Separability and Geometry of Object Manifolds in Deep Neural Networks

Uri Cohen*[1], SueYeon Chung*[2,4], Daniel D. Lee[3], and Haim Sompolinsky†[1,4]

[1]Edmond and Lily Safra Center for Brain Sciences
Hebrew University of Jerusalem, Israel
[2]Department of Brain and Cognitive Sciences
Massachusetts Institute of Technology, Cambridge, MA, USA
[3]Department of Electrical and Computer Engineering
Cornell Tech, New York, NY, USA
[4]Center for Brain Science
Harvard University, Cambridge, MA, USA

Abstract

Stimuli are represented in the brain by the collective population responses of sensory neurons, and an object presented under varying conditions gives rise to a collection of neural population responses called an "object manifold." Changes in the object representation along a hierarchical sensory system are associated with changes in the geometry of those manifolds, and recent theoretical progress connects this geometry with "classification capacity," a quantitative measure of the ability to support object classification. Deep neural networks trained on object classification tasks are a natural testbed for the applicability of this relation. We show how classification capacity improves along the hierarchies of deep neural networks with different architectures. We demonstrate that changes in the geometry of the associated object manifolds underlie this improved capacity, and shed light on the functional roles different levels in the hierarchy play to achieve it, through orchestrated reduction of manifolds' radius, dimensionality and inter-manifold correlations.

Introduction

The visual hierarchy of the brain has a remarkable ability to identify objects despite differences in appearance due to changes in position, illumination, and background conditions [1]. Recent research in machine learning has shown that deep convolutional neural networks (DCNNs) [2] can perform invariant object categorization with almost human-level accuracy [3], and that their network representations are similar to the brain's [4][5][6]. DCNNs are therefore very important as models of visual hierarchy [7][8][9], though understanding their operational capabilities and design principles remain a significant challenge [10].

In a visual hierarchy, the neuronal population response to stimuli belonging to the same object defines an "object manifold." The brain's ability to discriminate between objects can be mapped to the separability of object manifolds by a simple, biologically plausible readout, modeled as a *linear* hyperplane [11]. It has been hypothesized that the visual hierarchy "untangles" object manifolds to transform inseparable object manifolds into linearly separable ones. This approach underlies a number of studies on object representation in the brain [1], as well as on deep neural networks [12],[13],[14],[15]. As separability of manifolds depends on numerous variables – size and shape of the manifolds, number of neurons, cardinality of the set of manifolds, specific target labels, among others – it has been difficult to elucidate which specific properties of the representation truly contribute to untangling. Here we apply the theory of linear separability of manifolds [16] to establish analytically that separability depends on three measurable quantities, manifold dimension and extent, and inter-manifold correlation.

*These authors contributed equally to this work.
†corresponding author

For a system consisting of N neurons representing P object manifolds, the system "load" is defined by the ratio $\alpha = P/N$ [17]. In the regime where P and N are large, our theory shows the existence of a critical load value α_c, called "manifold classification capacity", above which $(P > \alpha_c N)$ object manifolds are not linearly separable with high probability. Intuitively, this definition of capacity serves as a measure of the linearly decodable information *per neuron* about object identity.

We first illustrate the role of manifold geometry on classification capacity by considering several limiting extremes. The largest possible value of α_c is 2 [17][16] and is achieved for manifolds consisting of single points [18], i.e. fully invariant object representations. For the lower bound, we consider *unstructured* manifolds, in which a set of $M \cdot P$ points is randomly grouped into P manifolds. Since the points of each manifold do not share any special geometric features, the capacity is equal to that of $M \cdot P$ points, i.e. the system is separable while $M \cdot P < 2N$ [17][19], or equivalently $\alpha_c = 2/M$. Thus, the capacity of structured "point-cloud" manifolds consisting of M points per manifold obeys $\frac{2}{M} \leq \alpha_c \leq 2$ (illustrated in figure 1a). Another illuminating limit is that of manifolds with infinite sizes, each spanning D randomly oriented axes of variation (illustrated in figure 1b), in which case $\alpha_c = 1/(D + 1/2)$ [16].

Since real world manifolds are expected to span high dimensional sub-spaces and consist of a large or infinite number of points, a substantial capacity implies constraints on both the manifold dimensionality and extent. Realistic manifolds (illustrated in 1c) are expected to show inhomogeneous variations along different manifold axes, raising the question of how to assess their *effective* dimensionality and extent. Our theory provides a precise definition of the relevant manifold dimension and radius, denoted D_M and R_M, that can describe their separability.

These quantities are derived from the structure of the optimal plane separating the manifolds, defined as the plane with the maximum margin. Extending the well known notion of support vectors, we show that each manifold contributes a unique point, called an anchor point, residing on the margins of the separating plane, as illustrated in figure 1d. Thus for a given manifold, its anchor point depends not only on its shape but also on the statistics of the ensemble of manifolds participating in the separation task. As the ensemble is varied, the manifold's anchor point will change, thereby generating a distribution of anchor points under the ensemble. The effective radius is the total variance of the anchor points and effective dimension is their spread along the different manifold axes.

Using statistical mechanical mean field techniques, we derive algorithms that allow measuring the capacity, R_M and D_M for manifolds given by either empirical data samples or from parametric generative models [16]. This mean field theory previously assumed that the position and orientation of different manifolds are uncorrelated. Here we extend the theory and apply it to realistic manifolds, such as those arising at different layers of deep network architectures, where substantial correlations between manifolds are expected. Our theory connects the functional description of classification capacity with a geometric description of manifold radii and dimensions. It allows us to systematically study how the layers of deep networks transform object manifolds, illuminating the effect of architectural building-blocks and non-linear operations on shaping manifold geometry and between-manifold correlations.

Understanding how information is represented by the population activity of neurons has been a key focus of research in computational neuroscience [20][10]. However, properly quantifying the representations between neural networks and across layers within a neural network remains an open and challenging problem in our understanding of neural networks. Recently, canonical correlation analysis (CCA) has been proposed to compare the representations in hierarchical deep networks [21][22]. Another approach, representational similarity analysis (RSA), uses similarity matrices to determine which stimuli are more correlated within neural data and in network representations [23][4][7]. Others have considered various geometric measures such as curvature [24] and dimensionality to capture the structure within neural representations [25][26][27][28][29][30][31]; but it unclear how these different measures are related to task performance such as classification accuracy. Conversely, others have explored functional aspects by using representations from different layers for use in transfer learning [32], or object classification [30], but without offering any insight as to why performance improves or deteriorates. Other attempts to characterize representations in deep neural hierarchies focused on single-neuron properties and their ability to support object invariance [13][33][34].

