bar{x}_\rho\|\rangle_\rho$ between two manifolds μ and ν. Centroid norms see only modest changes in network CR relative to network C, while centroid distances are virtually unchanged in distribution (**Fig. 12**).

3. *Differences and similarities across all networks C, CR, CRloc, R*

In the main text and in the two SI subsections above, we analyzed changes in the feature-layer manifold geometry of network CR compared to C. Here we extend that analysis also to networks CRloc and R. We further quantify representational similarity across all of these networks using centered kernel alignment (CKA; [35]) and their alignment to neural data using the public Brain-Score benchmarks [22].

Fig. figure 13a compares manifold shapes across these networks, reporting their average PC spectrum (left) as defined in the main text, their average dimensionality (middle), and normalized radius (right). We define dimensionality

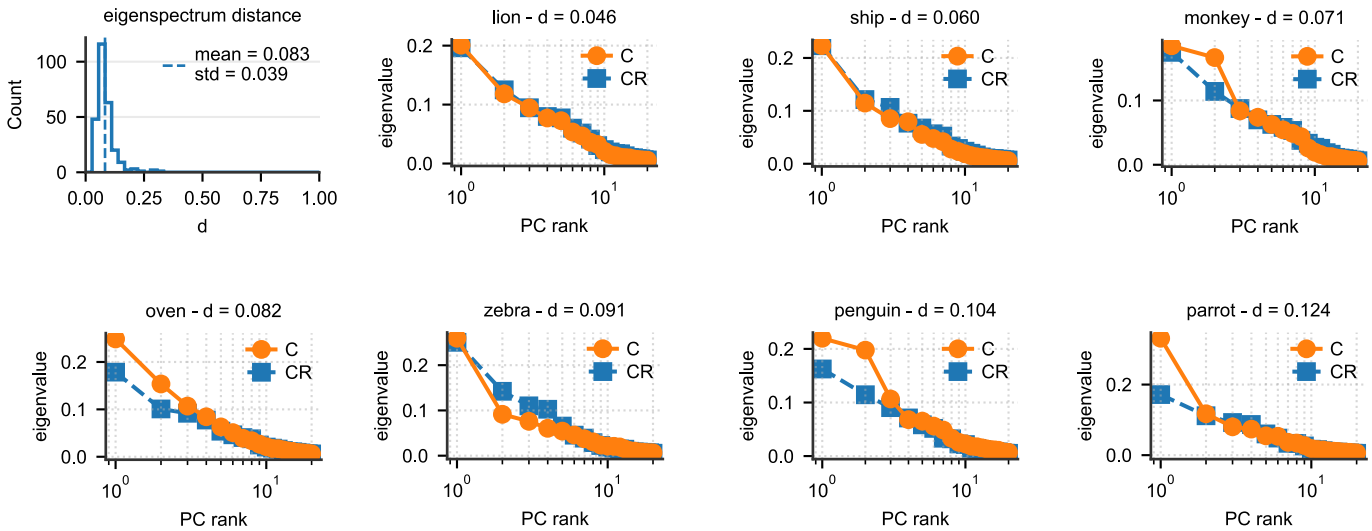

Figure 11. Change in manifold shapes between networks C and CR. Top left: histogram across categories of the spectral distance d^μ. Other panels: PC spectra for selected categories in networks C and CR. Categories were sampled randomly, with sampling probability weighted by a Gaussian distribution with mean and standard deviation equal to those of the empirical distribution of d^μ. For each category, the PC spectra are normalized by the category total variance, averaged between networks C and CR.

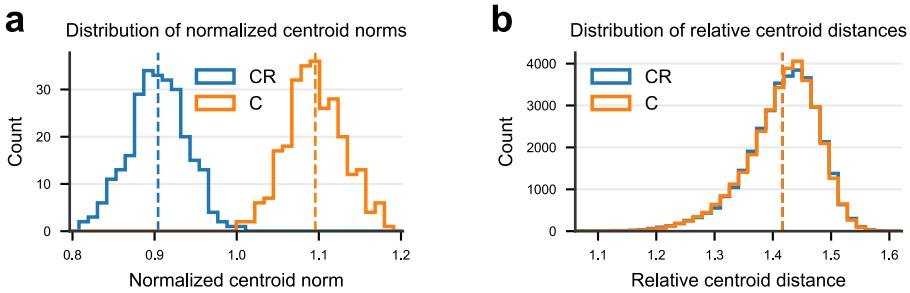

Figure 12. Manifold centroid norm and separation, in networks C (orange) and CR (blue). (a) Histogram of centroid norms $\|\bar{x}_\mu\|$. Centroid norms are normalized by their average across categories and across the two networks. (b) Histogram of relative centroid distances $\|\bar{x}_\mu - \bar{x}_\nu\|/\langle\|\bar{x}_\rho\|\rangle_\rho$. Vertical dashed lines indicate the corresponding averages.

by the participation ratio (PR): denoting by λ_d^μ, with $d = 1, \ldots, N$, the PC eigenvalues of the covariance matrix of the μ-th category manifold, its participation ratio is defined as$\mathrm{PR}^\mu = \frac{\left(\sum_{d=1}^N \lambda_d^\mu\right)^2}{\sum_{d=1}^N \left(\lambda_d^\mu\right)^2}$. We define the radius of the μ-th category manifold as $R^\mu = \left(\frac{\sum_{d=1}^N \left(\lambda_d^\mu\right)^2}{\sum_{d=1}^N \lambda_d^\mu}\right)^{1/2}$. In the figure, we report the radius normalized by the average centroid norm, $R^\mu/\langle\|\bar{x}^\nu\|\rangle_\nu$, which is the relevant quantity for comparisons across networks, since the problem of linear separability of categories is invariant under an overall rescaling of the neural representation space. **Fig. figure 13b** Compares properties of manifold centroids across networks: average centroid norm, and centroid separation, as defined in section IV C 2. **Fig. figure 13c** reports average mutual alignment between manifolds in each network, as defined in the main text. Finally, **Fig. figure 13d** reports linear CKA between each pair of networks [35]. We report the debiased version obtained by replacing the Hilbert-Schmidt Independence Criterion with its unbiased estimator as reported in [60]. In practice, because our dataset contains a large number of images, we found no appreciable difference between CKA and debiased CKA.

We begin by comparing networks C and CR. For these two networks, all the quantities reported in **Fig. figure 13a-c** have already been discussed in the main text and in the previous SI subsections. The main conclusion is that, under the constraint imposed by the classification objective, regression optimization leaves only a subtle signature on coarse measures of manifold geometry, such as manifold shape, centroid positions, and mutual alignment. Here we further show that this is reflected in a very high CKA score (CKA ~ 0.95) between the feature-layer representations of the two networks. Similarly high CKA similarity was also reported in a previous study [23].

This picture changes dramatically when the network is optimized exclusively for regression (network R). Without the constraint imposed by the classification objective, manifold dimensionality drops to values close to four—the number of encoded features—and manifold alignment increases drastically (**Fig. figure 13a**, middle, and **c**). The normalized radius also becomes much larger than in networks C and CR (**Fig. figure 13a**, right), due to a strong reduction in the average centroid norm **figure 13b**), which may explain the poor classification performance of network R. The drastic change in overall manifold geometry is reflected in a very low CKA similarity with both networks C and CR (CKA < 0.1, **Fig. figure 13d**).

Finally, we consider category manifolds in network CRloc, which appear to share some traits with both networks C/CR and network R. Network CRloc has an average manifold shape similar to that of networks C and CR, except for the ~ 4 largest PCs, which are more prominent in CRloc (**Fig. figure 13a**, left). This dominance of the top ~ 4 PCs is reminiscent of network R, with the difference that in CRloc these components emerge on top of a baseline PC spectrum similar to that of networks C and CR, whereas in network R the remaining PCs are largely suppressed. As a consequence, the dimensionality of CRloc, as measured by the participation ratio, is more similar to that of network R. The manifolds in CRloc also display a high degree of mutual alignment, comparable to that of network R (**Fig. figure 13c**), which is likely driven by a strong alignment of the top 4 PCs. At the same time, centroid norm and centroid separation (**Fig. figure 13b**), as well as normalized radius (**Fig. figure 13a**, right), remain much more similar to those of networks C and CR, consistent with the fact that CRloc is also trained to achieve good classification performance. This mixed manifold geometry of network CRloc is reflected in intermediate CKA similarity scores with all three of the other networks (**Fig. figure 13d**).

Fig. figure 14 reports neural-data alignment of networks C, CR, R, and an ImageNet-pretrained ResNet-50, as scored by the public Brain-Score benchmarks [22]. We include network R and the ImageNet-pretrained ResNet-50 mainly as controls. As noted in the main text, networks C and CR (as well as the ImageNet-pretrained ResNet-50) exhibit highly similar alignment scores across layers and regions, consistent with their high similarity under coarse manifold-geometry measures (manifold shapes, centroid norms and separation, and manifold alignment) and CKA. By contrast, for some regions such as V2 and IT, network R departs appreciably from the other networks only in the final layers. This is consistent with our geometric analysis: network R differs strongly from networks C and CR at the final feature layer under coarse manifold-geometry measures and CKA, whereas earlier layers remain much more similar.

D. Across-layer trends do not depend on the layer-freezing training schedule

In training the CNNs, we found that gradually unfreezing the weights of the ResNet-50 layers through the epochs, from the feature layer downstream to layer 1, yielded models with slightly better regression performance. In particular, the models reported in the main text were trained by unfreezing a new layer each epoch, in sequence: the feature layer, layer 4, and layer 3, while layers 2 and 1 were kept frozen throughout the training. Here in the SI, we show that the trends across layers reported in the main text (cf. **Fig.** 7) are virtually unchanged when network CR is trained end to end with all layers unfrozen from the very first epoch, rather than with the layer-unfreezing schedule described above. We call this network CR*. **Fig. 15** shows regression performance as a function of layer depth for networks CR, C and CR*. We still observe the trends reported in the main text: a sharp rise in regression performance emerges only from layer 4, as well as an appreciable distinction in regression performance between networks C and CR/CR*.

E. Further details on the effect of subsampling neural units

Effect of subsampling on the local error

In the main text, we examined the effects of subsampling object categories P and neural units N on the local-global error gap ΔE. Here we report the effect of subsampling on the other component of the global regression error: the local error E_{loc}. Because the local error E_{loc} measures within-category regression performance, subsampling the number of categories does not systematically affect it, aside from increased estimation noise. We therefore focus on the effect of subsampling neural units, which we report in **Fig. figure 16**.

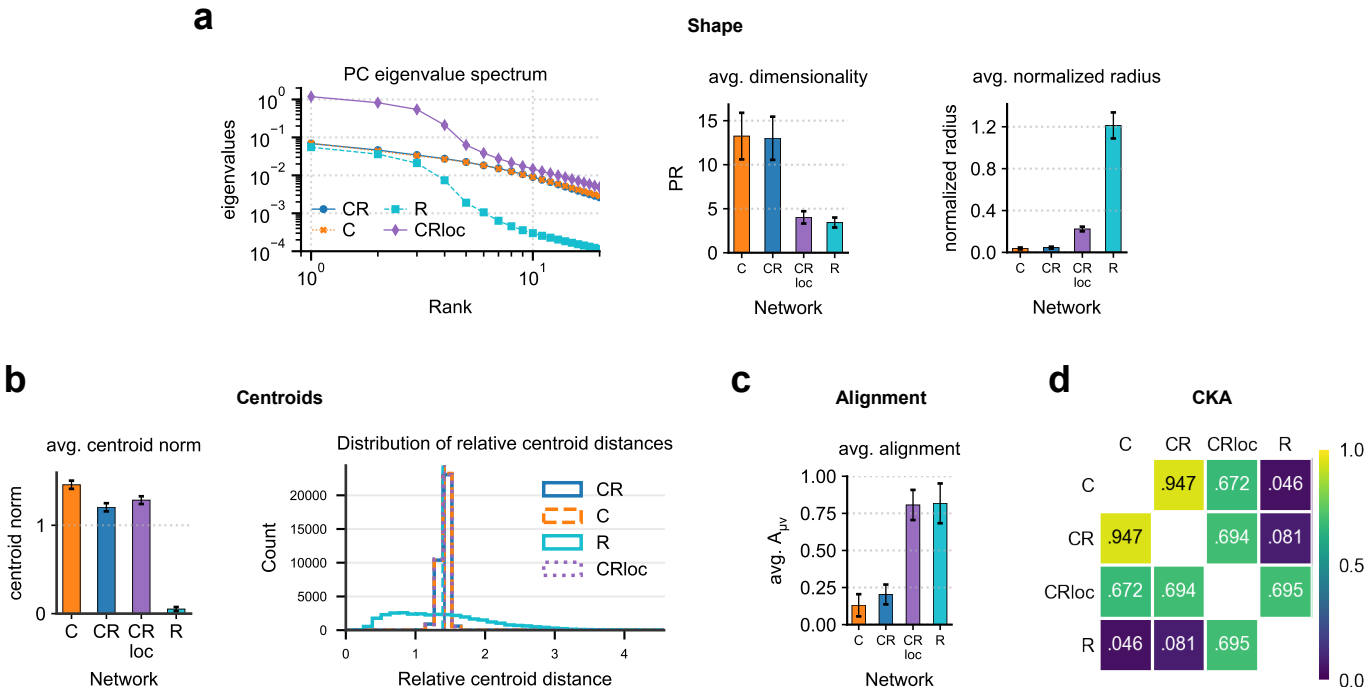

Figure 13. Differences in manifold geometry between networks C, CR, R, and CRloc. (**a**) Manifold shapes. Left: manifold PC spectrum, averaged across categories (error bars are SEM). For each manifold μ, the eigenvalues were normalized by the manifold total variance $\mathrm{Tr}\Sigma^{\mu}$, averaged across networks C, CR, R, and CRloc. Middle: manifold dimensionality (PR), averaged across categories. Error-bars are standard deviation across categories. Right: same as middle, but for normalized radius. (**b**) Centroids. Left: same as (a, middle/right), but for centroid norm. Right: histogram of relative centroid distances $\|\bar{x}_\mu - \bar{x}_\nu\|/\langle\|\bar{x}_\rho\|\rangle_\rho$. Vertical dashed lines indicate the average across all category pairs. (**c**) Same as (a, middle/right), but for the average mutual alignment, i.e., average of the off-diagonal entries of the global alignment matrix A. (**d**) Debiased CKA between all four networks.

