Supplementary Material for Characterizing emergent representations in a space of candidate learning rules for deep networks

Yinan Cao[*]
Department of Experimental Psychology
University of Oxford
Oxford, UK
y.cao@uke.de

Christopher Summerfield
Department of Experimental Psychology
University of Oxford
Oxford, UK
christopher.summerfield@psy.ox.ac.uk

Andrew Saxe
Department of Experimental Psychology
University of Oxford
Oxford, UK
CIFAR Azrieli Global Scholar, CIFAR
andrew.saxe@psy.ox.ac.uk

1 Supplemental methods

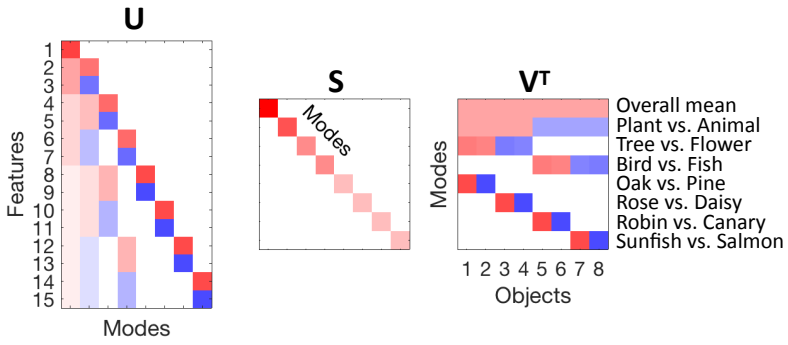
1.1 Extracting input-output mapping components for each hierarchical level

We apply singular value decomposition (SVD) to the dataset's input-output correlation matrix to extract the component of the input-output mapping for different hierarchical levels. Suppose the desired (target) output matrix is given by $\mathbf{Y}$ as shown in the main paper Fig. 1b, and input matrix is $\mathbf{X}$ where examples are placed in columns. In $\mathbf{X}$, each object's perceptual representation $\mathbf{x}^{\mu}$ (a column vector, where $\mu = 1...P$ indexing objects) is encoded by a one-hot input vector (Kronecker delta $\delta_{\mu i}$). Thus, the input-output correlation matrix is $\mathbf{\Sigma}^{yx} = \mathbf{Y}\mathbf{X}^T$. We use SVD on $\mathbf{\Sigma}^{yx}$, i.e., $\mathbf{\Sigma}^{yx} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, which results in three key elements fully characterizing the input-output mapping to be learned (visualized in Supp. Fig. 1). For the case of hierarchically structured data from a binary tree, the SVD structure conforms to hierarchical distinctions in the dataset [4]. The first element is $\mathbf{U}$, a feature-synthesizer matrix in which each column (a particular semantic dimension or 'mode') contains positive (negative) values for semantic features that objects categorized along this mode do (do not) possess. The second element is $\mathbf{S}$, a singular value matrix that has nonzero values only on the diagonal, and these values are arranged in a descending order. And the last element is $\mathbf{V}^T$, whose rows are object-analyzer vectors, whereby a binary code is assigned to classify objects according to each semantic mode (e.g., the 2nd row of $\mathbf{V}^T$ indicates the first 4 objects are plants, while the last 4 objects are animals, see Fig. 1b in the main paper).

To compute the strength of a network's input-output mapping for these hierarchical distinctions (Fig. 2 of the main paper), we explicitly use the two orthogonal matrices, $\mathbf{U}$ and $\mathbf{V}$, to compute the "effective" singular values as $\hat{\mathbf{S}} = \mathbf{U}^T\hat{\mathbf{y}}\mathbf{x}^T\mathbf{V}$. The diagonal elements of $\hat{\mathbf{S}}$ then encode the strength of the mapping at different hierarchy levels. The off-diagonal elements, representing coupling between different hierarchy levels, are not necessarily zero in general; however, under the assumptions that learning is sufficiently gradual (small learning rate), and the initial off-diagonal elements in $\hat{\mathbf{S}}$ are zero, then the dynamics of gradient descent will remain decoupled. In what follows, we show through

[*]This author is now affiliated to University Medical Center Hamburg-Eppendorf, Hamburg, Germany.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.



Supplementary Figure 1: SVD of input-output correlation matrix $\mathbf{\Sigma}^{yx}$ reveals semantic distinctions (modes) that mirror the hierarchical taxonomy.

simulations that modes in both layers remain approximately decoupled more broadly in our space of learning rules throughout the course of learning under this assumption of decoupled initial states.