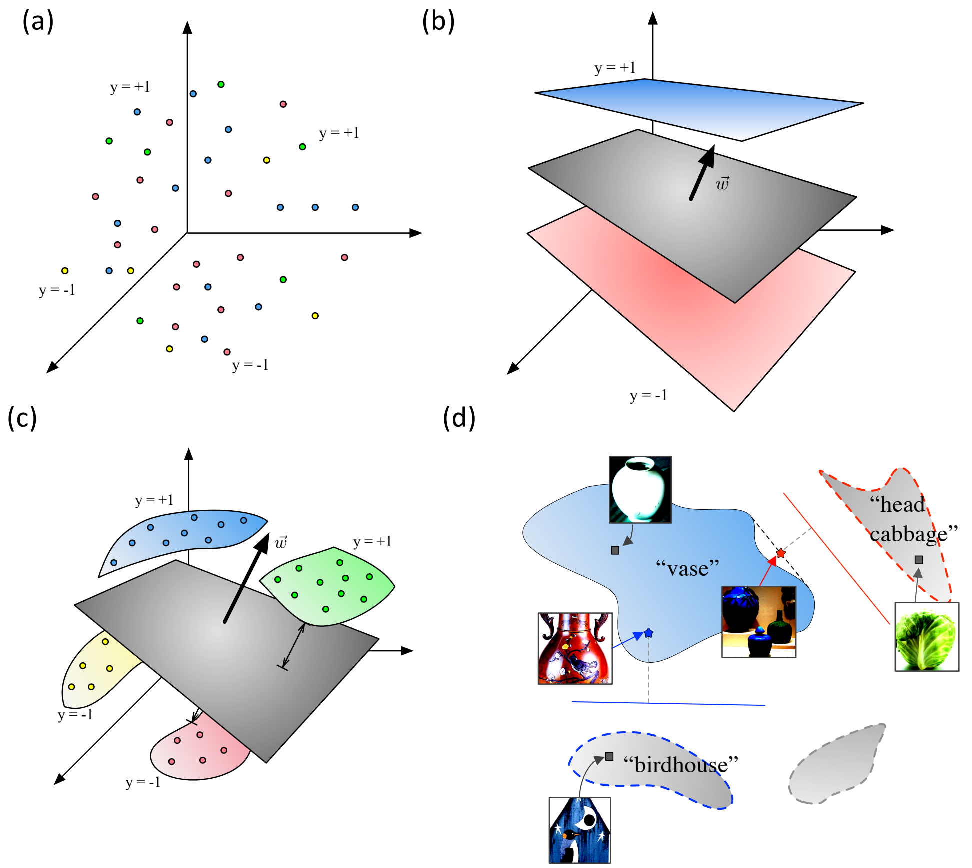

Figure 1: Illustration of manifolds extreme cases and anchor points. (a) Illustration of 4 manifolds (colored red, green, blue and yellow) consisting of random points. Blue and green manifolds have a target label of +1 while red and yellow manifolds have a target label of −1. (b) Illustration of 2 manifolds (colored red and blue) consisting of infinite size along a finite number of axes. A separating hyperplane (gray) is defined by a vector $\vec{w}$ and is orthogonal to the manifold subspaces. (c) Illustration of 4 manifolds (colored and labeled as in (a)) consisting of finite-sized manifolds. A separating hyperplane (gray) is defined by a vector $\vec{w}$. (d) Illustration of anchor points (stars), each a point in the convex hull of a manifold, defined as a linear sum of support vectors in a classification against another manifold. An anchor point of a 'vase' manifold (red star) against a 'head cabbage' class manifold (red dashed). An anchor point of a 'vase' manifold (blue star) against a 'birdhouse' class manifold (blue dashed). The typical example of each class manifold is marked as a square. The image associated with an anchor point is an image whose representation is closest to the anchor point on a 'vase' manifold in the last layer of AlexNet, where the anchor point is defined by the optimal hyperplane for classification between a 'vase' vs. a 'head cabbage', or vs. a 'birdhouse' class manifolds (where each of these manifolds is defined by the top half exemplars in terms of the scores, as detailed in Methods).

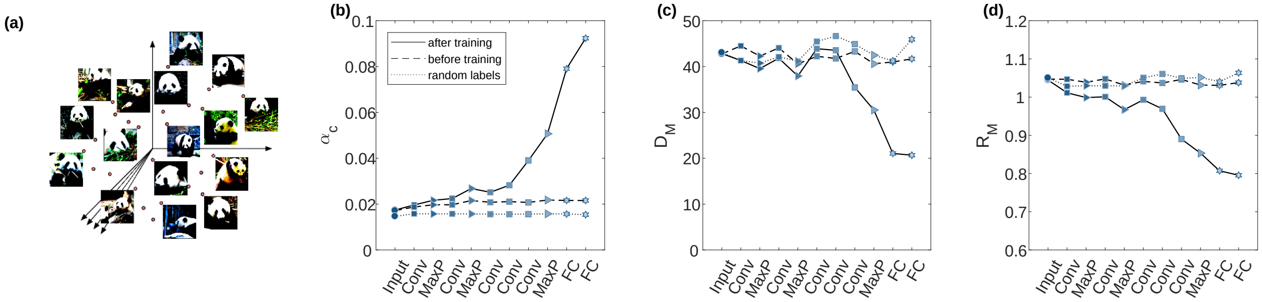

Figure 2: Manifolds capacity and geometry changes during learning. (a) Illustration of a point-cloud manifold for the 'giant panda' class from ImageNet, in high-dimensional state space. (b-d) Changes in capacity and manifold geometry along the layers of AlexNet point-cloud manifolds (top 10%) for fully-trained network (full line), randomly initialized network (dashed line) or randomly shuffled object manifolds (dotted line). (b) Changes in classification capacity. (c) Changes in mean manifold dimension. (d) Changes in mean manifold radii. The x-axis labels provides abbreviation of the layer types ('Input'- pixel layer, 'Conv'- convolutional layer, 'MaxP'- max-pooling layer, 'FC'- fully connected layer). Marker shape represents layer type (circle- pixel layer, square- convolution layer, right-triangle- max-pooling layer, hexagon- fully connected layer). Features in linear layers ('Conv', 'FC') are extracted after a ReLU non-linearity.

Results

Manifold capacity and geometry evolve during learning and across layers to enhance linear separability

To study the reformatting of object manifolds in deep networks, we apply our theory to DCNNs trained for object recognition tasks using supervised backpropagation training on large labeled datasets such as ImageNet [35]. Several state-of-the-art networks, such as AlexNet [36] and VGG-16 [37], share similar computational building blocks; those networks are composed of alternating layers of linear convolutions, point-wise ReLU nonlinearities and max-pooling, followed by several fully connected layers and a final soft-max linear classifier.

To demonstrate the applicability of our methods we measure classification capacity and analyze the geometry of manifolds consisting of high scoring samples (see Methods) from ImageNet classes [35] (illustrated in figure 2a) processed by AlexNet [36]. The full line in figure 2b demonstrates that the manifold classification capacity, measured using our mean-field theory (see Methods) increases along the hierarchy for a fully-trained network, while figure 2c-d exhibit concomitant decrease in manifolds' dimension and radius (measured using the theory, see Methods). The manifold radius represents the extent of a manifold normalized by the norm of its center (or equivalently by the typical distance to other manifolds), while the manifold dimension represents the effective number of axes of variation. We find the mean manifold dimension undergoes a pronounced decrease in the last layers, from more than 40 in the early layers to about 20 in the last layer (i.e. the network output or "feature layer"). Interestingly, mean manifold radius changes more gradually across layers yielding a modest decrease from above 1 in the first layer (i.e. the network input or "pixel layer") to about 0.8 in the last layer.

An important question in the theory of DCNNs is to what extent their impressive performance results from the underlying "inductive bias" such as the choice of architecture, nonlinearities, and initializing weights, prior to the training. We address this issue by processing the same images using an untrained network (with the same architecture but randomly initialized weights). As indicated by the dashed lines in figure 2b-d, an untrained network does not show any improvement in manifold representation, neither with respect to their classification capacity, nor with respect to their geometry, displaying an almost constant low capacity and high dimension, radius values across the entire hierarchy. To understand the origin of this baseline we repeated our analysis of the manifold representations in the fully trained network but with shuffled assignment of images into objects. This shuffling destroys any structure of the data, leaving only residual capacity due to the finite number of samples per manifold. As indicated by dotted lines in 2b-d, the properties of the shuffled manifolds are constant across the hierarchy, with values close to those of the untrained network. Importantly, the values of the shuffled manifolds are very close to those exhibited by the pixel layer representation of the true manifolds; this implies that for that layer, the manifold variability is so large that there is very little evidence of any underlying structure. In contrast, the last layers of

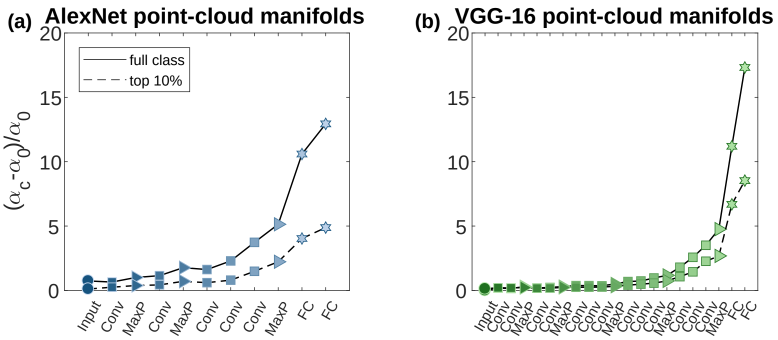

Figure 3: Capacity of point-clouds manifolds of ImageNet classes. (a-b) Normalized classification capacity for point-cloud manifolds of ImageNet classes (full line: full class manifolds; dashed line: top 10% manifolds) along the layers of AlexNet (a) and VGG-16 (b). Capacity is normalized by $\alpha_c = 2/\langle M_\mu \rangle_\mu$, the value expected for unstructured manifolds (see main text; M_μ denotes the number of samples from object μ). The x-axis labels provides abbreviation of the layer types. Marker shape represents layer type (circle- pixel layer, square- convolution layer, right-triangle- max-pooling layer, hexagon- fully connected layer). Features in linear layers are extracted after a ReLU non-linearity. Color (blue- AlexNet, green- VGG-16) changes from dark to light along the network.

the trained network exhibit substantial improvement in capacity and geometry, reflecting the emergence of robust object representations.