Effect of limited images per category on the estimation of the local-global gap

In the main text, we reported $\Delta E/E_{\mathrm{loc}}$ for regression of bounding-box features from the neural representations elicited by the image set used in [20, 34] to obtain macaque ventral-stream recordings (**Fig. 8c**). Specifically, we showed results for the recorded units in IT and V4, as well as for networks C and CR subsampled to 168 units, matching the number recorded in IT. Since all units are available in C and CR, one could in principle also study how $\Delta E/E_{\mathrm{loc}}$ varies with the number N of subsampled units, mirroring the analysis performed in the main text on our image dataset (**Fig. 8b**). However, this analysis is much less informative for the image set used in [20, 34], because that dataset contains only 400 images per category. When N becomes comparable to, or larger than, 400, the local regressors begin to overfit, artificially increasing the estimated local error E_{loc} and thereby underestimating the local–global gap. This effect is clearly visible in **Fig. 17**: beyond sufficiently large N, the local error reverses its expected downward trend and instead increases with N. As a consequence, the estimated local–global gap is progressively underestimated and can even become negative. More generally, this analysis highlights an important requirement for future experiments aiming to estimate the local–global error gap reliably: the number of images per category should exceed the number of recorded units, so as to avoid this overfitting regime.

F. Main text results for other bounding-box features

Here we report the same kind of results presented in the main text, for features that were not shown in the main text.

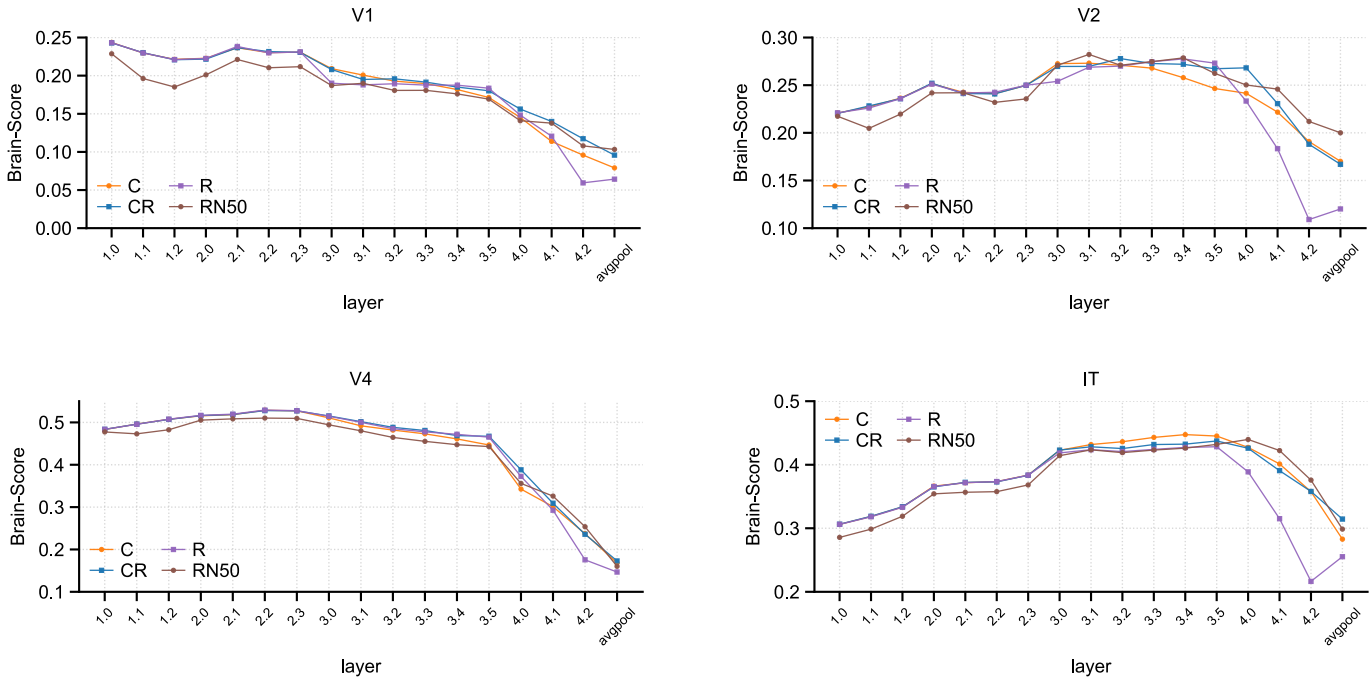

Figure 14. Neural-data alignment on public Brain-Score ventral-stream benchmarks. For each layer of networks C, CR, R, and an ImageNet-pretrained ResNet-50, we report the Brain-Score alignment to four publicly available neural benchmarks [34, 61]: FreemanZiemba2013.V1-pls (V1), FreemanZiemba2013.V2-pls (V2), MajajHong2015.V4-pls (V4), and MajajHong2015.IT-pls (IT).

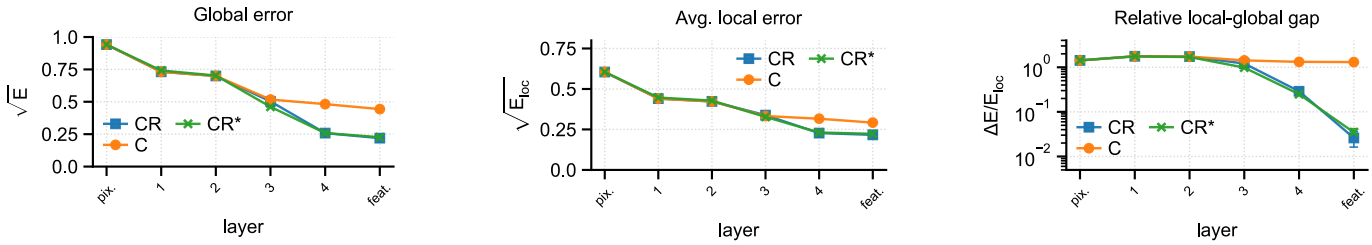

Figure 15. Regression performance along the visual hierarchy. Regression performance for the bounding-box parameter L_{h} using linear readouts from progressively deeper layers of networks C (orange circles, full line) and CR (blue squares, full line) and CR* (green cross, full line). At each layer, linear readouts were trained on activations randomly projected to 2048 dimensions, to match the dimensionality of the final feature layer. Note that global and local error are reported as root nMSE.

Feature-encoding direction vs PCs overlaps

Fig. 18 reports the overlap of the local feature-encoding direction with the top PCs of the corresponding category-manifold, averaged across categories. The trend is the same as discussed in the main text (cf. **Fig. 6**): in network CR compared to C, the feature-encoding directions are much more aligned with top manifold PCs.

Decoding performance along the visual hierarchy

Fig. 19 reports regression performance and associated theoretical measures for the bounding-box feature C_{h}. We focus on C_{h} because C_{v} shows the same qualitative behavior; similarly, in the main text we report L_{h}, while L_{v} follows an analogous trend. Overall, the across-layer behavior of center coordinates is similar to that of box lengths described in the main text (cf. **Fig. 7**), with one minor difference: in network C, the regression error for the center coordinates increases at the final transition from the last convolutional layer to the feature layer. A plausible explanation is that center coordinates rely more directly on the spatial arrangement of activity across the convolutional feature

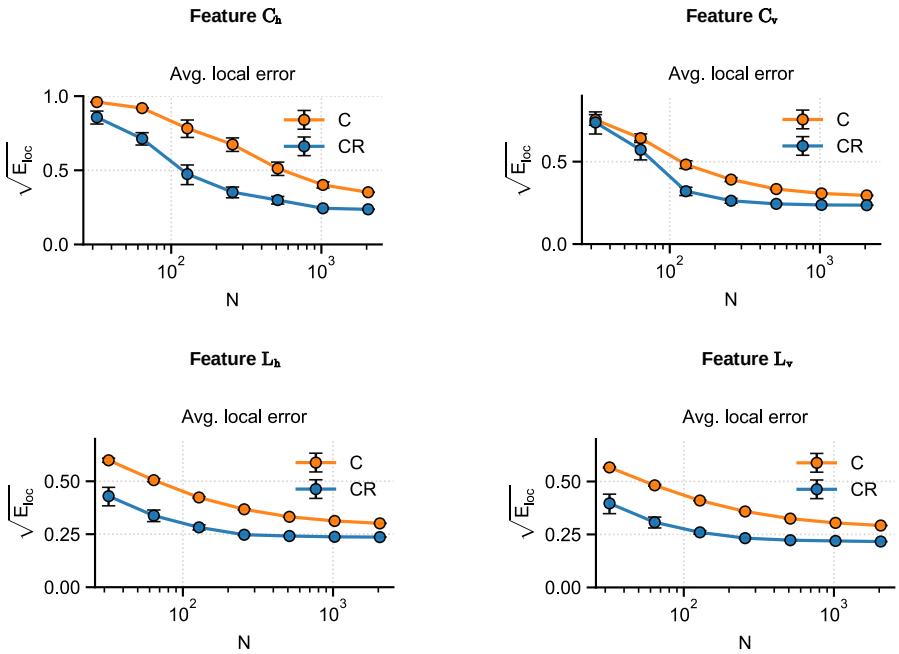

Figure 16. Local regression error E_{loc} for networks C (orange) and CR (blue) on our image dataset, as a function of the number N of randomly subsampled units (out of 2048 feature-layer units).

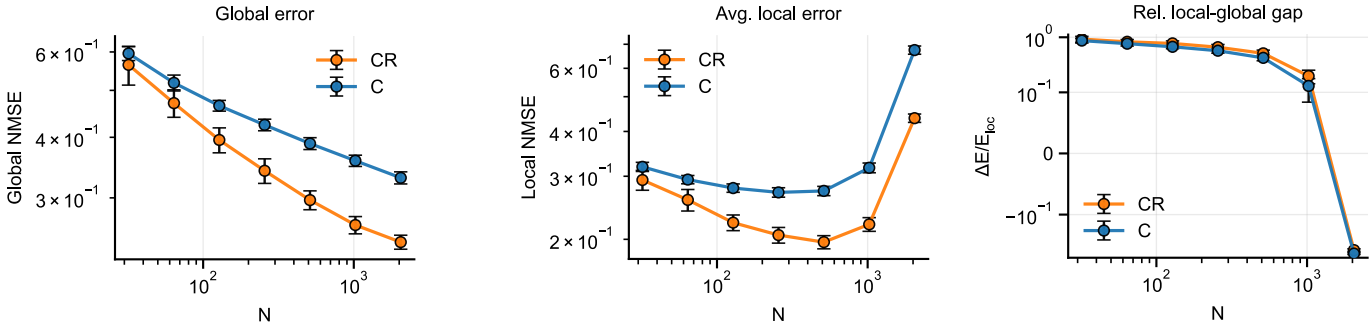

Figure 17. Global error, local error, and relative local–global gap in the regression of bounding-box parameter L_h, shown as a function of the number N of subsampled units, for networks C and CR on the image set used in [20, 34].

maps, because they encode *where* the object is located in the image. Global average pooling removes much of this spatial information by collapsing each feature map into a single scalar, and therefore can impair linear decoding of object position. By contrast, box lengths are invariant to object translation and thus may be less sensitive to the pooling operation. In network C, which is optimized only for classification, there is no pressure to recode positional information into a form that survives the final pooling step. In network CR, by contrast, explicit regression supervision can promote a reorganization of the last convolutional representation such that center-coordinate information remains linearly accessible after average pooling.

Subsampling effects

Fig. 20 reports the effect of subsampling neural units on the two components of global regression error—local error E_{loc} and local-global gap ΔE—for the bounding-box features not shown in the main text. The trend is the same as observed in the main text (cf. **Fig. 8b**): for low enough number of subsampled units N, the local-global gap ΔE in networks C and CR becomes indistinguishable. **Fig. 21** reports the analysis of regression performance on the images and corresponding neural recordings in [34], for the other bounding-box features not shown in the main text. The trend is the same as observed in the main text (**cf. Fig. 8c**): IT, V4, as well as networks C and CR subsampled

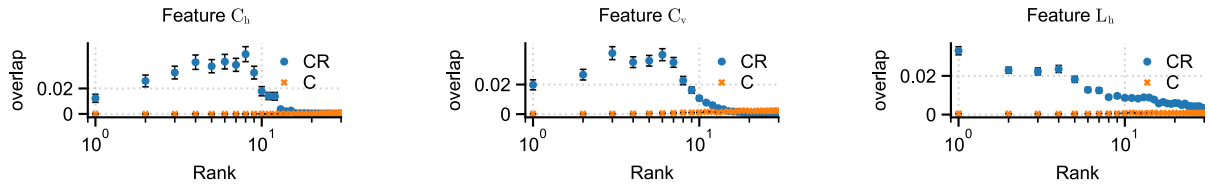

Figure 18. Feature-encoding direction vs PCs overlap. Squared cosine angle between the feature-encoding direction $\hat{w}^{\mu}$ and the PC directions of manifold μ (ordered by rank) for networks C (orange) and CR (blue), averaged across categories (error bars are SEM).