For standard error-corrective learning based on gradient descent, the weight updates for the first and second layer can be computed as

$$\frac{1}{\lambda}\Delta\mathbf{W}_1 = \mathbf{W}_2^T(\mathbf{y}-\hat{\mathbf{y}})\mathbf{x}^T \quad (1)$$

$$\frac{1}{\lambda}\Delta\mathbf{W}_2 = (\mathbf{y}-\hat{\mathbf{y}})\mathbf{x}^T\mathbf{W}_1^T. \quad (2)$$

where λ is a small learning rate ($\lambda \ll 1$), and $\hat{\mathbf{y}} = \mathbf{W}_2\mathbf{W}_1\mathbf{x}$ is the matrix of predicted outputs. Following [4], we replace $\mathbf{W}_2$ and $\mathbf{W}_1$ by $\overline{\mathbf{W}}_2$ and $\overline{\mathbf{W}}_1$, respectively, such that:

$$\mathbf{W}_1 = \mathbf{R}\overline{\mathbf{W}}_1\mathbf{V}^T \quad (3)$$

$$\mathbf{W}_2 = \mathbf{U}\overline{\mathbf{W}}_2\mathbf{R}^T \quad (4)$$

where $\mathbf{R}$ is an arbitrary orthogonal matrix i.e., $\mathbf{R}^T\mathbf{R} = \mathbf{I}$. Here, simulations shown in Supp. Fig. 2 reveal that the matrices $\overline{\mathbf{W}}_2(t)$ and $\overline{\mathbf{W}}_1(t)$ remain approximately diagonal under the learning dynamics, such that modes are approximately decoupled throughout learning when starting from small random weights. As a result, the learning dynamics of the overall input-output map of a network is

$$\mathbf{W}_2(t)\mathbf{W}_1(t) = \mathbf{U}\overline{\mathbf{W}}_2(t)\overline{\mathbf{W}}_1(t)\mathbf{V}^T, \quad (5)$$

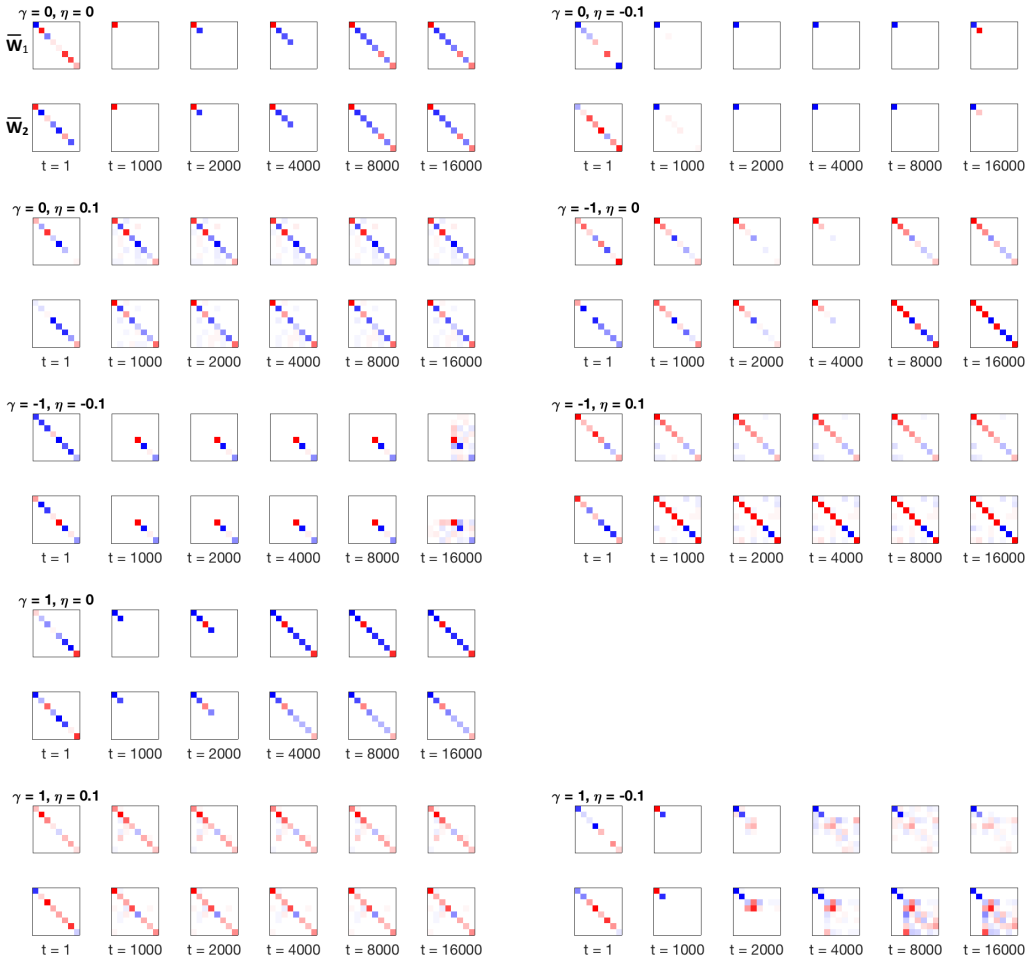
$$(6)$$

and the overall strength of semantic modes can be extracted from the diagonal of the matrix $\hat{\mathbf{S}}$,

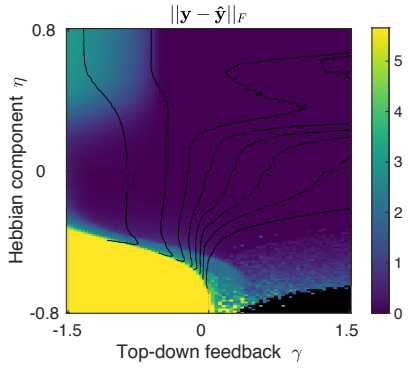
$$\hat{\mathbf{S}}(t) = \overline{\mathbf{W}}_2(t)\overline{\mathbf{W}}_1(t) = \mathbf{U}^T\mathbf{W}_2(t)\mathbf{W}_1(t)\mathbf{V}. \quad (7)$$