Capacity of point-cloud manifolds increases along the hierarchy for different networks and variability levels

How does manifold classification capacity depend on the statistics of images representing each object? To answer those questions we consider manifolds with two levels of variability, low variability manifolds consisting of images with the top 10% score ("top 10%", see Methods) and high variability manifolds consisting of all (roughly 1000) images per class ("full class"). Both manifold types exhibit enhanced capacity in the last layers, as shown in figure 3a which depicts the normalized manifold capacity. As the absolute value of capacity depends not only on the geometry of the point-cloud manifold but also on the number of samples per manifold (which differ between those two classes of manifolds), we emphasize the improvement in capacity relative to manifolds with shuffled labels (by subtraction and dividing by capacity expected for random points). This normalization highlights how manifold capacity improves from random-points level at the pixel layer (around 0) to an order of magnitude above it in the top layers of the network. Interestingly, although the high score manifolds exhibit higher absolute capacity than the full manifolds (since the former have fewer points and less variability, see figure SI1) the improvement *relative* to shuffled data is actually larger in the full manifolds.

How does the capacity vary between different DCNNs trained to preform the same object recognition task? Figure 3b shows the corresponding capacity results for a deeper DCNNs, VGG-16 [37]. Despite difference in the architecture the pattern of improved capacity is quite similar, with most of the increase taking place in the final layers of the networks. Notably, the deeper network exhibits higher capacity in the last layers compared to the shorter network, consistent with the improved performance of VGG-16 in the ImageNet task (this trend continues further with ResNet-50 [38], a deeper network with higher ImageNet performance and higher capacity, figure SI1).

Capacity of smooth manifolds increases along the hierarchy for different networks and variability levels

While point-cloud manifolds are natural to consider in the context of deep network training from examples, they represent a special kind of manifold with non-smooth nature and finite number of samples. Thus we turn to consider manifolds which naturally arise when stimuli has several latent parameters which are smoothly varied. We create a smooth manifold by warping a template image (again from the ImageNet data-set [35]) by multiple affine transformations (see illustration at figure 4a, and Methods). Such manifolds are computationally easy to

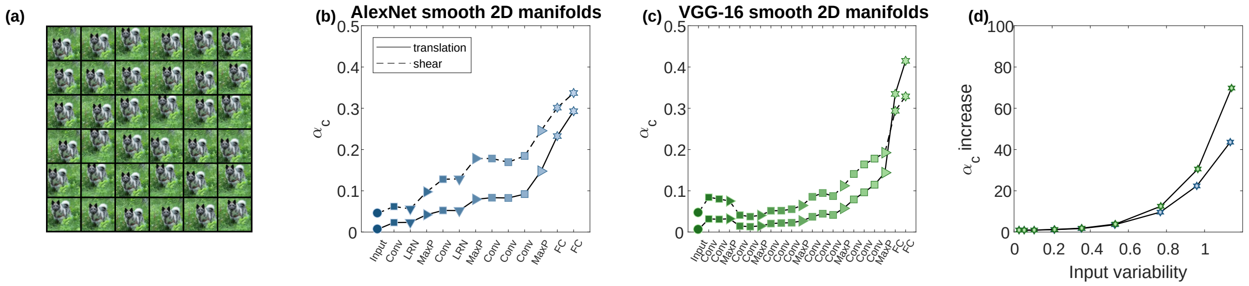

Figure 4: Capacity of smooth manifolds from warped ImageNet images. (a) Illustration of smooth, densely sampled affine transformed images; 36 samples from a 2-d translation manifold. Each manifold sample is associated with a coordinate specifying the horizontal and vertical translation of a base-image and corresponds to an image where the object is warped using the appropriate affine transformation. (b-c) Classification capacity for 2-d smooth manifolds (full line: translation; dashed line: shear) along the layers of AlexNet (b) and VGG-16 (c). The x-axis labels provides abbreviation of the layer types. Marker shape represents layer type (circle- pixel layer, square- convolution layer, right-triangle- max-pooling layer, hexagon- fully connected layer, down-triangle- local normalization). Features in linear layers are extracted after a ReLU non-linearity. Color (blue- AlexNet, green- VGG-16) changes from dark to light along the network. (d) Capacity increase from the input (pixel layer) to the output (features layer) of AlexNet (blue markers) and VGG-16 (green markers) for 2-d translation smooth manifolds. The capacity increase is specified as ratio of capacity at the last layer relative to the pixel layer (y-axis), at different levels of stimuli variation measured using SI equation 3 at the pixel layer (x-axis).

sample densely, thus allowing us to extrapolate to the case of infinite number of samples. Furthermore, it allows us to independently manipulate the intrinsic manifold dimension (controlled by the number of affine transform parameters) and the manifold extent (controlled by the maximal displacement of the object, see Methods).

The capacity for two representative smooth manifolds at different layers across two deep hierarchies is shown in figure 4b-c. A smooth, almost monotonic, improvement of capacity is observed along both networks, with most of the absolute increase achieved in the last layers, similar to the behavior observed for point-cloud manifolds. The relative increase in capacity from the first (pixel) layer to the last layer is shown in figure 4d for manifolds with different variability levels (measured at the pixel layer). Notably this increase is growing faster than manifold variability itself, reaching an increase of almost two orders of magnitude for manifolds with the highest variability considered here. Those observations persist for both 1-d and 2-d smooth manifolds and in all deep networks considered, including ResNet-50 (figure SI2). We note that due to the dense sampling, capacity measured here is a good approximation of the asymptotic value of the capacity of the underlying smooth manifolds (with number of samples going to infinity, see figure SI13).

Deep network layers reduce the dimension and radius of object manifolds

Changes in the measured classification capacity can be traced back to changes in manifold geometry along the network hierarchy, namely manifold radii and dimensions, which can be estimated from data using our mean field theory (see Methods, equations 3-4). Mean manifold dimension and radius along the deep hierarchies are shown in figures 5a,b respectively. The results exhibit a surprisingly consistent pattern of changes in the geometry of manifolds between different deep network architectures, along with interesting differences between the behavior of point-cloud and smooth manifolds. Figure 5a suggests that decreased dimension along the deep hierarchies is the main source of the observed increase in capacity from figures 3, 4 (and similar results for ResNet-50 are shown in figure SI3). Both point-cloud and smooth manifolds exhibit non-monotonic behavior of dimension, with increased dimension in intermediate layers; this increase of dimensionality can also be observed in other measures such as spectral participation ratio. A notable difference between those manifold types is a very large decrease in dimensions at the first layer of smooth translation manifolds (figure 5a, bottom), which can be seen as evidence for the ability of this convolution layer to overcome much of the effects of translation.

Figure 5b shows a small decrease in manifold radii along the deep hierarchy and across all manifolds (additional ResNet-50 results are shown in figure SI4). A slow but monotonic decrease is observed in point-cloud manifolds while for smooth manifolds we find a sharp decrease in the first layer and the final (fully-connected) layers, without decrease in radii at intermediate layers. Those differences may reflect the fact that the high variability of point-cloud