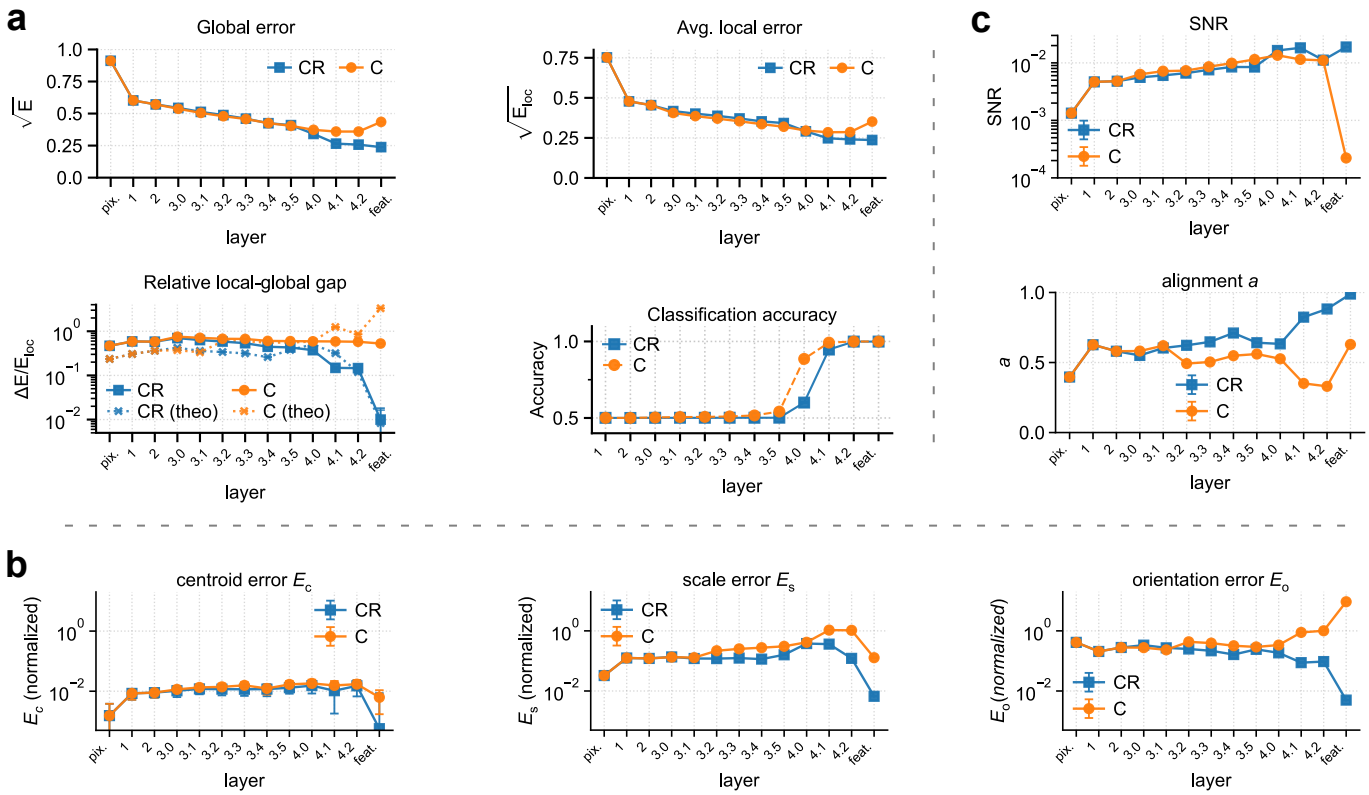

Figure 19. Decoding performance along the visual hierarchy. (**a**) Regression performance for the bounding-box parameter C_{h} and classification performance, using linear readouts from progressively deeper layers of networks C (orange circles, full line) and CR (blue squares, full line). For regression, we report the global regression error, as well as its decomposition into local error E_{loc} and relative local-global gap $\Delta E/E_{\mathrm{loc}}$. For the gap, we also report our theory prediction (x markers, dotted lines). Note that global and local error are reported as root nMSE. (**b**) The three contributions to the relative local-global gap, as identified by our theory: centroid error E_{c}, orientation error E_{o}, scale error E_{s}. These quantities are normalized in the same way as explained in **Fig. 5**. (**c**) The two elements contributing to the orientation error: SNR, and alignment a of the local feature-encoding directions.

to 168 units, matching the number recorded in IT, all show comparable values of $\Delta E/E_{\mathrm{loc}}$, as well as comparable contributions from each of the three error sources, with E_{o} and E_{c} emerging as the dominant terms.

V. IMAGE DATASET

Here we report additional example images from our dataset (**Figs. 22-25**) together with the bounding-box statistics of the full dataset.

Fig. 26a shows the global bounding-box statistics across all images and categories. To interpret this figure, recall the bounding-box sampling procedure. First, the maximum side length $L_{\max} = \max(L_h, L_v)$ is sampled uniformly,

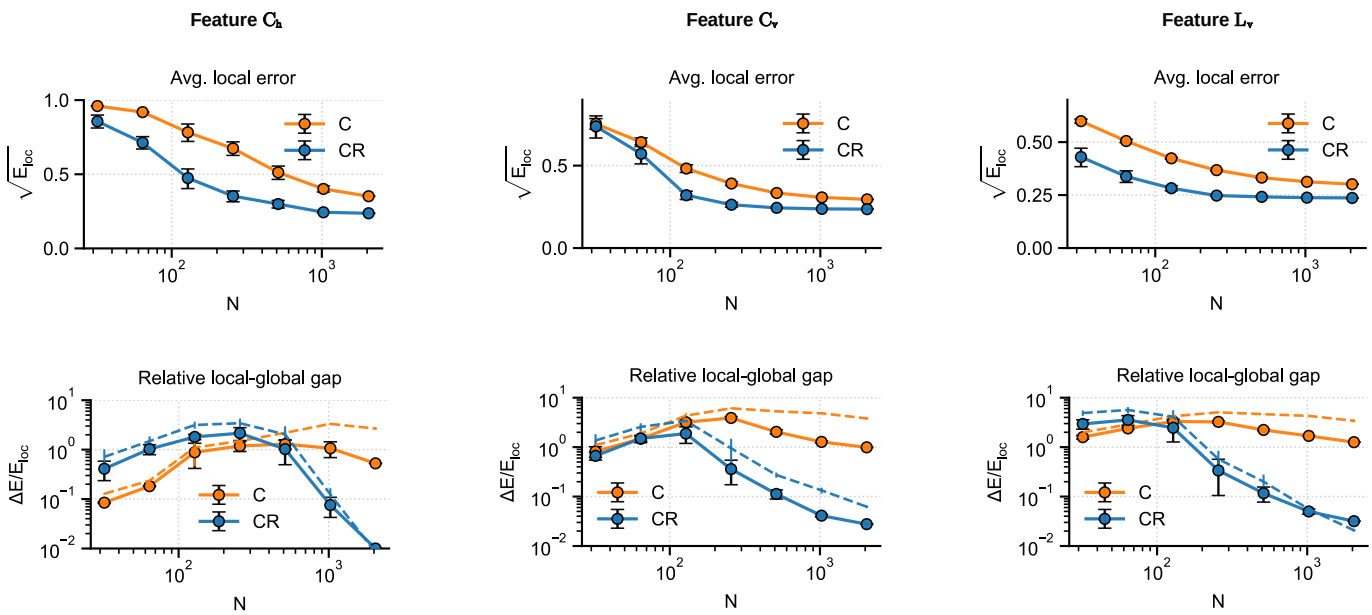

Figure 20. Effect of subsampling neural units. E_{loc} and $\Delta E/E_{\text{loc}}$ for regressing the bounding-box features (C_h, C_v, L_v) from networks C (orange) and CR (blue) on our dataset, as a function of the number N of randomly subsampled units (out of 2048 feature-layer units). For the gap, we also report our theory prediction (x markers, dotted lines). Error bars denote SEM across cross-validation splits and random unit subsamples.

$L_{\max} \sim \text{Unif}[200, 500]$, in pixel units (the full image is 512×512 pixels), consistent with the approximately uniform distribution shown in **Fig. 26a**, left. The aspect ratio L_h/L_v is then kept fixed with respect to the original seed image in order to preserve the natural appearance of the object. Across all categories, this yields an overall distribution of aspect ratios that is roughly symmetric around 1 (**Fig. 26a**, middle-left). The center coordinates are then sampled uniformly over all positions that keep the bounding box inside the canvas, namely $C_h \sim \text{Unif}[L_h/2,\, 512 - L_h/2]$ and $C_v \sim \text{Unif}[L_v/2,\, 512 - L_v/2]$. This produces distributions that are symmetric around the image center (**Fig. 26a**, middle-right and right). Note, however, that the center coordinates are more concentrated near the image center overall, because the allowed range of center positions depends on the bounding-box size. In particular, boxes with larger side lengths can occupy only a narrower range of center positions around the image center, since the box must remain fully inside the image. As a result, center coordinates closer to the image center occur more frequently in the dataset.

We also report bounding-box statistics on a per-category basis (**Fig. 26b**). The bounding-box sampling procedure does not depend explicitly on object category, except through the natural range of aspect ratios associated with different categories (for example, cars tend to be more extended along the horizontal axis, whereas penguins tend to be more extended along the vertical axis). This is reflected in the high uniformity across categories in the distributions of maximum side length and center coordinates. By contrast, aspect ratio is much more category-dependent, both in its within-category mean and in its within-category spread (standard deviation).

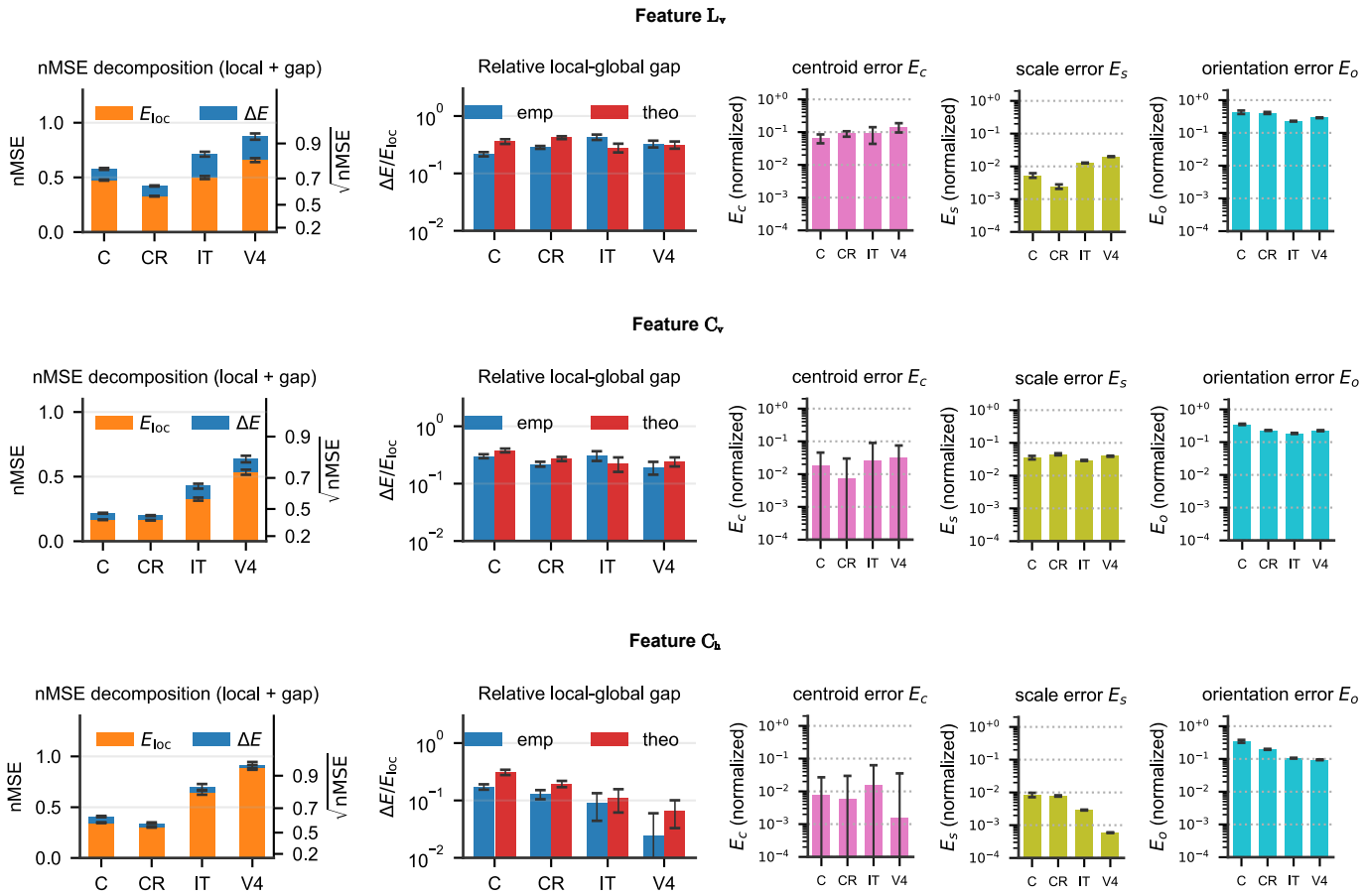

Figure 21. Analysis of regression performance on the images and corresponding neural recordings found in [34]. We report the analysis of regression from the IT and V4 units recorded in [34], as well as 168 randomly projected units from the best performing layer of networks C and CR. For features C_{h} and C_{v}, the best performing layer is layer 4, for feature L_v it is the feature layer. Error bars denote SEM across cross-validation splits and, where applicable, random unit subsamples.