1.2 Other forms of input structure

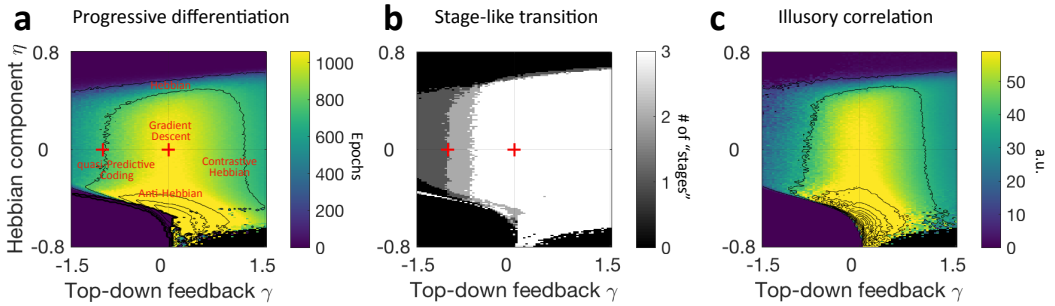
In the main paper, we assume the input representations $\mathbf{x}$ to be one-hot vectors (and we also assume that the number of input neurons is the number of input patterns, $N_i = P$). The task is to link each object's perceptual representation ($\mathbf{x}^\mu$, where $\mu = 1...P$), encoded by a one-hot input vector (Kronecker delta $\delta_{\mu i}$), to its set of associated binary properties. That is, simple input classes must be mapped to semantically rich output properties. We interpret the one-hot input vector for each object as a sparse orthogonal code for object identity such as those might exist at the top of the cortical



Supplementary Figure 2: $\overline{\mathbf{W}}_1(t)$ and $\overline{\mathbf{W}}_2(t)$ contain decoupled modes over the course of learning for most learning algorithms defined on the 2D space (except the 2 algorithms at the bottom ($\eta = -0.1$, $\gamma = \pm 1$) where simulations do not converge satisfactorily, also see Supp. Fig. 3).



Supplementary Figure 3: Frobenius norm of training error at the end of training. The black area (lower-right corner) = nan where the optimization completely failed. Note that there are several areas (e.g., upper-left corner and some points within anti-Hebbian learning) where the error did decrease, but not converge to near-zero within 10,000 training epochs.



Supplementary Figure 4: Learning dynamics under different learning rules in networks with an arbitrary orthonormal multi-dimensional input matrix.

visual hierarchy (or in a traditional vision deep network's classification layer). Our setting is in some ways a mirror of standard deep network approaches modeling the visual pathway (in primates in particular), which typically have complexly structured input features that are disentangled into one-hot classification outputs. Also, these whitened input representations are widely used as a result of pre-processing steps, i.e., inputs undergo a preprocessing step in which the stronger inputs are strengthened and the weaker inputs are weakened. There has been evidence that the dentate gyrus, through which inputs pass *en route* from the neocortex to the hippocampus, performs a computational function that resembles pattern separation, making the sensory inputs sparser and less overlapping [3, 2].

However, it seems critical to demonstrate that our framework is robust against a modification of this assumption about input structure. Here, we show that the conclusions presented in the main paper remain unchanged even if we relax the assumption of one-hot vectors (which are similar to grandmother-cell neurons: each object is represented by a dedicated single neuron). We use a non-localist, distributed representation in which the vectors x^μ form an orthonormal basis for the input-layer activity patterns. The differences in learning dynamics across different learning rules within the 2D space are robust against the shift from localist assumption to the current distributed assumption (Supp. Fig. 4). Thus, the localist assumption is not necessary, and is adopted only for simplicity and interpretability.

1.3 Choice of metric for progressive differentiation

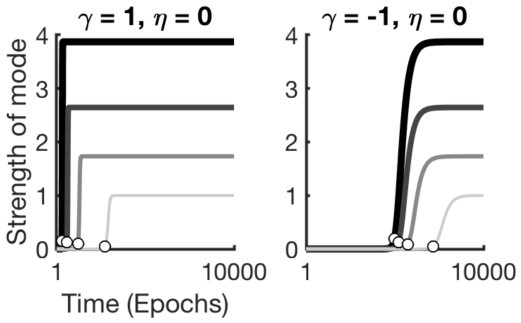
For progressive differentiation, we had investigated several alternatives (such as a normalized time lag divided by the overall learning speed) before settling on our metric (mean time lag in paper Fig. 2a). Consider Supp. Fig. 5. These settings both show qualitatively similar progressive differentiation, which is captured by our mean lag metric. Further dividing by the time to learn the smallest mode would make the $\gamma < 0$ case appear much less stage-like, because the overall learning rate is faster for $\gamma > 0$ but slower for $\gamma < 0$ (notably it is not simply determined by the absolute value of γ). We found this normalization would make the 2D-map visualization unintuitive.