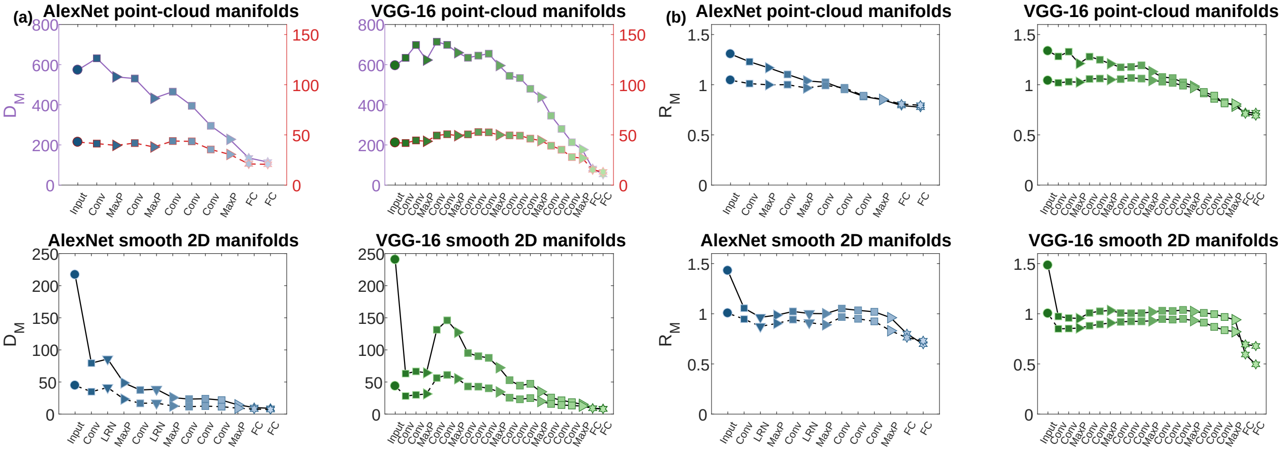

Figure 5: Manifold geometry. (a) Mean manifold dimension for point-cloud manifolds of AlexNet and VGG-16 (top, full-line: full-class manifolds, dashed-line: top 10% manifolds) and smooth 2-d manifolds for the same deep networks (bottom, full-line: translation manifolds, dashed-line: shear manifolds). Values of point-cloud top 10% manifolds are showed against a secondary y-axis (color-coded by the markers edge) to improve visibility. (b) Mean manifold radius for point-cloud manifolds of AlexNet and VGG-16 (top, full-line: full-class manifolds, dashed-line: top 10% manifolds) and smooth 2-d manifolds for the same deep networks (bottom, full-line: translation manifolds, dashed-line: shear manifolds).
The x-axis labels provides abbreviation of the layer types. Marker shape represents layer type (circle- pixel layer, square- convolution layer, right-triangle- max-pooling layer, hexagon- fully connected layer, down-triangle- local normalization layer). Features in linear layers are extracted after a ReLU non-linearity. Color (blue- AlexNet, green- VGG-16) changes from dark to light along the network.

manifolds needs to be reduced incrementally from layer to layer (both in terms of radius and dimension), utilizing the increased complexity of downstream features, while the variability created by local affine transformations is handled mostly by the local processing of the first convolutional layer (consistent with [34] reporting invariance to such transformations in the receptive field of early layers). The layer-wise compression of affine manifold plateaus in the subsequent convolutional layers, as the manifolds are already small enough. As signals propagate beyond the convolutional layers, the fully-connected layers resume geometric compression utilizing their many parameters, for both manifold types.

This geometric description allows us to further shed light on the structure of the smooth manifolds used here. For radius up to 1, the dimension of 2-d manifolds (e.g. created by vertical and horizontal translation) is just the sum of the dimensions of the two corresponding 1-d manifolds with the same maximal object displacement (figure SI5a); only for larger radii there is super-additive dimensions for 2-d manifolds. On the other hand, for all levels of stimulus variability the radius of 2-d manifolds is about the same as the value of the corresponding 1-d manifolds (figure SI5b). This highlights the non-linear super-additive mixing of manifold dimensions in large manifolds.

Deep network layers reduce correlations between object representations

Manifold geometry considered above characterizes the variability in object manifolds' shape but not the possible relations between them. Here we focus on the correlations between the centers of different manifolds (hereafter: center correlations), which may create clusters of manifolds in the representation state space. Our theory predicts that these correlations reduce classification capacity (SI section 3.1); thus, the amount of center correlations at each layer of a deep network is a computationally-relevant feature of the underlying manifold representation.

Importantly, for both point-cloud and smooth manifolds we find that in a network trained for object classification, center correlations decrease along the deep hierarchy (full lines in figure 6a-b; additional VGG-16 and ResNet-50 results are shown in figure SI6). As center correlations lower capacity, this decrease is interpreted as incremental improvement of the neural code for objects, and supports the improved capacity in figures 3-4. This is to be compared with center correlations at the same networks prior to training (dashed lines in figure 6a-b) where this decrease is not evident (except for the first convolutional layer in manifolds created by affine transformations, figure 6b). Thus this decorrelation of manifold centers is a result of the network training. Interestingly, the center

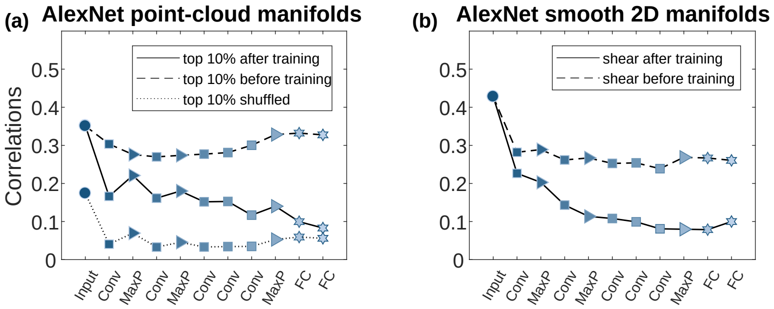

Figure 6: Correlations between manifolds. Changes of mean between-manifold correlations along the layers of AlexNet. (a) Center correlations for top 10% point-cloud manifolds in fully-trained network (full line), randomly initialized network (dashed line) or randomly shuffled object manifolds (dotted line). (b) Center correlations for smooth 2-d shear manifolds in fully-trained network (full line) or randomly initialized network (dashed line). The x-axis labels provides abbreviation of the layer types. Marker shape represents layer type (circle- pixel layer, square- convolution layer, right-triangle- max-pooling layer, hexagon- fully connected layer). Features in linear layers are extracted after a ReLU non-linearity. Color changes from dark to light along the network. Center correlations are $\rho_{CC} = \langle |\vec{x}^{\mu} \cdot \vec{x}^{\nu}|/||\vec{x}^{\mu}|| \cdot ||\vec{x}^{\nu}||\rangle_{\mu \neq \nu}$ where $\vec{x}^{\mu}$ is the center of object μ (SI equation 1).

correlations of shuffled manifolds where samples are randomly assigned to objects exhibit lower levels of correlations, which remain constant across layers after an initial decrease at the first convolutional layer.

Another source of inter-manifold correlations are correlations between the axes of variation of different manifolds; those also decrease along the network hierarchies (figure SI6) but their effect on classification capacity is small (as verified by using surrogate data, figure SI7).

Deep network building-blocks have a consistent effect on manifold geometry and correlations

To better understand the enhanced capacity exhibited by DCNNs we study the roles of the different network building-blocks. Based on our theory, any operation applied to a neural representation may change capacity by either changing the manifolds' dimensions, radii, or the inter-manifold correlations (where a reduction of these measures is expected to increase capacity).

Figure 7a-b shows the effect of single operations used in AlexNet and VGG-16. We find that the ReLU nonlinearity usually reduces center correlations and manifolds' radii, but increases manifolds' dimensions (figure 7a). This is expected as the nonlinearity tends to generate a sparse, higher dimensional, representations. On the other hand, pooling operations decrease both manifolds' radii and dimensions but usually increase correlations (figure 7b), the latter is presumably due to the underlying spatial averaging. Such clear behavior is not evident when considering convolutional or fully-connected operations in isolation (figure SI8).

In contrast to single operations, our analysis reveal that the computational building-blocks commonly used in AlexNet and VGG-16 perform consistent transformation on manifold properties (figure 7c-d). In those networks, the initial building blocks consist of sequences of convolution, ReLU operation followed by pooling, which consistently act to decrease correlations and tend to decrease both manifolds' radii and dimensions (figure 7c). On the other hand, the final building-block, composed of fully-connected operation followed by ReLU, decreases manifolds' radii and dimensions, but may increases correlations (figure 7d), similarly to the max-pooling operation (figure 7b). This analysis highlights how the networks architecture consistently reduces manifold correlations at the initial stages of the network and reduces the manifolds' dimension and radius at the final stages. It also explains the non-monotonic behavior of the manifold dimensions in figure 5a, where dimension increases in sequences of convolutional stages without intermediate pooling. Furthermore, as composite building blocks show more consistent behavior than individual operations, we understand why DCNNs with randomly initialized weights do not improve manifold properties. Only by appropriately trained weights, the combination of operations with often opposing effects yields a net improvement in manifold properties.