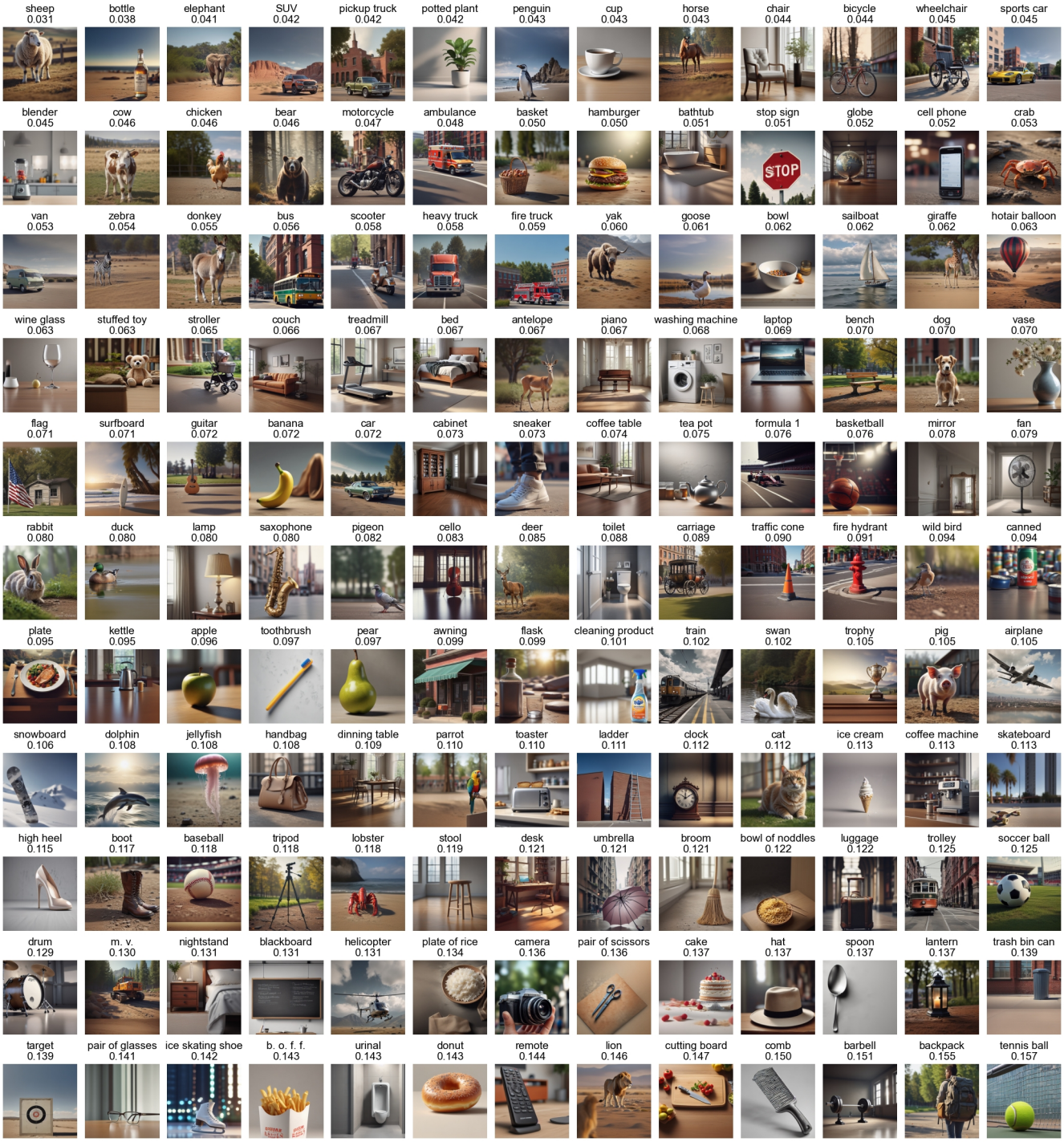

Figure 22. All categories in our dataset (part 1; continues in Fig. 23). One representative image per category is shown. Below the category label, we report the test regression $\sqrt{\text{nMSE}}$ achieved by network CR for that category, averaged across all images in the category and across the four bounding box parameters. Categories are ordered by ascending $\sqrt{\text{nMSE}}$.

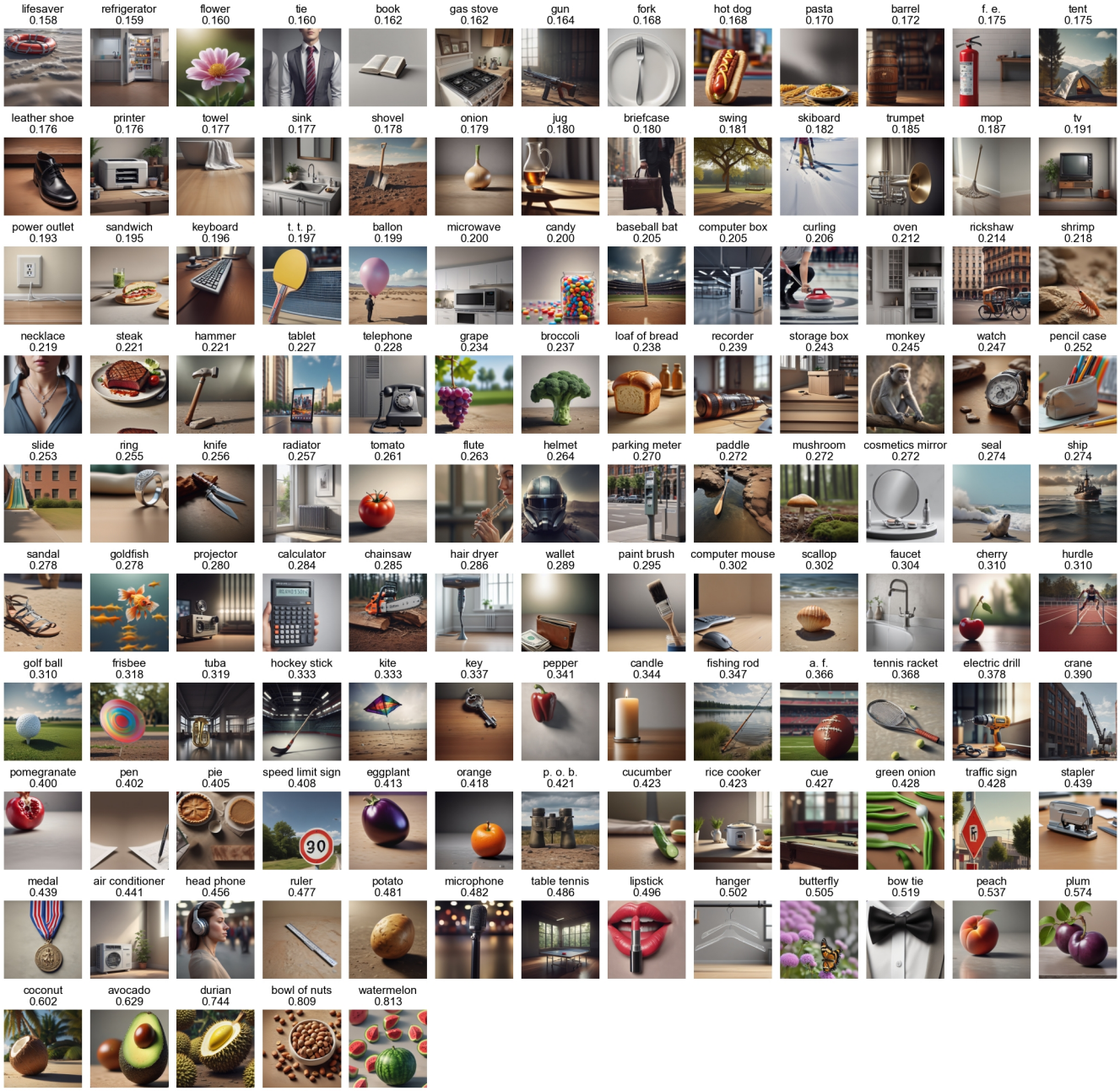

Figure 23. All categories in our dataset (part 2; see Fig. 22).

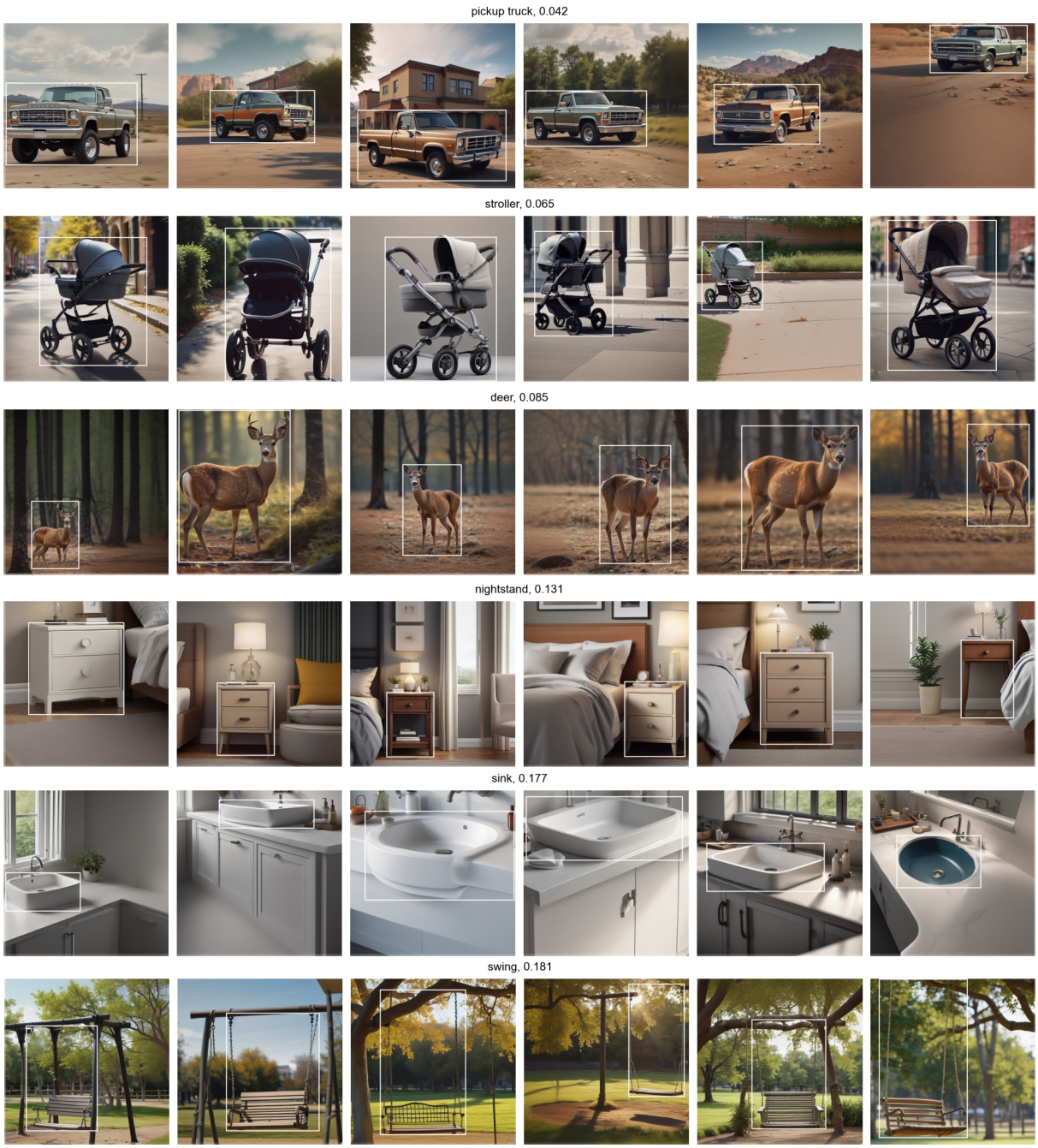

Figure 24. Bounding-box variations (part 1; continues in Fig. 25). For a given category, we show six randomly selected images from our dataset, with overlayed ground-truth bounding-box in white. Next to the category label, we report the test regression $\sqrt{\text{nMSE}}$ achieved by network CR for that category, averaged across all images in the category and across the four bounding box parameters. Categories are ordered by ascending $\sqrt{\text{nMSE}}$.

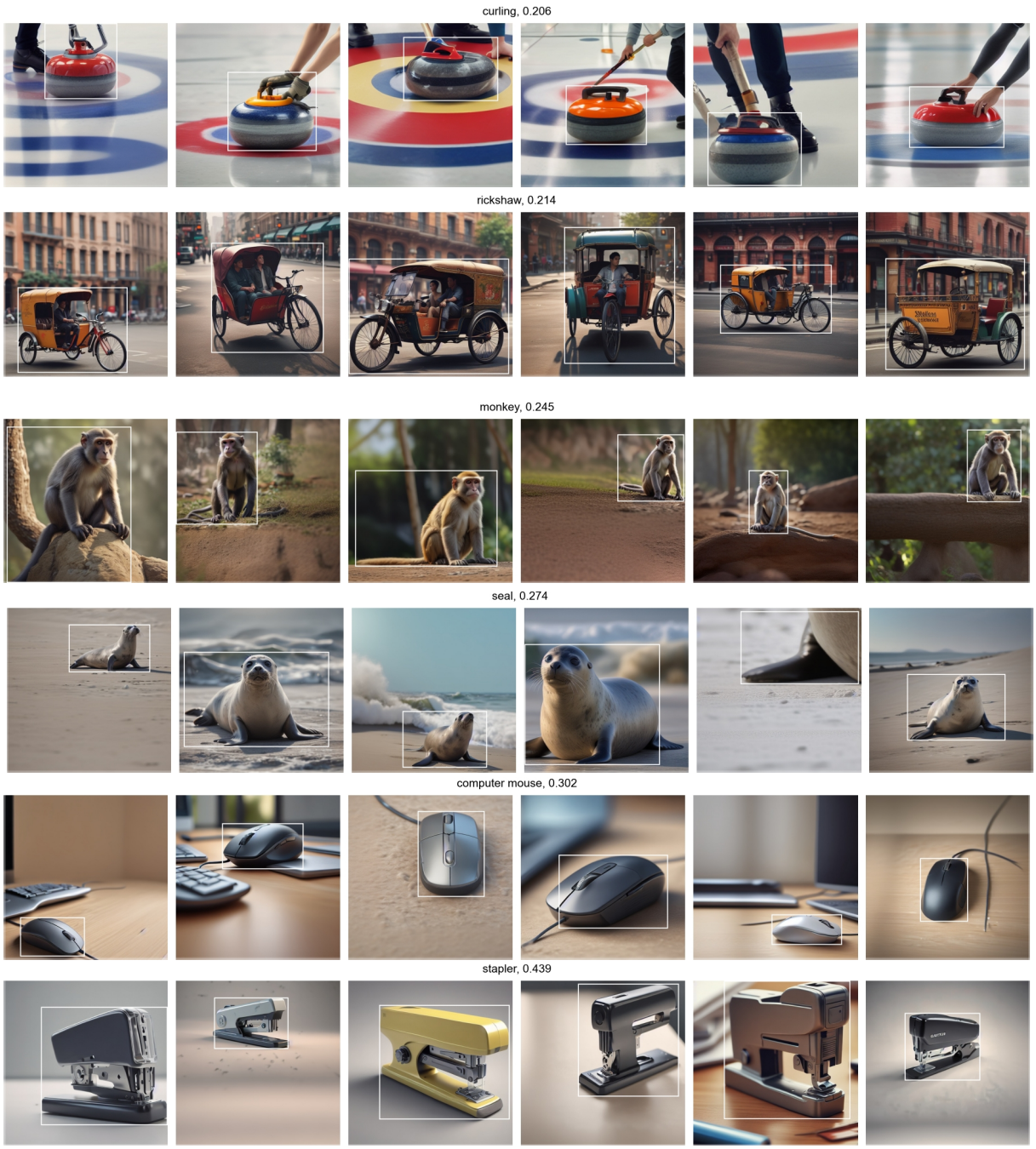

Figure 25. Bounding-box variations (part 2; see Fig. 24).