1.4 Network initialized from random weights of large norm

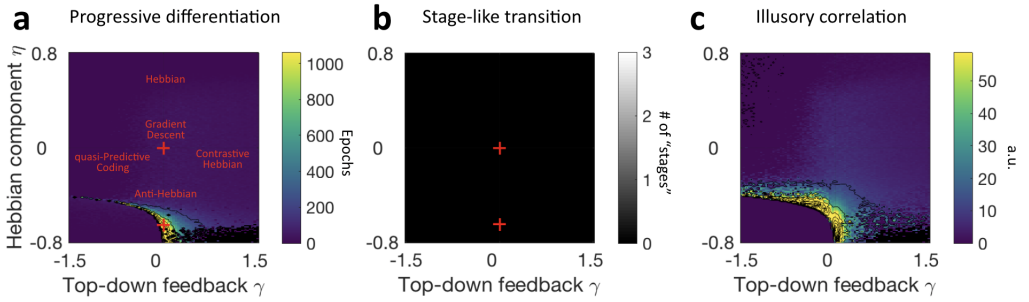
The results presented in the main text focus on the setting of "tabula rasa" learning where initial weights are very small. Here we note that initialization can dramatically influence learning dynamics. Gradient descent learning initialized with large random weights no longer exhibits progressive differentiation, stage-like transitions, or illusory correlations (Supp. Fig. 6). Strong anti-Hebbian learning ($\eta < 0$) still exhibits the differentiation between learning speeds of hierarchical categories and semantic illusions. However, no stage-like transitions are found in any of the learning rule initialized with large random weights.

1.5 Quasi-Predictive Coding inhibits hidden-layer activity via negative top-down feedback

Here we show that the quasi-predictive coding algorithm, when trained as an autoencoder, exactly cancels feedforward activity with top-down inhibition under the assumption of minimum norm weight



Supplementary Figure 5: Learning trajectories for two algorithms: $\gamma = 1$ (contrastive Hebbian) vs. $\gamma = -1$ (quasi-predictive coding). Notably, overall learning speed is not simply determined by the absolute value of γ): the overall learning rate is faster for $\gamma > 0$ but slower for $\gamma < 0$.



Supplementary Figure 6: Learning dynamics under different learning rules implemented in networks initialized with large random weights sampled from Gaussian distribution (SD = 0.1 in contrast to 10^{-10} as shown in Fig. 2 of the main paper).

configurations that achieve zero training error. For an autoencoder, if we set $\mathbf{y} = \mathbf{x}$, then when the network achieves zero training error at convergence, we get $\hat{\mathbf{y}} = \mathbf{W}_2\mathbf{W}_1\mathbf{x} \approx \mathbf{x}$, i.e., $\mathbf{W}_2\mathbf{W}_1 \approx \mathbf{I}$, and $\mathbf{W}_2^T \approx \mathbf{W}_1$ when the mapping is minimum norm. Thus, the hidden-unit activity vector after the backwards sweep update is given by $\mathbf{h} = \mathbf{W}_1\mathbf{x} + \gamma\mathbf{W}_2^T\hat{\mathbf{y}} \approx \mathbf{W}_1\mathbf{x} + \gamma\mathbf{W}_1\mathbf{x}$, and this shrinks to zero when $\gamma = -1$.

1.6 Illusory correlation

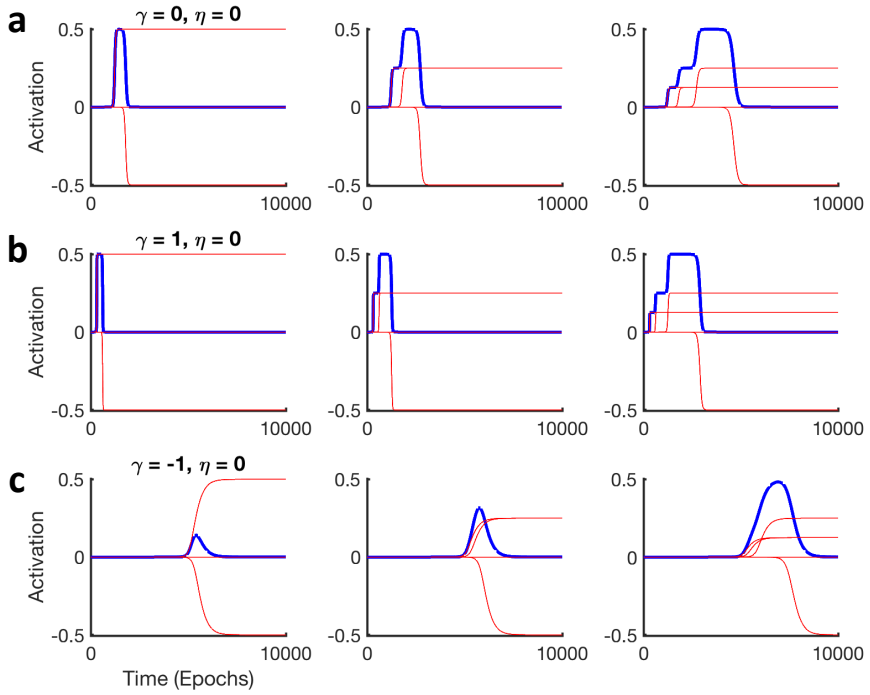
The predicted value of a feature m for item i is a sum of contributions from all semantic modes (red traces in Supp. Fig. 7). That is, the predicted value of a feature m for item i is $\Sigma_\alpha a^\alpha(t) u_m^\alpha v_i^\alpha$, where α indexes different modes, u_m^α is the m-th row/α-th column of matrix $\mathbf{U}$, and v_i^α is the α-th row/i-th column of matrix $\mathbf{V}^T$.