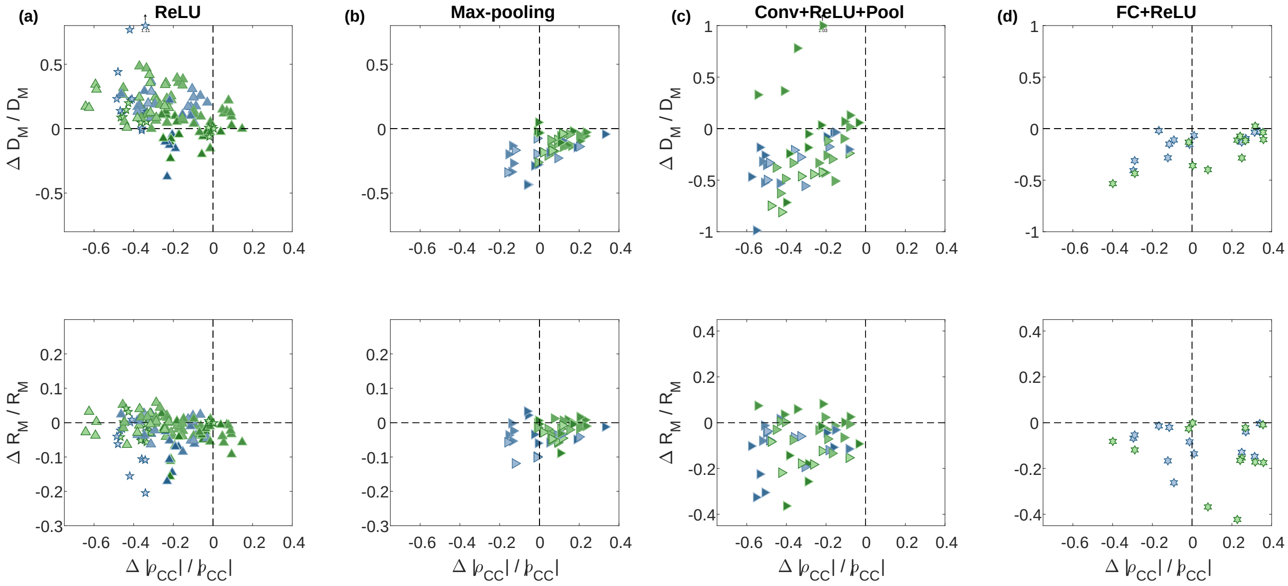

Figure 7: Manifold property changes by network building blocks. Changes in the relative manifold properties between the input and the output of different network building-blocks, shown as change in dimension vs change in center correlations (top) and change in radius vs change in center correlations (bottom). Each panel pools results from a specific building-block in AlexNet (blue markers) and VGG-16 (green markers) for both point-cloud manifolds (full class, top 10%) and smooth manifolds (1-d and 2-d, translation and shear). Marker shape represents layer type (square- convolution layer, right-triangle- max-pooling layer, hexagon- fully connected layer). For layer sequences marker shape represents the last layer in the sequence. For isolated ReLU marker shape represent previous layer type; pentagon- ReLU after fully-connected layer, up-triangle- ReLU after convolution layer. Color changes from dark to light along the network.
(a) Changes in manifold properties for isolated ReLU operations.
(b) Changes in manifold properties for isolated Max-pooling operations.
(c) Changes in manifold properties for a common sequence of operations: one or more repetitions of convolution, ReLU operations, with or without intermediate normalization operation, ending with a max-pooling operation.
(d) Changes in manifold properties for a common sequence of operations: fully-connected, ReLU operations.

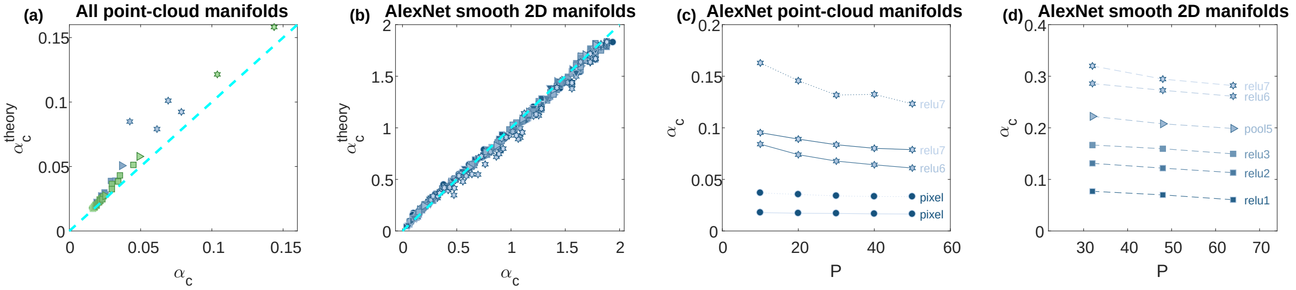

Figure 8: Theoretical predictions.
(a) Comparison of numerically measured capacity (x-axis) with the theoretical prediction (y-axis) for AlexNet, VGG-16 at different layers along the hierarchy (top 10% point-cloud manifolds).
(b) Comparison of numerically measured capacity (x-axis) with the theoretical prediction (y-axis) for AlexNet at different layers along the hierarchy and different levels of manifold variability (smooth 2-d manifolds).
(c) Numerically measured capacity (y-axis) at different number of objects (x-axis) for point-cloud manifolds at different layers (dashed line: top 10% manifolds; dotted line: top 5% manifolds).
(d) Numerically measured capacity (y-axis) at different number of objects (x-axis) for smooth 2-d shear manifolds. Marker shape represents layer type (circle- pixel layer, square- convolution layer, right-triangle- max-pooling layer, hexagon- fully connected layer, down-triangle- local normalization layer). Color (blue- AlexNet, green- VGG-16) changes from dark to light along the network.

Comparison of theory with numerically measured capacity

The results presented so far were obtained using algorithms derived from a mean field theory which is exact in the limit of large number of neurons N and number of manifolds P with additional simplifying statistical assumptions (SI section 3.1). But can the mean field equations describe capacity of finite-sized networks with realistic data? To address this question, we have numerically computed capacity at each layer of the network, using a linear classifier trained to classify object manifolds with random binary labels (using recently developed efficient algorithms for manifold classification [39], see Methods). Comparing the numerically measured values to theory shows good agreement for both point-cloud manifolds (figure 8a) and smooth manifolds (figure 8b for smooth 2-d manifolds in AlexNet and figure SI11 for smooth 2-d and 1-d manifolds in AlexNet, VGG-16 and ResNet-50). This agreement is a remarkable validation of the applicability of mean field theory to representations of realistic data sets generated by complex networks.

A fundamental prediction of the theory is that the maximal number of classifiable manifolds P_c is extensive, namely grows in proportion to the number of neurons in the representation N, hence their ratio α_c is unchanged. We validated this prediction in realistic manifolds, by measuring numerically the capacity upon changing both the number of neurons used for classification and the number of data manifolds. The results for both point-cloud manifolds (figure 8c) and smooth manifolds (figure 8d) indeed exhibits only a small dependence of capacity on the number of objects on which it is measured. For both manifold types capacity seem to saturate such that the value at a finite value of $P \approx 50$ is close to the asymptotic limit (additional results for 1-d and 2-d smooth manifolds with different variability levels are provided in figure SI9). Note that the mean field prediction of *extensivity* of classification holds for manifold ensembles whose individual geometric measures such as manifold dimension and radius do not scale with the representation size but retain a finite limit when N grows. Indeed, we found that the radius and dimensions of our data manifolds also show little dependence on the number of neurons for values of N larger than several hundred (figure SI10), consistent with the exhibited extensive capacity. The saturation of α_c and manifold geometry with respect to N implies that they can be estimated on a sub-sampled set of neurons (or subset of random projections of the representation) at each layer (Methods), a fact that we utilized in calculating our results of figures 2-7 and has important practical implications for the applicability of these measures to large networks.

Discussion

The goal of our study was to explicitly connect the *computational*, *geometric*, and *correlation*-based measures to describe "untangling of manifolds" in deep networks, using a new theoretical framework. To do this, we introduce classification capacity as a measure of the amount of decodable information per neuron about object manifolds. Combining tools from statistical physics and insights from high-dimensional geometry, we develop a mean field estimate of capacity and relate it to geometric measures of object manifolds as well as their correlation structure. Using these measures, we analyze how manifolds are reformatted as the signals propagate through the deep networks to yield an improved invariant object recognition at the last stages.

We find that the classification capacity of the object manifolds increases across the layers of trained DCNNs. At the input pixel layer, the extent of intra-manifold variability is so large that the resultant manifold properties are virtually indistinguishable from random points. Nevertheless, the trained networks show significant increases in capacity along with an overall reduction in the mean manifold dimensions and radii across the layers. In contrast, networks with the same architectures and random weights do not exhibit any improvement in capacity nor display any significant changes in dimension or extent of the manifolds. Similarly, shuffled data processed by trained networks also show no emergence of geometric structure in any layer. Our results show that both learning and input object manifold structure are necessary for the trained networks to successfully increase classification capacity and reformat the manifold geometry across the network layers.