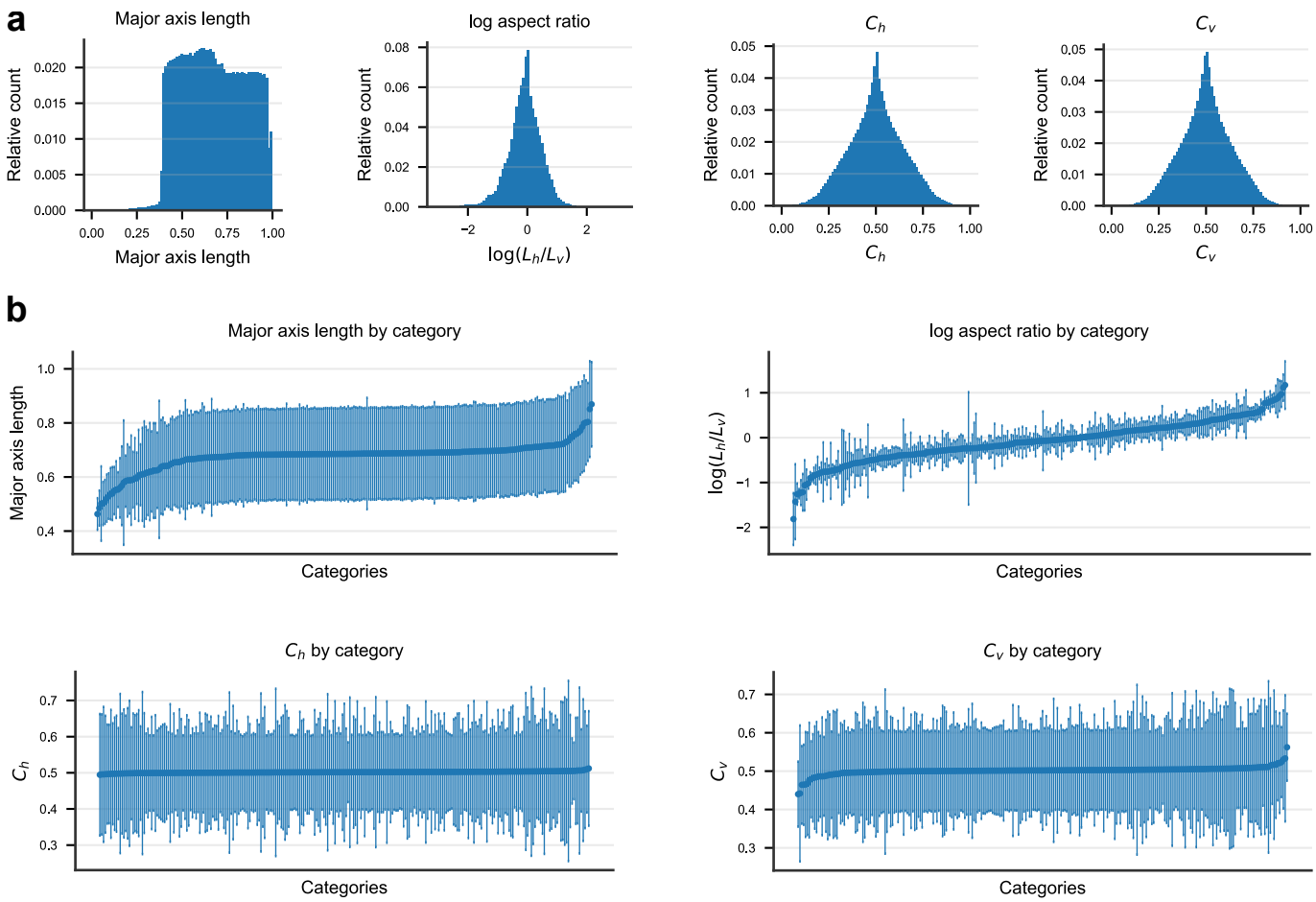

Figure 26. Bounding-box statistics in our image dataset. (**a**) Histogram across all images and all categories of major axis length $\max(L_h, L_v)$, log aspect ratio $\log(L_h/L_v)$, and center coordinates C_h and C_v. Major axis length and center coordinates are given in relative units, i.e. in pixel units divided by 512, which is the image side-length in pixel units. (**b**) Same quantities shown in (a), but on a per-category basis. Categories are ordered on the x-axis based on the within-category average of the considered quantity. Markers indicate the within-category average and errorbars the within-category standard deviation of the considered quantity.

VI. THEORY DERIVATION AND NUMERICAL VALIDATION

A. Rewriting of the local-global gap

Definitions. For convenience, we first recall the definitions of the global and local regression errors introduced in the Methods.

Without loss of generality, we consider globally centered data and labels, that is $\left\langle \langle x^\mu \rangle_\mu \right\rangle_C = 0$ and $\left\langle \langle y^\mu \rangle_\mu \right\rangle_C = 0$, where $\langle \ldots \rangle_\mu$ denotes averaging over all images within the μ-th category, and $\langle \ldots \rangle_C = \mathrm{Avg}_\mu [\ldots]$ denotes averaging over categories.

The global regression MSE is

$$E = \min_W \left\langle \left\langle \left(W^\top x^\mu - y^\mu \right)^2 \right\rangle_\mu \right\rangle_C , \tag{19}$$

while the local regression MSE for category μ is

$$E^\mu = \min_{(W^\mu, b^\mu)} \left[\left\langle \left(W^{\mu\top} x^\mu + b^\mu - y^\mu \right)^2 \right\rangle_\mu \right] \tag{20}$$

The average local error is $E_{\mathrm{loc}} = \langle E^\mu \rangle_C$, and the local-global error gap is defined as $\Delta E = E - E_{\mathrm{loc}}$. We denote by $\hat{y}^\mu = W_*^{\mu\top} x^\mu + b_*^\mu$ the prediction of the optimal local regressor for category μ, where $(W_*^\mu, b_*^\mu) = \mathrm{argmin}_{(W^\mu, b^\mu)} \left[\left\langle \left(W^{\mu\top} x^\mu + b^\mu - y^\mu \right)^2 \right\rangle_\mu \right]$.

Result. We now show that ΔE can be rewritten as a global regression error on the locally linearized labels $\hat{y}^\mu$, that is

$$\Delta E = \min_W \left[\left\langle \left\langle \left(W^\top x^\mu - \hat{y}^\mu \right)^2 \right\rangle_\mu \right\rangle_C \right] \tag{21}$$

Derivation. For each category μ, define the local residual

$$\epsilon^\mu = y^\mu - \hat{y}^\mu . \tag{22}$$

We can show that the residual satisfies

$$\langle \epsilon^\mu \rangle_\mu = 0, \qquad \langle x^\mu \epsilon^\mu \rangle_\mu = 0. \tag{23}$$

Indeed, the optimal local regressor is given by

$$W_*^\mu = K^{\mu +} d^\mu, \qquad b_*^\mu = \langle y^\mu \rangle_\mu - W_*^{\mu\top} \langle x^\mu \rangle_\mu , \tag{24}$$

where $K^\mu = \left\langle \delta x^\mu \delta x^{\mu\top} \right\rangle_\mu$, $d^\mu = \langle \delta x^\mu \delta y^\mu \rangle_\mu$, $\delta x^\mu = x^\mu - \langle x^\mu \rangle_\mu$, $\delta y^\mu = y^\mu - \langle y^\mu \rangle_\mu$, and $K^{\mu +}$ denotes the Moore-Penrose pseudoinverse. Therefore,

$$\epsilon^\mu = y^\mu - \hat{y}^\mu = \delta y^\mu - W_*^{\mu\top} \delta x^\mu . \tag{25}$$

Averaging immediately gives $\langle \epsilon^\mu \rangle_\mu = 0$. Moreover,

$$\langle x^\mu \epsilon^\mu \rangle_\mu = \langle \delta x^\mu \epsilon^\mu \rangle_\mu = d^\mu - K^\mu W_*^\mu = d^\mu - K^\mu K^{\mu +} d^\mu = 0, \tag{26}$$

which completes the derivation of Eq. (23).

Since $\hat{y}^\mu$ is itself a linear function of x^μ, from Eq. (23) it follows that

$$\langle \hat{y}^\mu \epsilon^\mu \rangle_\mu = 0. \tag{27}$$

We can then write, for any global regressor W,

$$\begin{aligned} E\left(W\right) &= \left\langle\left\langle\left(W^{\top}x^{\mu}-y^{\mu}\right)^{2}\right\rangle_{\mu}\right\rangle_{C} = \left\langle\left\langle\left(W^{\top}x^{\mu}-\hat{y}^{\mu}-\epsilon^{\mu}\right)^{2}\right\rangle_{\mu}\right\rangle_{C} \\ &= \left\langle\left\langle\left(W^{\top}x^{\mu}-\hat{y}^{\mu}\right)^{2}\right\rangle_{\mu}\right\rangle_{C} - 2\left\langle\left\langle\left(W^{\top}x^{\mu}-\hat{y}^{\mu}\right)\epsilon^{\mu}\right\rangle_{\mu}\right\rangle_{C} + \left\langle\left\langle\epsilon^{\mu 2}\right\rangle_{\mu}\right\rangle_{C} \end{aligned} \tag{28}$$

where in the last step we expanded the square. The cross term vanishes, since

$$\left\langle\left(W^{\top}x^{\mu}-\hat{y}^{\mu}\right)\epsilon^{\mu}\right\rangle_{\mu} = W^{\top}\left\langle x^{\mu}\epsilon^{\mu}\right\rangle_{\mu} - \left\langle\hat{y}^{\mu}\epsilon^{\mu}\right\rangle_{\mu} = 0. \tag{29}$$

By construction, $\left\langle\epsilon^{\mu 2}\right\rangle_{\mu} = E^{\mu}$, and therefore $\left\langle\left\langle\epsilon^{\mu 2}\right\rangle_{\mu}\right\rangle_{C} = E_{\mathrm{loc}}$. Minimizing $E\left(W\right)$ with respect to W, we thus obtain

$$E = \min_{W}\left\langle\left\langle\left(W^{\top}x^{\mu}-\hat{y}^{\mu}\right)^{2}\right\rangle_{\mu}\right\rangle_{C} + E_{\mathrm{loc}}, \tag{30}$$

which immediately implies Eq. (21).

B. Derivation of the local-global gap in terms of manifold geometry

Generative model. For convenience, we first recall the definitions and assumptions of the manifold generative model introduced in the Methods. As explained in the Methods, we derive the local-global gap in the locally-centered case $\left\langle x^{\mu}\right\rangle_{\mu} = 0, \left\langle y^{\mu}\right\rangle_{\mu} = 0, \forall\mu$, and then quantify the effect of manifolds centroids by the centroid error. Without loss of generality, each point in manifold μ is parameterized as

$$x^{\mu} = \sum_{d=1}^{N} s_{d}^{\mu}u_{d}^{\mu}, \tag{31}$$

where $\{u_{d}^{\mu}\}_{d=1}^{N}$ is an orthonormal basis of $\mathbb{R}^{N}$, and s_{d}^{μ} is the coordinate along direction d. We choose

$$u_{1}^{\mu} = \frac{W_{*}^{\mu}}{\|W_{*}^{\mu}\|} \tag{32}$$

to be the feature-encoding direction, while the remaining directions $u_{d>1}^{\mu}$ are chosen to diagonalize the covariance in the subspace orthogonal to u_{1}^{μ}. With this choice, the covariance of the manifold coordinates takes the form

$$\left\langle s^{\mu}s^{\mu\top}\right\rangle_{\mu} = \Sigma^{\mu} = \begin{pmatrix} \sigma_{1}^{\mu 2} & \Sigma_{12}^{\mu} & \cdots & \Sigma_{1N}^{\mu} \\ \Sigma_{21}^{\mu} & \sigma_{2}^{\mu 2} & 0 & 0 \\ \vdots & 0 & \ddots & 0 \\ \Sigma_{N1}^{\mu} & 0 & 0 & \sigma_{N}^{\mu 2} \end{pmatrix}. \tag{33}$$

Note that local centering of the manifolds implies $\langle s_{d}^{\mu}\rangle = 0$. Let us also define

$$r^{\mu} = \|W_{*}^{\mu}\| \tag{34}$$

as the scale of the feature encoding, and note that

$$\hat{y}^{\mu} = r^{\mu}s_{1}^{\mu} \tag{35}$$

We assume the following across-manifold statistics:

1. The feature-encoding directions u_1^μ are isotropically distributed around a common mean direction

$$u_1^\mu = \sqrt{a} u_1 + \sqrt{1-a}\eta^\mu \,, \tag{36}$$

where $a \in [0,1]$ is the alignment strength, $u_1 \in \mathbb{R}^N$ is a unit-norm vector representing the common mean direction, and $\eta^\mu \in \mathbb{R}^N$ is a random i.i.d. Gaussian vector with $\langle \eta^\mu \rangle_C = 0$ and $\left\langle \eta^\mu \eta^{\mu\top} \right\rangle_C = \frac{1}{N}\mathbb{I}$. Under this definition, u_1^μ has unit norm on average, i.e. $\left\langle u_1^{\mu\top} u_1^\mu \right\rangle_C = 1$, with fluctuations that vanish as powers of $1/N$, which is sufficient for our derivations in the limit $N \to \infty$. For the coming derivations, it is useful to explicitly write the statistics of u_1^μ up to second order

$$\langle u_1^\mu \rangle_C = \sqrt{a} u_1 \tag{37}$$

$$\left\langle u_1^\mu u_1^{\mu\top} \right\rangle_C = a u_1 u_1^\top + (1-a)\frac{1}{N}\mathbb{I} \tag{38}$$

2. All manifold directions with $d > 1$ are mutually uncorrelated and isotropically distributed in neuronal space, conditioned on being orthogonal to u_1^μ (as by their definition as the elements of an orthonormal basis). Concretely,

$$\langle u_d^\mu \rangle_C = 0 \tag{39}$$

$$\left\langle u_d^\mu u_{d'}^{\mu\top} \right\rangle_C = \frac{1}{N}\delta_{d,d'}\left(\mathbb{I} - \left\langle u_1^\mu u_1^{\mu\top} \right\rangle_C\right) = \frac{1}{N}\delta_{d,d'}\left(\mathbb{I} - a\, u_1 u_1^\top\right) + \mathcal{O}\left(\frac{1}{N^2}\right). \tag{40}$$

3. Aside from the above correlations, all manifold parameters are otherwise uncorrelated.

Asymptotic regime. Our main theory is derived in the asymptotic regime $P \to \infty$ with N large but finite. Assuming N is large, our theory neglects correction of $\mathcal{O}\left(N^{-1}\right)$. On this regard, we assume the SNR$=\sum_d \left\langle \sigma_d^{\mu 2} \right\rangle_C / \left\langle \sigma_d^{\mu 2} \right\rangle_C$ to be extensive in N, SNR $= \mathcal{O}(N)$. This is important to keep the orientation error E_o of $\mathcal{O}(1)$. This choice is motivated by the empirical observation that the orientation and scale errors are of similar magnitude, and therefore the orientation error should be considered as an $\mathcal{O}(1)$ term, rather then neglected as an $\mathcal{O}\left(N^{-1}\right)$ term. We emphasize that this assumption is used as a controlled scaling prescription for our theoretical derivation. It should not be interpreted as a literal claim about the behavior of variance in arbitrarily large biological networks. Rather, it provides a convenient large-N continuation of the empirical finite-N regime in which we find the orientation error to remain finite.

Result. For the generative model and asymptotic regime described above, and neglecting $\mathcal{O}\left(N^{-1}\right)$ corrections, we find

$$\Delta E = \hat{\sigma}^2 \left[1 - \frac{1}{(1+E_\mathrm{s})(1+E_\mathrm{o})}\right] \tag{41}$$

$$E_\mathrm{s} = \frac{\mathrm{Var}_\mu\left[r^\mu\right]}{\mathrm{Avg}_\mu\left[r^\mu\right]^2} \tag{42}$$

$$E_\mathrm{o} = \frac{1-a}{N \cdot a \cdot \mathrm{SNR}} \tag{43}$$

where $\mathrm{SNR} = \frac{\left\langle \sigma_1^{\mu 2} \right\rangle_C}{\sum_{d=1}^N \left\langle \sigma_d^{\mu 2} \right\rangle_C}$, and $\hat{\sigma}^2 = \left\langle \left\langle \delta\hat{y}^{\mu 2} \right\rangle_\mu \right\rangle_C$ with $\delta\hat{y}^\mu = \hat{y}^\mu - \langle \hat{y}^\mu \rangle_\mu$.

Derivation. Recall that

$$\Delta E = \min_W \left[\left\langle \left\langle \left(W^\top x^\mu - \hat{y}^\mu\right)^2 \right\rangle_\mu \right\rangle_C\right] \tag{44}$$

minimizing w.r.t. W we obtain

$$\Delta E = \hat{\sigma}^2 - d^\top K^+ d \tag{45}$$

where

$$K = \left\langle \left\langle x^\mu x^{\mu\top} \right\rangle_\mu \right\rangle_C \tag{46}$$

$$d = \left\langle \left\langle x^\mu \hat{y}^\mu \right\rangle_\mu \right\rangle_C \tag{47}$$

$$\hat{\sigma}^2 = \left\langle \left\langle \hat{y}^{\mu 2} \right\rangle_\mu \right\rangle_C \tag{48}$$

Under our assumptions of manifold statistics, we have, neglecting $\mathcal{O}\left(N^{-1}\right)$ corrections,

$$K = \left\langle \sigma_1^{\mu 2} \right\rangle_C \left(a u_1 u_1^\top + \frac{1-a}{N} \mathbb{I} \right) + \frac{1}{N} \sum_{d>1} \left\langle \sigma_d^{\mu 2} \right\rangle_C \left(\mathbb{I} - a u_1 u_1^\top \right) \tag{49}$$

$$d = \left\langle r^\mu \right\rangle_C \left\langle \sigma_1^{\mu 2} \right\rangle_C \sqrt{a} u_1 \tag{50}$$

$$\hat{\sigma}^2 = \left\langle r^{\mu 2} \right\rangle_C \left\langle \sigma_1^{\mu 2} \right\rangle_C \tag{51}$$

We note that d is an eigenvector of K, therefore we can readily derive

$$d^\top K^+ d = \frac{a \left\langle r^\mu \right\rangle_C^2 \left\langle \sigma_1^{\mu 2} \right\rangle_C^2}{a \left\langle \sigma_1^{\mu 2} \right\rangle_C + \frac{1-a}{N} \sum_d \left\langle \sigma_d^{\mu 2} \right\rangle_C} \tag{52}$$

Normalizing by $\hat{\sigma}^2 = \left\langle r^{\mu 2} \right\rangle_C \left\langle \sigma_1^{\mu 2} \right\rangle_C$ we obtain

$$\frac{d^\top K^+ d}{\hat{\sigma}^2} = \frac{\left\langle r^\mu \right\rangle_C^2}{\left\langle r^{\mu 2} \right\rangle_C} \frac{1}{1 + \frac{1}{N} \frac{1-a}{a} \frac{\sum_d \left\langle \sigma_d^{\mu 2} \right\rangle_C}{\left\langle \sigma_1^{\mu 2} \right\rangle_C}} \tag{53}$$

We immediately identify the second term to be $\frac{1}{1+E_\mathrm{o}}$. Defining $\delta r^\mu = r^\mu - \left\langle r^\mu \right\rangle_C$, we can rewrite the first term as

$$\frac{\left\langle r^\mu \right\rangle_C^2}{\left\langle r^\mu \right\rangle_C^2 + \left\langle \delta r^{\mu 2} \right\rangle_C} = \frac{1}{1 + \frac{\left\langle \delta r^{\mu 2} \right\rangle_C}{\left\langle r^\mu \right\rangle_C^2}} = \frac{1}{1 + E_\mathrm{s}} \tag{54}$$

Therefore

$$\frac{d^\top K^+ d}{\hat{\sigma}^2} = \frac{1}{\left(1 + E_\mathrm{s}\right)\left(1 + E_\mathrm{o}\right)} \tag{55}$$

which plugged into Eq. (45) gives exactly Eq. (41), as desired.

C. Derivation of the local-global gap in the regime of finite number of categories

Here we derive the local-global gap ΔE in the regime of finite category density, by studying the thermodynamic limit $P, N \to \infty$ at fixed $\alpha = \frac{P}{N} \in (0, \infty)$. Our previous result for infinitely many categories is recovered as the limiting case $\alpha \to \infty$. We recall that we assume that $\mathrm{SNR} = \mathcal{O}(N)$ or, equlvalently for $\left\langle \sigma_1^{\mu 2} \right\rangle = \mathcal{O}(1)$, $\sum_d \left\langle \sigma_d^{\mu 2} \right\rangle_C = \mathcal{O}(N)$.

1. Gibbs distribution

To derive the finite-P corrections to the local-global gap, we use the replica method. We start from the training objective

$$\Delta E(W) = \left\langle \left\langle \left(W^\top x^\mu - \hat{y}^\mu\right)^2 \right\rangle_\mu \right\rangle_C , \tag{56}$$

whose minimum is ΔE. We introduce the Gibbs distribution

$$p(W) = \frac{1}{Z} \exp\left(-\beta \tilde{\Delta E}(W)\right), \tag{57}$$

with $\tilde{\Delta E}(W) = \frac{P}{2}\Delta E(W)$ and inverse temperature $\beta \in [0, \infty)$. The corresponding partition function is

$$Z = \int dW \, \exp\left(-\beta \tilde{\Delta E}(W)\right). \tag{58}$$

The average training error under this measure is

$$\langle \Delta E(W) \rangle_{p(W)} = -\frac{2}{P} \frac{\partial \ln Z}{\partial \beta}. \tag{59}$$

In the zero-temperature limit $\beta \to \infty$, the measure $p(W)$ concentrates on the minimizer W_* of $\Delta E(W)$ and therefore

$$\lim_{\beta \to \infty} \langle \Delta E(W) \rangle_{p(W)} = \Delta E(W_*) = \Delta E. \tag{60}$$

The quenched disorder here is given by the random manifold parameters entering x^μ, namely the manifold directions. We assume that the training error is self-averaging, that is it coincides with its disorder average $\Delta E = \langle \Delta E \rangle_C$. Thus, ΔE can be obtained from the disorder-averaged partition function $\langle \ln Z \rangle_C$ using the replica method.

2. Auxiliary fields

We write explicitly the shift term appearing in $\Delta E(W)$ (Eq. (56)) as

$$W^\top x^\mu - \hat{y}^\mu = s_1^\mu \left(\sqrt{a}\, W^\top u_1 + \sqrt{1-a}\, W^\top \eta^\mu - r^\mu\right) + \sum_{d=2}^{N} s_d^\mu W^\top u_d^\mu. \tag{61}$$

This suggests introducing the order parameter

$$m = W^\top u_1, \tag{62}$$

and auxiliary fields

$$v_1^\mu = W^\top \eta^\mu, \qquad v_d^\mu = W^\top u_d^\mu, \quad d = 2, \ldots, N. \tag{63}$$

In terms of these fields, the energy becomes

$$\tilde{\Delta E}(v, m) = \frac{1}{2} \sum_{\mu=1}^{P} \left\langle \left(s_1^\mu \left(\sqrt{a}\, m + \sqrt{1-a}\, v_1^\mu - r^\mu\right) + \sum_{d=2}^{N} s_d^\mu v_d^\mu \right)^2 \right\rangle_\mu . \tag{64}$$

Carrying out the average over the manifold coordinates s^μ, we obtain

$$\tilde{\Delta E}(v,m) = \frac{1}{2}\sum_{\mu=1}^{P}\Bigg[a\,\sigma_1^{\mu 2}m^2 + 2\sqrt{a(1-a)}\,\sigma_1^{\mu 2}mv_1^\mu + 2\sqrt{a}\sum_{d=2}^{N}\Sigma_{1d}^\mu m v_d^\mu - 2\sqrt{a}\,\sigma_1^{\mu 2}r^\mu m$$

$$+(1-a)\sigma_1^{\mu 2}(v_1^\mu)^2 + 2\sqrt{1-a}\sum_{d=2}^{N}\Sigma_{1d}^\mu v_1^\mu v_d^\mu - 2\sqrt{1-a}\,\sigma_1^{\mu 2}r^\mu v_1^\mu + \sum_{d=2}^{N}\sigma_d^{\mu 2}(v_d^\mu)^2$$

$$-2r^\mu\sum_{d=2}^{N}\Sigma_{1d}^\mu v_d^\mu + \sigma_1^{\mu 2}(r^\mu)^2\Bigg]. \tag{65}$$

We next enforce the definitions of the auxiliary fields through Dirac delta representations. Introducing the conjugate fields $\hat{m}$, $\hat{v}_1^\mu$, and $\hat{v}_d^\mu$, the partition function can be written as

$$Z \propto \int DW\,dm\,d\hat{m}\,Dv\,D\hat{v}\,\exp\Big(A(W,m,\hat{m},v,\hat{v}) - \beta\tilde{\Delta E}(v,m)\Big), \tag{66}$$

where

$$A(W,m,\hat{m},v,\hat{v}) = -i\hat{m}\,m + i\hat{m}\,W^\top u_1 - i\sum_{\mu=1}^{P}\hat{v}_1^\mu v_1^\mu + i\sum_{\mu=1}^{P}\hat{v}_1^\mu W^\top\eta^\mu - i\sum_{\mu=1}^{P}\sum_{d=2}^{N}\hat{v}_d^\mu v_d^\mu + i\sum_{\mu=1}^{P}\sum_{d=2}^{N}\hat{v}_d^\mu W^\top u_d^\mu. \tag{67}$$

Here and throughout, for any integration variable z carrying one or more discrete indices, we denote by

$$Dz \equiv \prod_{\{\text{all indices of } z\}} dz \tag{68}$$

the corresponding integration measure. For example,

$$DW = \prod_{k=1}^{N} dW_k, \qquad Dv = \prod_{\mu=1}^{P}\prod_{d=1}^{N} dv_d^\mu, \qquad D\hat{v} = \prod_{\mu=1}^{P}\prod_{d=1}^{N} d\hat{v}_d^\mu. \tag{69}$$

3. *Replicas*

To compute the disorder-averaged free energy, we use the replica identity

$$\langle \ln Z\rangle_C = \lim_{n\to 0}\frac{\langle Z^n\rangle_C - 1}{n}. \tag{70}$$