To examine illusory correlations, we consider the fate of a feature which should be off, but which has a strong positive loading onto the higher dimensions, but a negative loading on the lowest dimension, producing a U-shaped learning trajectory (see Supp. Fig. 7).

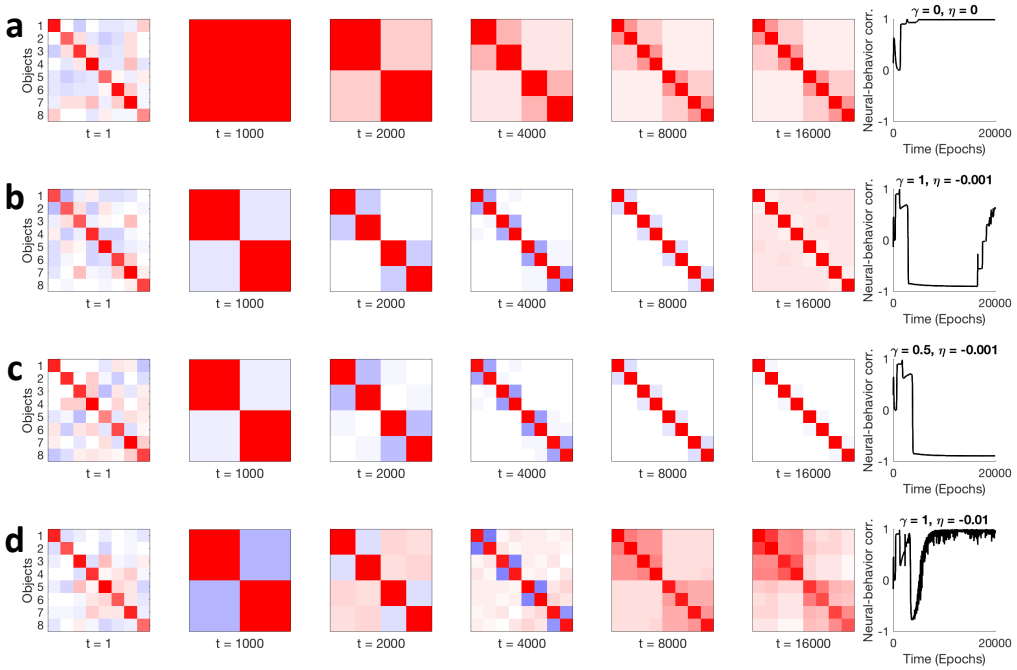
Different learning rules predict distinct extents to which such illusory correlations will manifest, in terms of not only different timing and duration but also different magnitudes of errors for different levels of hierarchical dimensions. For instance, quasi-predictive coding ($\gamma = -1$, $\eta = 0$; Supp. Fig. 7) predicts diminished illusion magnitudes for broader hierarchical dimensions (also see Fig. 2d in the main paper, quasi-predictive coding learns all but the finest hierarchical level at a similar time).

1.7 Neural representations and visualization

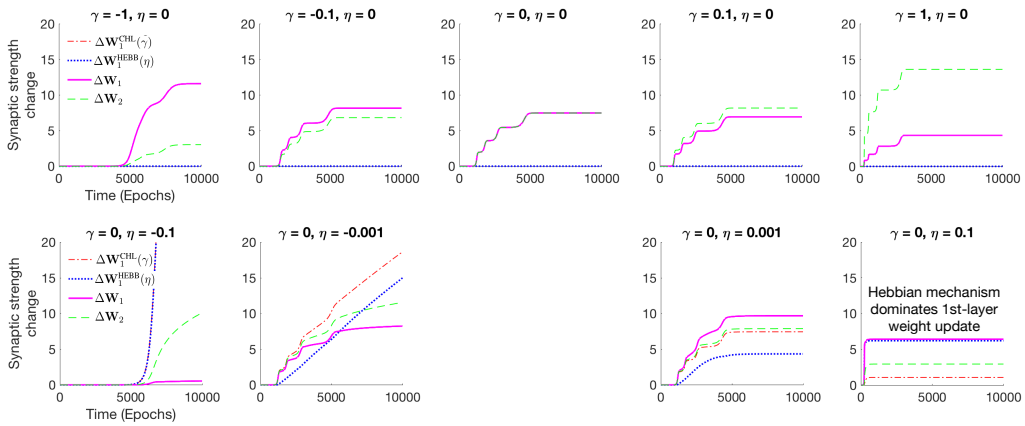
Comparisons between representations in deep networks and the brain have frequently made use of representational similarity analysis (RSA) [1], which considers two representations to be equivalent



Supplementary Figure 7: Predicted value (blue) of feature *can move* (left), *has petals* (middle), or *has needle-like leaves* (right), for an arbitrary item *oak* during learning. Red traces represent the contributions from individual semantic modes. (**a**) Gradient descent. (**b**) Contrastive Hebbian learning. (**c**) Quasi-predictive coding.