For both point-cloud and smooth manifolds across multiple DCNN architectures (AlexNet, VGG-16, and ResNet-50), we find improved manifold classification capacity (figures 3,4,SI1,SI2) associated with decreased manifold dimension and radius across the layers (figures 5,SI3,SI4). Our findings suggest that different network hierarchies employ similar strategies to process stimulus variability, with image warping handled by the initial convolution and final layers, and intermediate layers needed to gradually reduce the dimension and radius of point-cloud manifolds. We find that lowering the dimensionality of *each object manifold* is one of the primary reasons for the improved separation capabilities of the trained networks.

The networks also effectively reduce the inter-manifold correlations, and some of the improved capacity exhibited by the DCNNs results from decreased manifold center correlations across the layers (figures 6, SI6). As with the manifold geometrical measures, the improved decorrelation is specific to networks with trained weights. In networks with random weights, the center correlations remain high across all layers. Other studies of object representations in DCNNs and in the visual system focused on the block structure of correlation matrices reflecting higher order object categories (such as animals and inanimate objects); to the best of our knowledge this is the first study that highlights the overall decrease of correlations between object manifolds and its computational significance. Decorrelation between *neuronal responses* has been one of earliest principles proposed to explain principles of neural coding in early stages of sensory processing [40][41]. Interestingly, here we find that decorrelation between *object representations* is a substantial computational principle in higher processing stages.

In this work we have not addressed the question of extracting physical variables of stimuli, such as pose or articulation. In principle, reformatting of object manifolds might also involve alignment of their axes of variation, so that information about physical variables can be easily readout by projection on subspace orthogonal to those which contain the object identity information [42]. Alternatively, separate channels may be specialized for such tasks. Interestingly, in artificial networks, the axes-axes alignment across manifolds is reduced in late stages (see figure SI6), consistent with the fact that they were trained to perform only object recognition tasks. This is qualitatively consistent with the study of information processing in deep networks [43] which proposes a systematic decrease along the network hierarchy in information about the stimulus accompanied by increased representation of task related variables. It would be interesting to examine systematically if high level neural representations in the brain such as IT cortex show similar patterns or channel both type of information in separate dimensions [11][44].

Our analysis of the effect of common computational building blocks in DCNNs (such as convolution, ReLU nonlinearity, pooling and fully connected layers) shows that single stages do not explain the overall improvement in manifold structure. Some individual stages transform manifold geometry differently dependent on their position in the network (e.g. convolution, figure SI8). Other stages exhibit trade-offs between different manifold features; for instance, the ReLU nonlinearity tends to reduce radius and correlations but increase the dimensionality. In contrast, composite building blocks, comprising a sequence of spatial integration, local nonlinearities and non-local pooling yield a consistent reduction in manifold radius and dimension in addition to reduced correlations across the different manifold types and network architectures.

We find very similar behavior in manifolds propagated through another class of deep networks, Residual Networks, that are not only much deeper but also incorporate a special set of skip connections between consecutive modules. In an analysis of ResNet-50 with 50 layers (SI1-SI6), we find quite similar behavior to the networks in figures 2-6. In the ResNet architecture, each skip module exhibits consistent reductions in manifold dimensions, radii and correlations similar to the changes in the other network architectures (figures 7,SI8).

Consistent across all the networks we studied, the increase in capacity is modest for most of the initial layers and improves considerably in the last stages (typically after the last convolution building block). This trend is even more pronounced in the deep ResNet architectures (figures SI1-SI2). This does not imply that previous stages are not important. Instead, it reflects the fact that capacity intimately depends on the incremental improvement of a number of factors including geometry and correlation structure.

Given the ubiquity of the changes in the manifold representations found here, we predict that similar patterns will be observed for sensory hierarchies in the brain. One issue of concern is trial-to-trial variability in neuronal responses. Our analysis assumes deterministic neural responses with sharp manifold boundaries, but it can be extended to the case of stochastic representations where manifolds are not perfectly separable. Alternatively, one can interpret the trial averaged neural responses as representing the spatial averaging of the responses of a group of stochastic neurons, with similar signal properties but weak noise correlations. To properly assess the properties of perceptual manifolds in the brain, responses of moderately large subsampled populations of neurons to numerous objects with multiple sources of physical variability is required. Such datasets are becoming technically feasible with advanced calcium imaging [26]. Recent work has also enabled quantitative comparisons to DCNNs from electrophysiological recordings from V4 and IT cortex in macaques [45].

One extension of our framework would relate capacity to generalization, the ability to correctly classify test points drawn from the manifolds but not part of the training set. While not addressed here, we expect that it will depend on similar geometric manifold measures, namely stages with reduced R_M and D_M will exhibit better generalization ability. Those geometric manifold measures can be related to optimal separating hyperplanes which are known to provide improved generalization performance in support vector machines.

The statistical measures introduced in this work (capacity, geometric manifold properties and center correlations) can also be used to guide the design and training of future deep networks. By extending concepts of efficient coding and object representations to higher stages of sensory processing, our theory can help elucidate some of the fundamental principles that underlie hierarchical sensory processing in the brain and in deep artificial neural networks.

Methods

Summary of manifold classification capacity Following the theory introduced in [16], manifolds are described by $D+1$ coordinates, one of them defines the location of the manifold center and the others the axes of the manifold variability. The set of points that define the manifold within its subspace of variability is formally designated as $\mathcal{S}$ which can represent a collection of finite number of data points or a smooth manifold (e.g., sphere or a curve). An ensemble of P manifolds is defined by assuming the center locations and the axes' orientations are random (focusing first on the case where all manifolds have the same *shape*). Classification capacity with a margin $\kappa \geq 0$ is defined as $\alpha_c(\kappa) = P_c/N$ where P_c is the maximum number of manifolds that can be linearly separated with margin κ using random binary labels. In mean field theory, capacity takes the form of

$$\alpha_c^{-1}(\kappa) = \langle F(\vec{T}) \rangle_{\vec{T}} \tag{1}$$

$$F(\vec{T}) = \min_{\vec{V}} \left\{ \left\| \vec{V} - \vec{T} \right\|^2 \mid \min_{\vec{S}} \left\{ \vec{V} \cdot \vec{S} \mid \vec{S} \in \mathcal{S} \right\} - \kappa \geq 0 \right\} \tag{2}$$

where $\langle \ldots \rangle_{\vec{T}}$ is an average over random $D+1$ dimensional vectors $\vec{T}$ whose components are i.i.d. normally distributed $T_i \sim \mathcal{N}(0,1)$. The components of the vector $\vec{V}$ represent the signed fields induced by the optimal separating vector $\mathbf{w}$ on the axes of a single manifold. The Gaussian vector $\vec{T}$ represents the contribution part of the variability in $\vec{V}$ due to *quenched* variability in the manifolds' orientations and labels. The inequality constraints on $\vec{V}$ in F ensures that the projections of all the points on $\mathbf{w}$ are greater or equal to the margin. Note that in the mean field theory, appropriate for the limit of large N and P, the representation size N does not appear. The analysis in this paper focuses on the case $\kappa = 0$ which describe the maximal number of separable manifolds per neuron.

Manifold anchor points The maximum margin solution vector can always be written as *a linear combination of a set of support vectors*. In the case of manifold separation, the solution vector can be decomposed into at most P representative vectors from each manifold, such that $\mathbf{w} = \sum_{\mu=1}^{P} \lambda_\mu y^\mu \tilde{\mathbf{x}}^\mu$, $\lambda_\mu \geq 0$ where $\tilde{\mathbf{x}}^\mu \in \mathrm{conv}\,(M^\mu)$ is a representative vector in the convex hull of the μ-th manifold, M^μ. These vectors play a key role in the theory, as they comprehensively determine the separating plane. We denote these representative points from each manifold as the *manifold anchor points* (details can be found in [16]).