Replicating the partition function, we obtain

$$\langle Z^n\rangle_C \propto \int DW\,Dm\,D\hat{m}\,Dv\,D\hat{v}\,\left\langle \exp\left(A(\{W^\alpha,m^\alpha,\hat{m}^\alpha,v^\alpha,\hat{v}^\alpha\}) - \beta\sum_{\alpha=1}^{n}\tilde{\Delta E}(v^\alpha,m^\alpha)\right)\right\rangle_C, \tag{71}$$

with

$$A = -i\sum_{\alpha=1}^{n}\hat{m}^{\alpha}m^{\alpha} + i\sum_{\alpha=1}^{n}\hat{m}^{\alpha}W^{\alpha\top}u_1 - i\sum_{\alpha=1}^{n}\sum_{\mu=1}^{P}\hat{v}_1^{\mu\alpha}v_1^{\mu\alpha} + i\sum_{\alpha=1}^{n}\sum_{\mu=1}^{P}\hat{v}_1^{\mu\alpha}W^{\alpha\top}\eta^{\mu}$$

$$-i\sum_{\alpha=1}^{n}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\hat{v}_d^{\mu\alpha}v_d^{\mu\alpha} + i\sum_{\alpha=1}^{n}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\hat{v}_d^{\mu\alpha}W^{\alpha\top}u_d^{\mu}, \tag{72}$$

and

$$\tilde{\Delta E}(v^{\alpha}, m^{\alpha}) = \frac{1}{2}\sum_{\mu=1}^{P}\Bigg[a\,\sigma_1^{\mu 2}(m^{\alpha})^2 + 2\sqrt{a(1-a)}\,\sigma_1^{\mu 2}m^{\alpha}v_1^{\mu\alpha} + 2\sqrt{a}\sum_{d=2}^{N}\Sigma_{1d}^{\mu}m^{\alpha}v_d^{\mu\alpha} - 2\sqrt{a}\,\sigma_1^{\mu 2}r^{\mu}m^{\alpha}$$

$$+(1-a)\sigma_1^{\mu 2}(v_1^{\mu\alpha})^2 + 2\sqrt{1-a}\sum_{d=2}^{N}\Sigma_{1d}^{\mu}v_1^{\mu\alpha}v_d^{\mu\alpha} - 2\sqrt{1-a}\,\sigma_1^{\mu 2}r^{\mu}v_1^{\mu\alpha} + \sum_{d=2}^{N}\sigma_d^{\mu 2}(v_d^{\mu\alpha})^2$$

$$-2r^{\mu}\sum_{d=2}^{N}\Sigma_{1d}^{\mu}v_d^{\mu\alpha} + \sigma_1^{\mu 2}(r^{\mu})^2\Bigg]. \tag{73}$$

4. Average over the quenched disorder

We now average over the quenched disorder, i.e. over the random manifold directions. Concretely, we average over the random vectors η^{μ} and u_d^{μ} for $d > 1$. This boils down to the calculation of standard Gaussian integrals, at the end of which we obtain

$$\langle Z^n\rangle_C \propto \int DQ\,D\hat{Q}\,DW\,Dm\,D\hat{m}\,Dv\,D\hat{v}\,\exp\left(A(Q,W,m,\hat{m},v,\hat{v}) + B(Q,\hat{Q},W) - \beta\sum_{\alpha=1}^{n}\tilde{\Delta E}(v^{\alpha},m^{\alpha})\right), \tag{74}$$

where

$$B(Q,\hat{Q},W) = \frac{N}{2}\mathrm{Tr}\,Q^{\top}\hat{Q} - \frac{1}{2}\sum_{\alpha,\beta=1}^{n}\hat{Q}^{\alpha\beta}W^{\alpha\top}W^{\beta}, \tag{75}$$

$$A = -i\sum_{\alpha=1}^{n}\hat{m}^{\alpha}m^{\alpha} + i\sum_{\alpha=1}^{n}\hat{m}^{\alpha}W^{\alpha\top}u_1 - i\sum_{\alpha=1}^{n}\sum_{\mu=1}^{P}\hat{v}_1^{\mu\alpha}v_1^{\mu\alpha} - i\sum_{\alpha=1}^{n}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\hat{v}_d^{\mu\alpha}v_d^{\mu\alpha}$$

$$-\frac{1-a}{2}\sum_{\mu=1}^{P}\sum_{\alpha,\beta=1}^{n}\hat{v}_1^{\mu\alpha}Q^{\alpha\beta}\hat{v}_1^{\mu\beta} - \frac{1}{2}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\sum_{\alpha,\beta=1}^{n}\hat{v}_d^{\mu\alpha}\left(Q^{\alpha\beta} - \frac{a}{N}m^{\alpha}m^{\beta}\right)\hat{v}_d^{\mu\beta}, \tag{76}$$

and, by means of Dirac delta representations, we introduced the overlap order parameters

$$Q^{\alpha\beta} = \frac{1}{N}W^{\alpha\top}W^{\beta}, \tag{77}$$

and their conjugates $\hat{Q}^{\alpha\beta}$.

5. Gaussian integrals in the auxiliary fields

We now perform, in this order, the Gaussian integrals over $\hat{v}$, W, and $\hat{m}$. Neglecting additive constants independent of β, and dropping the subleading $\mathcal{O}(1)$ log-determinant contribution generated by the $\hat{m}$ integral, we obtain

$$\langle Z^n \rangle_C \propto \int DQ\, D\hat{Q}\, Dm\, Dv\, \exp\left(A(Q,\hat{Q},m,v) + B(Q,\hat{Q},m) - \beta \sum_{\alpha=1}^{n} \tilde{\Delta E}(v^\alpha, m^\alpha) \right), \tag{78}$$

with

$$A = -\frac{1}{2}\sum_{\alpha,\beta=1}^{n} m^\alpha \hat{Q}^{\alpha\beta} m^\beta - \frac{1}{2}(1-a)^{-1} \sum_{\mu=1}^{P}\sum_{\alpha,\beta=1}^{n} v_1^{\mu\alpha} \left(Q^{-1}\right)^{\alpha\beta} v_1^{\mu\beta}$$

$$-\frac{1}{2}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\sum_{\alpha,\beta=1}^{n} v_d^{\mu\alpha} \left[Q^{-1} + \frac{a}{N}\frac{Q^{-1}mm^\top Q^{-1}}{1-\frac{a}{N}m^\top Q^{-1} m} \right]^{\alpha\beta} v_d^{\mu\beta}, \tag{79}$$

and

$$B = \frac{N}{2}\mathrm{Tr}\, Q^\top \hat{Q} - \frac{1}{2}PN \ln\det(Q) - \frac{1}{2}N)P \ln\left(1 - \frac{a}{N} m^\top Q^{-1} m\right) - \frac{1}{2}N \ln\det(\hat{Q}), \tag{80}$$

where we used the identities

$$\left(Q - \frac{a}{N}mm^\top\right)^{-1} = Q^{-1} + \frac{a}{N}\frac{Q^{-1}mm^\top Q^{-1}}{1-\frac{a}{N}m^\top Q^{-1} m}, \tag{81}$$

and

$$\det\left(Q - \frac{a}{N}mm^\top\right) = \left(1 - \frac{a}{N} m^\top Q^{-1} m\right)\det(Q). \tag{82}$$

6. Replica symmetric ansatz

Assuming a replica-symmetric saddle point, we make the ansatz

$$Q = (q_1 - q_0)\,\mathbb{I} + q_0 \mathbf{1}, \qquad \hat{Q} = (\hat{q}_1 - \hat{q}_0)\,\mathbb{I} + \hat{q}_0 \mathbf{1}, \qquad m^\alpha = m, \tag{83}$$

Using standard identities for replica-symmetric matrices, we then obtain

$$A(Q,\hat{Q},m,v) = -\frac{1}{2}n\,(\hat{q}_1 - \hat{q}_0)\,m^2 - \frac{1}{2}\frac{(1-a)^{-1}}{q_1 - q_0}\sum_{\mu=1}^{P}\sum_{\alpha=1}^{n}(v_1^{\mu\alpha})^2 + \frac{1}{2}\frac{(1-a)^{-1}q_0}{(q_1-q_0)^2}\sum_{\mu=1}^{P}\left(\sum_{\alpha=1}^{n} v_1^{\mu\alpha}\right)^2$$

$$-\frac{1}{2}\frac{1}{q_1-q_0}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\sum_{\alpha=1}^{n}(v_d^{\mu\alpha})^2 + \frac{1}{2}\frac{q_0}{(q_1-q_0)^2}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\left(\sum_{\alpha=1}^{n} v_d^{\mu\alpha}\right)^2 - \frac{1}{2}\frac{a}{N}\frac{m^2}{(q_1-q_0)^2}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\left(\sum_{\alpha=1}^{n} v_d^{\mu\alpha}\right)^2, \tag{84}$$

and

$$B(Q,\hat{Q},m) = \frac{Nn}{2}\left[(q_1-q_0)(\hat{q}_1-\hat{q}_0)+(q_1-q_0)\hat{q}_0+(\hat{q}_1-\hat{q}_0)q_0\right]$$

$$-\frac{1}{2}PN\,n\left[\ln(q_1-q_0)+\frac{q_0}{q_1-q_0}\right]-\frac{1}{2}N\,n\left[\ln(\hat{q}_1-\hat{q}_0)+\frac{\hat{q}_0}{\hat{q}_1-\hat{q}_0}\right]+\frac{1}{2}NP\frac{a}{N}\,n\,\frac{m^2}{q_1-q_0}. \tag{85}$$

where we retained only terms up to linear order in n.

7. *Remaining Gaussian integrals*

Note that Eq. (84) contains replica-coupling terms of the form $(\sum_\alpha v_1^{\mu\alpha})^2$ and $(\sum_\alpha v_d^{\mu\alpha})^2$. We decouple these terms by standard Hubbard-Stratonovich transformations, introducing Gaussian auxiliary fields t_1^μ and t_d^μ for $d=2,\dots,N$. At fixed t, the integrals over the replicated variables factorize over α. Since the replica limit $n\to 0$ is taken at the end, we retain only the linear term in n, using

$$\int Dt^\mu\left[Z_\mu(t^\mu)\right]^n = 1+n\int Dt^\mu \ln Z_\mu(t^\mu)+\mathcal{O}(n^2), \tag{86}$$

where Dt^μ denotes the standard Gaussian measure over the auxiliary fields associated with manifold μ. We then perform sequentially the Gaussian integrals over v_d^μ for $d=2,\dots,N$, over v_1^μ, and finally over the auxiliary fields t^μ. This yields

$$\langle Z^n\rangle_C \propto \int dm\,dq_0\,d\delta q\,d\hat{q}_0\,d\delta\hat{q}\,\exp\left(\frac{Pn}{2}S_{\rm RS}(m,q_0,\delta q,\hat{q}_0,\delta\hat{q})\right), \tag{87}$$

where

$$\frac{2}{Pn}S_{\rm RS}(m,q_0,\delta q,\hat{q}_0,\delta\hat{q}) = -\beta\,\frac{1}{P}\sum_{\mu=1}^{P}\sigma_1^{\mu 2}r^{\mu 2}-\beta a\left\langle\sigma_1^{\mu 2}\right\rangle_C m^2+2\beta\sqrt{a}\,\frac{1}{P}\sum_{\mu=1}^{P}r^\mu\sigma_1^{\mu 2}\,m-\frac{\beta}{P}\delta\hat{q}\,m^2$$

$$+\alpha^{-1}\left[\delta q\,\delta\hat{q}+\beta^{-1}\delta q\,\hat{q}_0+\beta\,\delta\hat{q}\,q_0\right]-\alpha^{-1}\left[\ln(\beta\delta\hat{q})+\frac{\hat{q}_0}{\beta\delta\hat{q}}\right]$$

$$-N\left[\ln(\beta^{-1}\delta q)+\frac{\beta q_0}{\delta q}\right]+\frac{\beta a\,m^2}{\delta q}-\frac{1}{P}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\ln\beta U_d^\mu+\frac{1}{P}\beta\sum_{\mu=1}^{P}\sum_{d=2}^{N}S_d^\mu\,(\delta_m^\mu)^2+\frac{1}{P}\beta\sum_{\mu=1}^{P}\sum_{d=2}^{N}(U_d^\mu)^{-1}V_2$$

$$-\frac{1}{P}\sum_{\mu=1}^{P}\ln\left(\beta U_1^\mu-\beta\sum_{d=2}^{N}S_d^\mu\right)+\frac{1}{P}\beta\sum_{\mu=1}^{P}\left(U_1^\mu-\sum_{d=2}^{N}S_d^\mu\right)^{-1}\left(\sigma_1^{\mu 2}-\sum_{d=2}^{N}S_d^\mu\right)^2(\delta_m^\mu)^2$$

$$+\frac{1}{P}\beta\sum_{\mu=1}^{P}\left(U_1^\mu-\sum_{d=2}^{N}S_d^\mu\right)^{-1}V_1+\frac{1}{P}\beta\sum_{\mu=1}^{P}\left(U_1^\mu-\sum_{d=2}^{N}S_d^\mu\right)^{-1}V_2\sum_{d=2}^{N}\left((U_d^\mu)^{-1}\Sigma_{1d}^\mu\right)^2, \tag{88}$$

with

$$U_1^\mu=\sigma_1^{\mu 2}+\frac{(1-a)^{-1}}{\delta q},\qquad U_d^\mu=\sigma_d^{\mu 2}+\frac{1}{\delta q},\qquad d=2,\dots,N, \tag{89}$$

$$V_1 = \frac{(1-a)^{-1} q_0}{\delta q^2}, \qquad V_2 = \frac{q_0}{\delta q^2} - \frac{a}{N}\frac{m^2}{\delta q^2}, \tag{90}$$

$$S_d^\mu = (U_d^\mu)^{-1} (\Sigma_{1d}^\mu)^2, \qquad \delta_m^\mu = r^\mu - \sqrt{a}\, m. \tag{91}$$

Note that we performed the change of variables

$$\beta^{-1} \delta q = q_1 - q_0 \tag{92}$$

$$\beta \delta \hat{q} = \hat{q}_1 - \hat{q}_0 \tag{93}$$

while keeping q_0 and $\hat{q}_0$ unchanged.