Supplementary Figure 8: Time-dependent neural representational similarity matrix (RSM; first 6 panels of each row) and neural-behavior correlation over training time (last panel of each row). (**a**) Gradient descent ($\gamma = \eta = 0$); (**b**) A learning algorithm with top-down feedback and a weak anti-Hebbian mechanism, corresponding to Figure 5b in the main paper. (**c**) and (**d**) are two variants to (**b**), albeit still within the fourth quadrant of the 2D space of learning rules, demonstrating that the anticorrelation is a common emergent feature for learning rules with positive γ and negative η.



Supplementary Figure 9: Cumulative sum over training time of the norm of connection weights update. $\Delta \mathbf{W}_1^{\text{CHL}}(\gamma)$ is the CHL update in the first layer, $\Delta \mathbf{W}_1^{\text{HEBB}}(\eta)$ is the correlational Hebbian update in the first layer, $\Delta \mathbf{W}_1$ is the total weight update in the first layer, and $\Delta \mathbf{W}_2$ is the total weight update in the second layer.

when the pairwise distances between neural representations for different inputs are identical. We therefore compute the neural representational similarity matrix (RSM) $\mathbf{\Sigma}_{ij}^h = \mathbf{h}_i^T \mathbf{h}_j$, where $\mathbf{h}_i$ is the hidden representation of item i.

We fit non-metric MDS models to these distances using **mdscale** in MATLAB, with **stress1** criterion and 10 random starting points. For the time-dependent MDS analysis, we stack the hidden-unit activity matrices generated at different training epochs into a big activity matrix, and compute the column-wise Euclidean distances of this big matrix of N_h rows by $N_p \times N_t$ columns (N_h is the number of hidden units, N_p the number of objects, and N_t the total number of epochs). We finally apply MDS to the big distance matrix to ensure the distances among objects and the evolution of these distances are estimated within the same multi-dimensional space.

In Supp. Fig. 8, we visualize the time-dependent neural representations $\mathbf{\Sigma}_{ij}^h$ for learning algorithms with $\gamma > 0$ and $\eta < 0$. We also plot the correlation between hidden layer representations and the desired behavioral correlations $\mathbf{\Sigma}_{ij}^y = \mathbf{y}_i^T \mathbf{y}_j$.

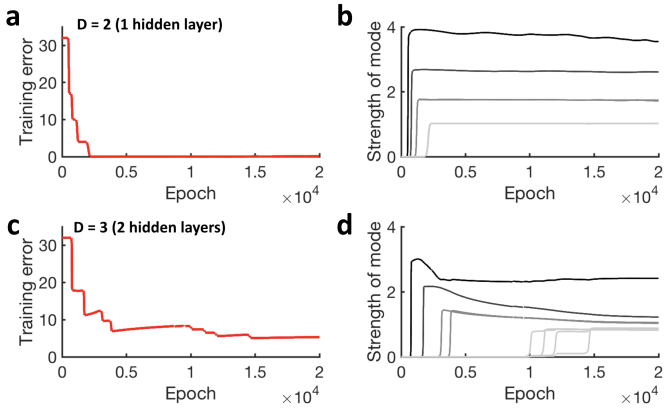
1.8 Synaptic weight updates across layers

In the model structure described in the paper, weights are updated with a learning rule defined by two parameters, γ and η, which govern the nature of a contrastive Hebbian update $\Delta \mathbf{W}_i^{\text{CHL}}(\gamma), i = 1, 2$ and a standard Hebbian update $\Delta \mathbf{W}_1^{\text{HEBB}}(\eta)$, respectively, such that the total update is $\Delta \mathbf{W}_1 = \Delta \mathbf{W}_1^{\text{CHL}}(\gamma) + \Delta \mathbf{W}_1^{\text{HEBB}}(\eta)$, and $\Delta \mathbf{W}_2 = \Delta \mathbf{W}_2^{\text{CHL}}(\gamma)$. In other words, the connection matrix $\mathbf{W}_1$ in the first layer is updated at each training time by two components, one from the CHL, and one from the Hebbian term, whereas the connection matrix $\mathbf{W}_2$ in the second layer is by its nature controlled by the CHL.