Geometric properties of manifolds For a given manifold, its anchor point depends on the other manifolds in the ensemble. Variatrion in their location or orientation induce in general variation in the anchor point. In the mean field theory, this variation is given as a distribution of of anchor points $\tilde{S}$ induced by the Gaussian vectors $\vec{T}$ give rise to manifold anchor geometry, allowing us to define the object manifold's effective radius and effective dimension, i.e. the values relevant for manifold capacity. Note that the geometric properties are actually computed using $\delta\tilde{S} = \tilde{S} - S_0$, the projection of the anchor point $\tilde{\mathbf{x}}$ onto the D-dimensional subspace of each manifold, $\tilde{S}$, minus (orthogonal to) the manifold center, S_0. The manifold's effective radius is defined as R_{M}, defined by the mean squared length of these anchor points, $\widetilde{\delta S}(\vec{T})$, where

$$R_{\mathrm{M}}^2 = \left\langle \left\| \widetilde{\delta S}(\vec{T}) \right\|^2 \right\rangle_{\vec{T}} \tag{3}$$

and the effective dimension, D_{M}, is defined by the angular spread between $\delta\vec{T} = \vec{T} - T_0$ (T_0 is $\vec{T}$ in the direction of the center S_0) and the corresponding anchor point $\widetilde{\delta S}(\vec{T})$ in the manifold subspace,

$$D_{\mathrm{M}} = \left\langle \left(\delta\vec{T} \cdot \widehat{\delta S}(\vec{T}) \right)^2 \right\rangle_{\vec{T}} \tag{4}$$

where $\widehat{\delta S}$ is a unit vector in the direction of $\widetilde{\delta S}$ (for detailed derivation see Section 4-D of [16]).

Furthermore, the theory provides a precise connection between of manifold capacity, a measure of separability, and relevant manifold dimensionality and radius, denoted $D_M$and R_M, which can be concisely summarized as:

$$\alpha_c(\kappa) \approx \alpha_{Ball}\left(\kappa, R_M, D_M\right) \tag{5}$$

where $\alpha_{Ball}(\kappa, R, D)$ is the expression for the capacity of L_2 balls with radius R and dimension D, for an imposed margin κ ([46] and SI equation 4). For a general manifold we interpret the radius as maximal variation per dimension (in units of the manifold's center norm) while the dimension as the number of effective axes of variation, with the manifold's total extent given by $R_M\sqrt{D_M}$.

The results from [46] were derived for manifolds assuming the manifold centers are uncorrelated; in the following section we will demonstrate how to overcome this limitation.

Capacity of manifolds with low-rank centers correlation structure A theoretical analysis of classification capacity for manifolds with correlated centers is possible using the same tools as [16] and is provided in SI section 3.1. Denoting x^μ the center of mass of manifold $\mu = 1..P$, assuming the $P \times P$ dimensional correlation matrix between manifold centers $C_{\mu\nu} = \langle x^\mu x^\nu \rangle$ satisfies a "low-rank off-diagonal structure"

$$C = \Lambda + C_K \tag{6}$$

where Λ is diagonal and C_K is of rank K, i.e. can be written as $C_K = \sum_1^K c_k \vec{u}_k \vec{u}_k^T$. The theory predicts that for $K \ll P$ the capacity depends on the structure of the manifolds projected to the null-space of the common components, as defined by $\{\vec{u}_k\}_{k=1}^K$ (see SI section 3.1).

Recovering low-rank centers correlations structure In order to take into account the correlations between the centers of the manifolds that may exist in realistic data, before computing the effective radius and dimension, we first recover the "common components" of the centers by finding an orthonormal set $V \in \mathbb{R}^{N\times K}$ such that the centers projected to its null-space have approximately diagonal correlation structure. Then the entire manifolds are projected into the null-space of the common components. As the residual manifolds have uncorrelated centers, classification capacity is predicted from the theory for uncorrelated manifolds (equation 1). The validity of this prediction is demonstrated numerically for smooth manifolds in figure SI11. Furthermore, the manifolds geometric properties R_M, D_M from equation 3 and equation 4 are calculated from the residual manifolds using the procedure from [16]. Those are expected to approximate capacity using equation 5 when the dimension is substantial; the validity of this approximation for smooth manifolds is demonstrated numerically in figure SI12. The full procedure is described in SI section 2.1.

Inhomogeneous ensemble of manifolds The object manifolds considered above may each have a unique shape and size. For a mixture of heterogeneous manifolds [16], classification capacity for the ensemble of object manifolds is given by averaging the inverse of the object manifold capacity estimated from each manifold separately: $\alpha^{-1} = \left\langle \alpha_\mu^{-1} \right\rangle_\mu$. Reported capacity value α_c are calculated by evaluating the mean field estimate from individual manifolds and averaging their inverse over the entire set of P manifolds. Similarly, the displayed radius and dimensions are averages over the manifolds (using a regular averaging). An example of distribution of geometric metrics over the different manifolds is shown in figures SI3, SI4.

Measuring capacity numerically from samples Classification capacity can be measured numerically by performing linear classification of manifolds. Consider a population of N neurons which represents P objects through their collective responses to samples of those objects. Assuming the objects are linearly separable using the entire population of N neurons, we seek the typical sub-population size n where those P objects are no longer separable. For a given sub-population size n we first project the N dimensional response to the lower dimension n using random projections; using sub-sampling rather than random projections provide very similar results but breaks down for very sparse population responses (figure SI14). Second, we estimate the fraction of linearly separable dichotomies by randomly sampling binary labels for the object manifolds and checking if the sub-population data is linearly separable using those labels. Testing for linearly separability of manifold can be done using regular optimization procedures (i.e. using quadratic optimization), or using efficient algorithms developed specifically for the task of manifold classification [39]. As n increase the fraction of separable dichotomies goes from 0 to 1 and we numerically measure classification capacity as $\alpha_c = P/n_c$ where the fraction surpasses 50%; a binary search for the exact value n_c is used to find this transition. The full procedure is described in SI section 2.2.

Generating point-cloud and smooth manifolds The pixel-level representation of each point-cloud manifold is generated using samples from a single class from ImageNet data-set [35]. We have chosen $P = 50$ classes (the names and identifiers of the classes used are provided in SI section 4.1). The extent of the point-clouds can be varied despite the lack of generative model for the level of variability in ImageNet data-set, by utilizing the scores assigned to each images in a corresponding node of the last layer, essentially indicating how template-like an image is. Thus we consider the two types of manifolds: (1) "full class" manifolds, where all exemplars from the given class are used, or (2) "top 10%" manifolds, where just the 10% of the exemplars with large confidence in class-membership, as measured by the value (score) in the softmax layer, at the node corresponding to the ground-truth class of the exemplar image in ImageNet (pretrained AlexNet model from PyTorch implementation was used for the score throughout). The difference between randomly-samples exemplars (which occupy the "full class" manifolds) and high-confidence exemplars (which occupy the "top 10%" manifolds) is illustrated in SI section 4.3.

The pixel-level representation of each smooth manifold is generated from a single ImageNet image. Only images with an object bounding-box annotation [35] were used; at the base image the object occupied the middle 75% of a 64×64 image. Manifolds samples are then generated by warping the base image using an affine transformation with either 1 or 2 degrees of freedom. Here we have used 1-d manifolds with horizontal or vertical translation, horizontal or vertical shear; and 2-d manifolds with horizontal and vertical translation or horizontal and vertical shear. The amount of variation in the manifold is controlled for by limiting the maximal displacement of the object corners, thus allowing for generating manifolds with different amount of variability. Manifolds with maximal displacement of up to 16 pixels where used; the resultant amount of variability is quantified by the value of "input variability" (shown in figures 4,SI2,SI5), measured using SI equation 3 at the pixel layer. Here $P = 128$ base images were used to generate 1-d manifolds and $P = 64$ to generate 2-d manifolds, both without using images of the same ImageNet class (thumbnails of the base images used for both 1-d and 2-d manifolds are provided in SI section 4.2). The number of samples for each of those manifolds is chosen such that capacity would approximately saturate, thus allowing to extrapolate to the case of infinite number of samples.

For both point-cloud and smooth manifolds, representations for all the layers along the different deep hierarchies considered are generated from the pixel-level representation by propagating the images along the hierarchies. Both PyTorch [47] and MatConvNet [48] implementations of the DCNNs were used. At each layer a fixed population of $N = 4096$ neurons was randomly sampled once and used in the analysis, both when numerically calculating capacity and when measuring capacity and manifold properties using the mean field theory. The first layer of each network is defined as the pixel layer; the last is the feature layer (i.e. before a fully-connected operation and a soft-max non-linearity). Throughout the analysis convolutional and fully-connected layers are analyzed after applying local ReLU non-linearity (unless referring explicitly to the isolated operations as in figures 7,SI8).

References

[1] James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–34, feb 2012.

[2] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, may 2015.

[3] Charles F. Cadieu, Ha Hong, Daniel L. K. Yamins, Nicolas Pinto, Diego Ardila, Ethan A., Solomon, Najib J. Majaj, James J. DiCarlo, Ethan a Solomon, Najib J. Majaj, and James J. DiCarlo. Deep Neural Networks Rival the Representation of Primate IT Cortex for Core Visual Object Recognition. *PLoS computational biology*, 10(12):e1003963, dec 2014.