8. *Saddle point*

In the limit $N, P \to \infty$, with $\frac{P}{N} = \alpha \in [0, \infty]$, the integral in (87) can be solved via saddle point. We note that S_{RS} is linear in q_0 and $\hat{q}_0$, so we do not need to compute the saddle point solution for q_0 and $\hat{q}_0$ explicitly, as any term of S_{RS} that is linear in these variables vanishes under the saddle point condition. The saddle point equation w.r.t. $\hat{q}_0$ gives

$$\delta q = \delta \hat{q}^{-1} \tag{94}$$

The saddle point equation w.r.t. q_0 gives the self-consistent equation for δq

$$\boxed{\delta \hat{q} \frac{1}{N} \frac{1}{P} \sum_\mu \left\{ \sum_{d=2}^N (U_d^\mu)^{-1} + (1-a)^{-1} \left[U_1^\mu - \sum_{d=2}^N S_d^\mu \right]^{-1} + \left[U_1^\mu - \sum_{d=2}^N S_d^\mu \right]^{-1} \sum_{d=2}^N \left((U_d^\mu)^{-1} \Sigma_{1d}^\mu \right)^2 \right\} = 1 - \frac{1}{N} \alpha^{-1}} \tag{95}$$

9. *Order parameter in the $\alpha \to \infty$ limit*

It is useful to study the order parameter in the limit $\alpha \to \infty$. We rewrite the self-consistent equation Eq. (95)

$$\frac{1}{N} \frac{1}{P} \sum_\mu \left\{ \sum_{d=2}^N \frac{\delta \hat{q}}{\sigma_d^{\mu 2} + \delta \hat{q}} + \frac{(1-a)^{-1} \delta \hat{q}}{\sigma_1^{\mu 2} + (1-a)^{-1} \delta \hat{q} - \sum_{d>1} \frac{\left(\Sigma_{1d}^\mu\right)^2}{\sigma_d^{\mu 2} + \delta \hat{q}}} \right.$$
$$\left. + \frac{\delta \hat{q}}{\sigma_1^{\mu 2} + (1-a)^{-1} \delta \hat{q} - \sum_{d>1} \frac{\left(\Sigma_{1d}^\mu\right)^2}{\sigma_d^{\mu 2} + \delta \hat{q}}} \sum_{d=2}^D \left(\frac{\Sigma_{1d}^\mu}{\sigma_d^{\mu 2} + \delta \hat{q}} \right)^2 \right\} = 1 - \frac{1}{N} \alpha^{-1} \tag{96}$$

It is readily verified that $\delta \hat{q} \overset{\alpha \to \infty}{\to} \infty$. Indeed, taking this limit we get the identity

$$\frac{1}{P} \frac{1}{N} \sum_\mu \left\{ \sum_{d=2}^N 1 + 1 + \mathcal{O}\left(\frac{1}{\delta \hat{q}} \right) \right\} = 1 + \mathcal{O}\left(\alpha^{-1} \right) \tag{97}$$

To determine the asymptotic behavior of $\delta \hat{q}$, we expand the l.h.s of Eq. (96) for $\delta \hat{q} \to \infty$. Term by term, we have

$$\frac{1}{\frac{\sigma_d^{\mu 2}}{\delta \hat{q}} + 1} \sim 1 - \frac{1}{\delta \hat{q}} \sigma_d^{\mu 2} \tag{98}$$

$$\frac{1}{1+\frac{\sigma_1^{\mu 2}}{(1-a)^{-1}\delta\hat{q}}+\mathcal{O}\left(\frac{1}{\delta\hat{q}^2}\right)} \sim 1-\frac{1}{\delta\hat{q}}\left(1-a\right)\sigma_1^{\mu 2} \tag{99}$$

while the last term is of subleading order $\mathcal{O}\left(\frac{1}{\delta\hat{q}^2}\right)$. Plugging this back into Eq. (96) we get

$$\boxed{\delta\hat{q} \overset{\alpha\to\infty}{\sim} \alpha\left[\left(1-a\right)\left\langle\sigma_1^{\mu 2}\right\rangle_C+\sum_{d>1}\left\langle\sigma_d^{\mu 2}\right\rangle_C\right]} \tag{100}$$

Note that, at least for large α, the order parameter is proportional to $\sum_{d>1}\left\langle\sigma_d^{\mu 2}\right\rangle_C=\mathcal{O}\left(N\right)$. This will be useful to determine which terms can be neglected in the final expression for ΔE, because of subleading order in N, and which should be kept.

10. Local-global gap at the saddle point

At the saddle point, the local-global gap is given by

$$\Delta E=\min_m\left[\Delta E\left(m\right)\right] \tag{101}$$

where

$$\Delta E\left(m\right)=-\frac{1}{n}\frac{2}{P}\frac{1}{\beta}S_{\mathrm{RS}}=\frac{1}{P}\sum_{\mu=1}^{P}\sigma_1^{\mu 2}r^{\mu 2}+a\left\langle\sigma_1^{\mu 2}\right\rangle_C m^2-2\sqrt{a}m\frac{1}{P}\sum_{\mu=1}^{P}r^{\mu}\sigma_1^{\mu 2}+\frac{1}{P}\delta\hat{q}\,m^2 \tag{102}$$

$$-\delta\hat{q}\,a\,m^2-\frac{1}{P}\sum_{\mu=1}^{P}\sum_{d=2}^{N}S_d^{\mu}\left(r^{\mu}-\sqrt{a}\,m\right)^2+\frac{1}{N}\delta\hat{q}^2am^2\frac{1}{P}\sum_{\mu=1}^{P}\sum_{d=2}^{N}(U_d^{\mu})^{-1} \tag{103}$$

$$-\frac{1}{P}\sum_{\mu=1}^{P}\left(U_1^{\mu}-\sum_{d=2}^{N}S_d^{\mu}\right)^{-1}\left(\sigma_1^{\mu 2}-\sum_{d=2}^{N}S_d^{\mu}\right)^2\left(r^{\mu}-\sqrt{a}\,m\right)^2 \tag{104}$$

$\delta\hat{q}$ is determined by the self-consistent equation Eq. (95), and we have kept only those terms that are of leading order in both β as $\beta\to\infty$, and in N or P, in the limit $N,P\to\infty$, with $\frac{P}{N}=\alpha\in[0,\infty]$. Note that some terms in $\Delta E\left(m\right)$, such as $\frac{1}{P}\delta\hat{q}\,m^2$, may appear to be of subleading order in this limit, but are in fact to be kept because, at least for large α, the order parameter $\delta\hat{q}$ is of $\mathcal{O}\left(N\right)$.

Observing that Eq. (104) is quadratic in m, it's minimization is readily obtained as

$$\Delta E = -\frac{k_1^2}{4k_2} + k_0 \tag{105}$$

with

$$k_2 = a\left\langle \sigma_1^{\mu 2}\right\rangle_C + \frac{1}{P}\delta\hat{q} - a\delta\hat{q} - a\frac{1}{P}\sum_{\mu=1}^{P}\sum_{d=2}^{N} S_d^\mu + \frac{a}{N}\left(\delta\hat{q}\right)^2 \frac{1}{P}\sum_{\mu=1}^{P}\sum_{d=2}^{N}\left(U_d^\mu\right)^{-1}$$
$$- a\frac{1}{P}\sum_{\mu}\left(U_1^\mu - \sum_{d=2}^{N} S_d^\mu\right)^{-1}\left(\sigma_1^{\mu 2} - \sum_{d=2}^{N} S_d^\mu\right)^2 \tag{106}$$

$$k_1 = -2\sqrt{a}\frac{1}{P}\sum_{\mu=1}^{P} r^\mu \sigma_1^{\mu 2} + 2\sqrt{a}\frac{1}{P}\sum_{\mu=1}^{P}\sum_{d=2}^{D} S_d^\mu r^\mu + 2\sqrt{a}\frac{1}{P}\sum_{\mu}\left(U_1^\mu - \sum_{d=2}^{N} S_d^\mu\right)^{-1}\left(\sigma_1^{\mu 2} - \sum_{d=2}^{N} S_d^\mu\right)^2 r^\mu \tag{107}$$

$$k_0 = \frac{1}{P}\sum_{\mu=1}^{P}\sigma_1^{\mu 2} r^{\mu 2} - \frac{1}{P}\sum_{\mu=1}^{P}\sum_{d=2}^{N} S_d^\mu r^{\mu 2} - \frac{1}{P}\sum_{\mu=1}^{P}\left(U_1^\mu - \sum_{d=2}^{N} S_d^\mu\right)^{-1}\left(\sigma_1^{\mu 2} - \sum_{d=2}^{N} S_d^\mu\right)^2 r^{\mu 2} \tag{108}$$

11. Local-global gap in the $\alpha \to \infty$ limit

It is instructive to check that our expression for the local-global gap in the case of infinite categories, i.e. $\alpha \to \infty$, is recovered. As $\delta\hat{q} \to \infty$, most terms in Eqs. (106-108) that depend on $\delta\hat{q}$ vanish, except for the following three terms in Eqs. (106)

$$+\frac{1}{P}\delta\hat{q} \sim \frac{\alpha}{P}\left[(1-a)\left\langle \sigma_1^{\mu 2}\right\rangle_C + \sum_{d>1}\left\langle \sigma_d^{\mu 2}\right\rangle_C\right] = \frac{1}{N}\left[(1-a)\left\langle \sigma_1^{\mu 2}\right\rangle_C + \sum_{d>1}\left\langle \sigma_d^{\mu 2}\right\rangle_C\right] \tag{109}$$

$$-a\delta\hat{q} \tag{110}$$

$$+\frac{a}{N}\left(\delta\hat{q}\right)^2 \frac{1}{P}\sum_{\mu}\sum_{d>1}\left(U_d^\mu\right)^{-1} = \frac{a}{N}\frac{1}{P}\sum_{\mu}\sum_{d>1}\delta\hat{q}\left[\frac{1}{1+\frac{\sigma_d^{\mu 2}}{\delta\hat{q}}}\right] \sim a\delta\hat{q} - \frac{a}{N}\sum_{d>1}\left\langle \sigma_d^{\mu 2}\right\rangle_C \tag{111}$$

The second term, which is diverging, cancels with part of the third, and we have (assuming r^μ and σ_1^μ to be uncorrelated as in our infinite-P theory)

$$k_2 = a\left\langle \sigma_1^{\mu 2}\right\rangle_C + (1-a)\frac{1}{N}\sum_{d>1}\left\langle \sigma_d^{\mu 2}\right\rangle_C \tag{112}$$

$$k_1 = -2\sqrt{a}\left\langle r^\mu\right\rangle_C \left\langle \sigma_1^{\mu 2}\right\rangle_C \tag{113}$$

$$k_0 = \left\langle \sigma_1^{\mu 2}\right\rangle_C \left\langle r^{\mu 2}\right\rangle_C \tag{114}$$

which plugged into Eq. (105) give exactly our prediction for ΔE in the limit $\alpha \to \infty$, Eq. (41).

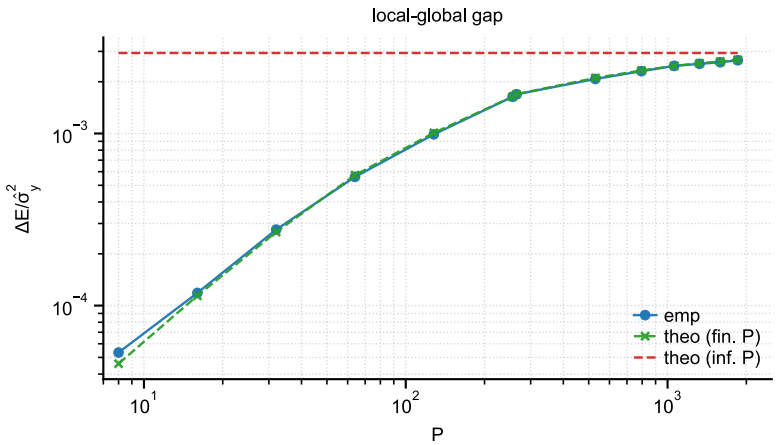

Figure 27. Theory validation. Training local-global error gap for category manifolds generated with the methods described in section VI D, for bounding-box feature L_h. Blue: measured gap; green: finite-P theory; red line: infinite-P theory.

D. Numerical validation

In the main text, we tested the theory predictions against the *test* regression error measured from the empirical data and found good agreement. Here, instead, we validate the theory in the setting for which it is derived: the *training* regression error, for manifolds generated under the theory generative model. In this case, we expect virtually perfect agreement up to finite-size corrections, which is indeed what we observe (**Fig. figure 27**).

The generative model still leaves freedom in the choice of its parameters, including the alignment a, the local scales r^μ, and the manifold covariances Σ^μ. In particular, the entries $\sigma_1^{\mu 2}$ and $\sigma_{d>1}^{\mu 2}$ contribute to the SNR, while the off-diagonal terms Σ_{1d}^μ determine how the feature-encoding direction is embedded within each manifold. To test the theory in a realistic regime, we chose these parameters to match those measured from the category manifolds at the feature layer of network CR (and the corresponding local feature-encoding directions for a given bounding-box feature). Concretely, we sampled $u_1^\mu = \sqrt{a}\, u_1 + \sqrt{1-a}\, \eta^\mu$,with u_1 and a estimated from the data. Similarly, we sampled $r^\mu \sim \mathcal{N}\big(\mathrm{Avg}_\mu[r^\mu], \mathrm{Var}_\mu[r^\mu]\big)$, and we took the covariance matrices Σ^μ directly from the empirical category manifolds. When P exceeded the 265 category manifolds available from the data, we sampled Σ^μ uniformly from these 265 empirical covariances for categories with $\mu > 265$. **Fig. figure 27** shows the local-global error gap as measured from this set of category manifold, comapred against the infinite-P and finite-P theory.