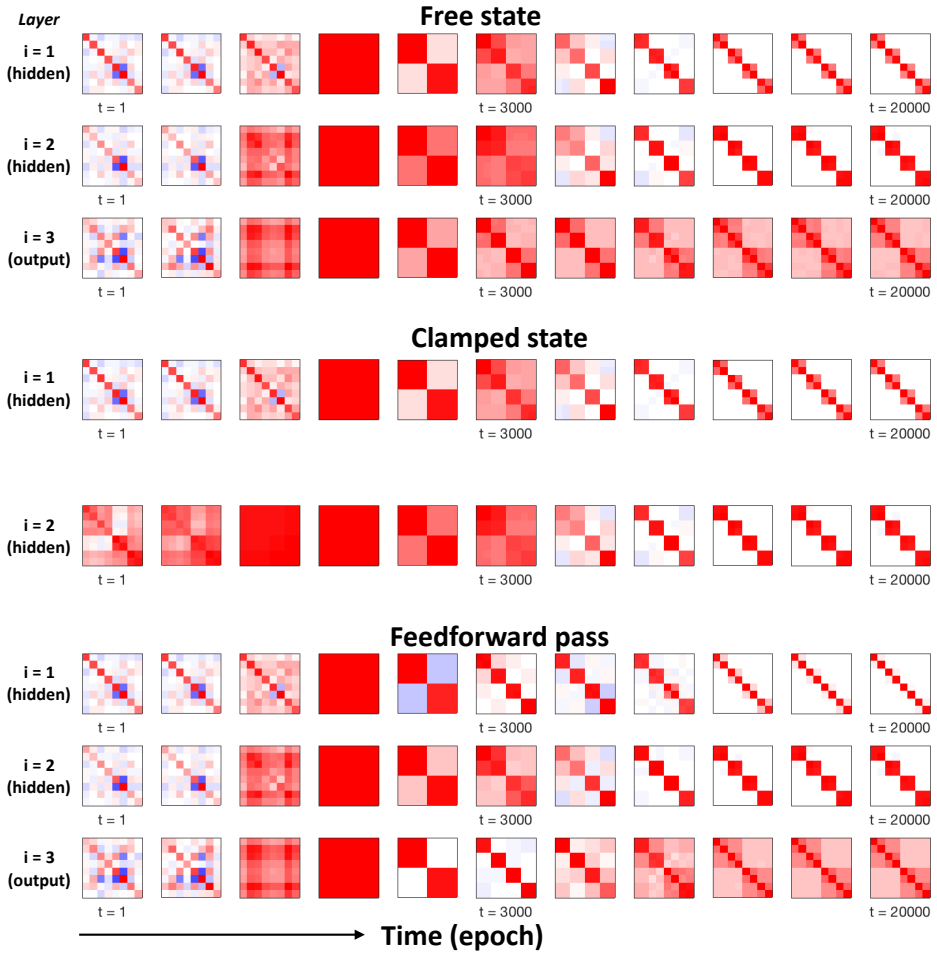
Here, we show in detail the competing contributions from CHL versus Hebbian term to synaptic weight update in the first layer (Supp. Fig. 9). Strong Hebbian algorithms yield first layer weights that are dominated by unsupervised Hebbian updates.

1.9 Generalization to arbitrarily deep linear networks

Saxe et al. [4] showed that for deep linear networks trained with gradient descent, the real difference is between 0 vs. 1 hidden layer. Here, we examine the influence of depth on our multi-layer model with top-down feedbacks and correlational Hebbian components. Consider a multi-layer network with $D + 1$ layers of neurons (i.e., D layers of synaptic weights), with 0 being the input layer and D the output layer. The activities of the i-th layer of neurons are denoted by the vector $\mathbf{h}_i$ (i = 1 to D; $\mathbf{h}_0 = \mathbf{x}$). Synaptic weights mapping the activities of layer $(i-1)$ to those of layer i are denoted by $\mathbf{W}_i$. In order to learn, weights are updated with a learning rule defined by two parameters, γ and η,



Supplementary Figure 10: Training error (squared Frobenius norm of error) and mode strengths as a function of network depth. Deeper networks train slower, but preserves similar hallmarks of learning dynamics such as stage-like transition and progressive differentiation for CHL combined with a weak Hebbian component ($\gamma = 0.05$, $\eta = 0.001$). Simulations are based on learning rate $\lambda = 0.01$; 32 hidden units within each hidden layer.



Supplementary Figure 11: Time-varying RSM of the neural activities at each layer of neurons for a deep network with 2 hidden layers.

which govern the nature of a contrastive Hebbian update $\Delta \mathbf{W}_i^{\text{CHL}}(\gamma)$ (i = 1 to D), and a correlational Hebbian update $\Delta \mathbf{W}_i^{\text{HEBB}}(\eta)$ (i = 1 to $D-1$)[2], respectively, such that the total weight update is the sum of the CHL contribution and the Hebbian contribution: $\Delta \mathbf{W}_i = \Delta \mathbf{W}_i^{\text{CHL}}(\gamma) + \Delta \mathbf{W}_i^{\text{HEBB}}(\eta)$ (i = 1 to $D-1$), and $\Delta \mathbf{W}_D = \mathbf{W}_D^{\text{CHL}}(\gamma)$. Following a standard CHL [6] in a deep linear network, the dynamic equation of the activity at layer i is:

$$\frac{d\mathbf{h}_i}{dt} = -\mathbf{h}_i + \mathbf{W}_i \mathbf{h}_{i-1} + \gamma \mathbf{W}_{i+1}^T \mathbf{h}_{i+1} \tag{8}$$

In general, we simulate the dynamic systems for coupled layers $i = 1$ to D using forward Euler method (time-step $dt = 0.05$ ms for a 3-layer network, and $dt = 0.5$ ms for a 4-layer network, total simulation time = 100 time steps)[3].