[4] Nikolaus Kriegeskorte, Marieke Mur, Douglas a Ruff, Roozbeh Kiani, Jerzy Bodurka, Hossein Esteky, Keiji Tanaka, and Peter a Bandettini. Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron*, 60(6):1126–41, dec 2008.

[5] Daniel L K Yamins, Ha Hong, Charles F Cadieu, Ethan a Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences of the United States of America*, 111(23):8619–24, jun 2014.

[6] Nikolaus Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1:417–446, 2015.

[7] Seyed-Mahdi Khaligh-Razavi and Nikolaus Kriegeskorte. Deep Supervised, but Not Unsupervised, Models May Explain IT Cortical Representation. *PLoS Computational Biology*, 10(11):e1003915, nov 2014.

[8] Saeed Reza Kheradpisheh, Masoud Ghodrati, Mohammad Ganjtabesh, and Timothée Masquelier. Deep Networks Can Resemble Human Feed-forward Vision in Invariant Object Recognition. *Scientific Reports*, 6:1–24, 2016.

[9] Haiguang Wen, Junxing Shi, Wei Chen, and Zhongming Liu. Deep Residual Network Predicts Cortical Representation and Organization of Visual Features for Rapid Categorization. *Scientific Reports*, 8(1):1–17, 2018.

[10] David GT Barrett, Ari S Morcos, and Jakob H Macke. Analyzing biological and artificial neural networks: challenges with opportunities for synergy? *Current opinion in neurobiology*, 55:55–64, 2019.

[11] James J DiCarlo and David D Cox. Untangling invariant object recognition. *Trends in cognitive sciences*, 11(8):333–41, aug 2007.

[12] M Ranzato, Fu Jie Huang, Y L Boureau, and Y Lecun. Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[13] J Goodfellow, Quoc V Le, Andrew M Saxe, Honglak Lee, and Andrew Y Ng. Measuring invariance in deep networks. *Advances in Neural Information Processing Systems (NIPS)*, 22:646–654, 2009.

[14] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[15] Ben Poole, Subhaneil Lahiri, Maithra Raghu, Jascha Sohl-Dickstein, and Surya Ganguli. Exponential expressivity in deep neural networks through transient chaos. In *Advances in neural information processing systems*, pages 3360–3368, 2016.

[16] SueYeon Chung, Daniel D. Lee, and Haim Sompolinsky. Classification and Geometry of General Perceptual Manifolds. *Physical Review X*, 8(3):031003, jul 2018.

[17] Elizabeth J. Gardner. The space of interactions in neural network models. *Journal of Physics A: Mathematical and General*, 21(1):257–270, 1988.

[18] Elizabeth J. Gardner and B Derrida. Three unfinished works on the optimal storage capacity of networks. *Journal of Physics A: Mathematical and General*, 22(12):1983–1994, 1989.

[19] Thomas M. Cover. Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. *IEEE Transactions on Electronic Computers*, EC-14(3):326–334, jun 1965.

[20] Bruno B Averbeck, Peter E Latham, and Alexandre Pouget. Neural correlations, population coding and computation. *Nature reviews neuroscience*, 7(5):358, 2006.

[21] Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, pages 6076–6085, 2017.

[22] Ari Morcos, Maithra Raghu, and Samy Bengio. Insights on representational similarity in neural networks with canonical correlation. In *Advances in Neural Information Processing Systems*, pages 5727–5736, 2018.

[23] Roozbeh Kiani, Hossein Esteky, Koorosh Mirpour, and Keiji Tanaka. Object category structure in response patterns of neuronal population in monkey inferior temporal cortex. *Journal of . . .*, pages 4296–4309, 2007.

[24] Olivier J Hénaff, Robbe LT Goris, and Eero P Simoncelli. Perceptual straightening of natural videos. *Nature neuroscience*, page 1, 2019.

[25] Peiran Gao and Surya Ganguli. On simplicity and complexity in the brave new world of large-scale neuroscience. *Current opinion in neurobiology*, 32:148–155, 2015.

[26] Carsen Stringer, Marius Pachitariu, Nicholas Steinmetz, Matteo Carandini, and Kenneth D Harris. High-dimensional geometry of population responses in visual cortex. *bioRxiv*, page 374090, 2018.

[27] Ashok Litwin-Kumar, Kameron Decker Harris, Richard Axel, Haim Sompolinsky, and LF Abbott. Optimal degrees of synaptic connectivity. *Neuron*, 93(5):1153–1164, 2017.

[28] Matthew S Farrell, Stefano Recanatesi, Guillaume Lajoie, and Eric Shea-Brown. Dynamic compression and expansion in a classifying recurrent network. *bioRxiv*, page 564476, 2019.

[29] David Sussillo and L F Abbott. Generating coherent patterns of activity from chaotic neural networks. *Neuron*, 63(4):544–57, aug 2009.

[30] Amr Bakry, Mohamed Elhoseiny, Tarek El-Gaaly, and Ahmed Elgammal. Digging Deep into the Layers of CNNs: In Search of How CNNs Achieve View Invariance. *arXiv preprint*, pages 1–20, 2015.

[31] N Alex Cayco-Gajic and R Angus Silver. Re-evaluating circuit mechanisms underlying pattern separation. *Neuron*, 101(4):584–602, 2019.

[32] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *Proceedings of the International Joint Conference on Neural Networks*, 2016-Octob:2560–2567, nov 2014.

[33] Matthew D Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. *arXiv preprint arXiv:1311.2901*, nov 2013.

[34] Santiago A Cadena, Marissa A Weis, Leon A Gatys, Matthias Bethge, and Alexander S Ecker. Diverse feature visualizations reveal invariances in early layers of deep neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 217–232, 2018.

[35] Jia Deng, Wei Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2–9, 2009.

[36] Alex Krizhevsky, I Sutskever, and G Hinton. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information . . .*, pages 1–9, 2012.

[37] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556*, sep 2014.

[38] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[39] SueYeon Chung, Uri Cohen, Haim Sompolinsky, and Daniel D. Lee. Learning Data Manifolds with a Cutting Plane Method. *Neural Computation*, 30(10):2593–2615, oct 2018.

[40] William E Vinje and Jack L Gallant. Sparse coding and decorrelation in primary visual cortex during natural vision. *Science*, 287(5456):1273–1276, 2000.

[41] Xaq Pitkow and Markus Meister. Decorrelation and efficient coding by retinal ganglion cells. *Nature neuroscience*, 15(4):628, 2012.

[42] Silvia Bernardi, Marcus K Benna, Mattia Rigotti, Jerome Munuera, Stefano Fusi, and Daniel Salzman. The geometry of abstraction in hippocampus and prefrontal cortex. *bioRxiv*, page 408633, 2018.

[43] Ravid Shwartz-Ziv and Naftali Tishby. Opening the black box of deep neural networks via information. *arXiv preprint arXiv:1703.00810*, 2017.

[44] Mattia Rigotti, Omri Barak, Melissa R Warden, Xiao-Jing Wang, Nathaniel D Daw, Earl K Miller, and Stefano Fusi. The importance of mixed selectivity in complex cognitive tasks. *Nature*, 497(7451):585, 2013.

[45] Martin Schrimpf, Jonas Kubilius, Ha Hong, Najib J Majaj, Rishi Rajalingham, Elias B Issa, Kohitij Kar, Pouya Bashivan, Jonathan Prescott-Roy, Kailyn Schmidt, et al. Brain-score: which artificial neural network for object recognition is most brain-like? *BioRxiv*, page 407007, 2018.

[46] SueYeon Chung, Daniel D. Lee, and Haim Sompolinsky. Linear readout of object manifolds. *Physical Review E*, 93(6):060301, 2016.

[47] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[48] Andrea Vedaldi and Karel Lenc. Matconvnet - convolutional neural networks for MATLAB. *arXiv preprint arXiv:1412.4564*, abs/1412.4564, 2014.

Acknowledgement

We would like to thank Jim DiCarlo, Yair Weiss, Aharon Azulay, Joel Dapello for valuable discussions. The work is partially supported by the Gatsby Charitable Foundation, the Swartz Foundation, the National Institutes of Health (Grant No. 1U19NS104653) and the MAFAT Center for Deep Learning. D.D.L. acknowledges support from the NSF, NIH, DOT, ARL, AFOSR, ONR, and DARPA.

Author Contribution

All authors contributed to the development of the project. UC and SC performed simulations. All participated in the analysis and interpretation of the data. All authors wrote the paper.

Additional information

Competing financial interests The authors declare no competing financial interests.