CHL is implemented via a contrast between two states of the network, one in which the activity of the output neurons are kept free, and one in which the output neurons are clamped to the desired target values $\mathbf{y} \in R^{N_o}$. In the 'free' (i.e., input-driven) state, the hidden-layer activity dynamics $\mathbf{h}_i^f$ after recurrence is given by

$$\frac{d\mathbf{h}_i^f}{dt} = \begin{cases} -\mathbf{h}_i^f + \mathbf{W}_i \mathbf{h}_{i-1}^f + \gamma \mathbf{W}_{i+1}^T \mathbf{h}_{i+1}^f & \text{if } i = 1 \text{ to } D-1 \\ -\mathbf{h}_i^f + \mathbf{W}_i \mathbf{h}_{i-1}^f & \text{if } i = D \end{cases} \tag{9}$$

In the 'clamped' (i.e., target-driven) state, hidden layer activity dynamics is given by

$$\frac{d\mathbf{h}_i^c}{dt} = \begin{cases} -\mathbf{h}_i^c + \mathbf{W}_i \mathbf{h}_{i-1}^c + \gamma \mathbf{W}_{i+1}^T \mathbf{h}_{i+1}^c & \text{if } i = 1 \text{ to } D-2 \\ -\mathbf{h}_i^c + \mathbf{W}_i \mathbf{h}_{i-1}^c + \gamma \mathbf{W}_{i+1}^T \mathbf{y} & \text{if } i = D-1 \end{cases} \tag{10}$$

The output layer in the clamped state is always clamped at the desired value $\mathbf{h}_D^c = \mathbf{y}$. The overall weight updates of the CHL contribution could be computed as the contrast between the clamped and free states:

$$\frac{1}{\lambda} \Delta \mathbf{W}_i^{\text{CHL}}(\gamma) = \gamma^{i-D} [\mathbf{h}_i^c (\mathbf{h}_{i-1}^c)^T - \mathbf{h}_i^f (\mathbf{h}_{i-1}^f)^T], i = 1 \text{ to } D \tag{11}$$

where λ is the learning rate.

In addition, the correlational Hebbian update term $\Delta \mathbf{W}_i^{\text{HEBB}}(\eta)$ is a standard correlation rule in which co-active neurons potentiate their connections, and is unsupervised. Because simple Hebbian learning updates are known to be unstable, we include a norm constraint (known as Oja's rule) for positive coefficients η, and normalize the update using the Euclidean norm $\|\cdot\|_2$ of the weights for negative η [5]. The n^{th} row of $\Delta \mathbf{W}_i^{\text{HEBB}}(\eta)$, which we denote as $\Delta \mathbf{w}_i(\eta)_n^T$, corresponding to the weights into the n^{th} hidden unit, is given by

$$\Delta \mathbf{w}_i^{\text{HEBB}}(\eta)_n^T = \begin{cases} \eta \mathbf{h}_{i,n}^{\text{ff}} (\mathbf{h}_{i-1}^T - \mathbf{h}_{i,n}^{\text{ff}} \mathbf{w}_{i,n}^T) & \text{if } \eta > 0 \\ \eta \mathbf{h}_{i,n}^{\text{ff}} \mathbf{h}_{i-1}^T / (1 + \|\mathbf{w}_{i,n}\|_2^2) & \text{otherwise} \end{cases} \tag{12}$$

where $i = 1$ to $D-1$, and $\mathbf{h}_i^{\text{ff}} = \mathbf{W}_i \mathbf{h}_{i-1}$ is the feedforward activity pass from layer $i-1$ to layer i. Extra depth gives rise to slower learning speed, but preserves qualitative features like stage-like transitions, and the formatting of internal representations (Supp. Fig. 10 and Supp. Fig. 11).

References

[1] N. Kriegeskorte, M. Mur, D. Ruff, R. Kiani, J. Bodurka, H. Esteky, K. Tanaka, and P. Bandettini. Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron*, 60(6):1126–1141, 2008.

[2] J. K. Leutgeb, S. Leutgeb, M.-B. Moser, and E. I. Moser. Pattern separation in the dentate gyrus and CA3 of the hippocampus. *Science*, 315(5814):961–966, 2007.

[3] C. E. Myers and H. E. Scharfman. Pattern separation in the dentate gyrus: A role for the CA3 backprojection. *Hippocampus*, 21(11):1190–1215, 2011.

[2] The correlational Hebbian learning is only applied to the mappings between successive hidden layers and the mapping from input layer to the first hidden layer.

[3] In the case of $i = D$, $\mathbf{h}_{i+1} = 0$ and $\mathbf{W}_{i+1} = 0$.

[4] A. Saxe, J. McClelland, and S. Ganguli. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116(23):11537–11546, 2019.
[5] N. N. Schraudolph and T. J. Sejnowski. Competitive anti-hebbian learning of invariants. In *Advances in Neural Information Processing Systems*, pages 1017–1024, 1992.
[6] X. Xie and H. Seung. Equivalence of backpropagation and contrastive Hebbian learning in a layered network. *Neural Computation*, 15(2):441–454, 2003.