# A Minimum Perturbation Theory of Deep Perceptual Learning

Haozhe Shan*

*Center for Brain Science, Harvard University,*

*Cambridge, MA, United States and*

*Program in Neuroscience, Harvard Medical School, Boston, MA, United States*

Haim Sompolinsky

*Center for Brain Science, Harvard University, Cambridge, MA, United States*

*Edmond and Lily Safra Center for Brain Sciences,*

*Hebrew University of Jerusalem, Jerusalem, Israel and*

*Racah Institute of Physics, Hebrew University of Jerusalem, Jerusalem, Israel*

(Dated: September 3, 2022)

## Abstract

Perceptual learning (PL) involves long-lasting improvement in perceptual tasks following extensive training and is accompanied by modified neuronal responses in sensory cortical areas in the brain. Understanding the dynamics of PL and the resultant synaptic changes is important for causally connecting PL to the observed neural plasticity. This is theoretically challenging because learning-related changes are distributed across many stages of the sensory hierarchy. In this work, we modeled the sensory hierarchy as a deep nonlinear neural network and studied PL of fine discrimination, a common and well-studied paradigm of PL. Using tools from statistical physics, we developed a mean-field theory of the network in the limit of large number of neurons and large number of examples. Our theory suggests that, in this "thermodynamic" limit, the input-output function of the network can be exactly mapped to that of a deep *linear* network, allowing us to characterize the space of solutions for the task. Surprisingly, we found that modifications of synaptic weights in lower levels of the hierarchy are both sufficient and necessary for PL. To address the degeneracy of the space of solutions, we postulate that PL dynamics are constrained by a normative "minimum perturbation" (MP) principle, which favors weight matrices with minimal changes relative to their pre-learning values. Interestingly, MP plasticity induces changes to weights and neural representations in all layers of the network, except for the readout weight vector. In addition, such plasticity can be learned simply through slow learning. We further elucidate the properties of MP changes and compare them against experimental findings. Overall, our statistical mechanics theory of PL provides mechanistic and normative understanding of several important empirical findings of PL.


## I. INTRODUCTION

Perceptual learning, the improvement of performance in perceptual tasks after practice, is one of the most basic forms of learning in the brain and has been extensively studied experimentally [1–8]. Physiologically, PL is accompanied by long-lasting changes to neuronal response properties in cortical areas. Connecting physiological changes to behavioral observations has been challenging, in part due to the complex learning dynamics and processing


* hshan@g.harvard.edu; http://hzshan.github.io

in the sensory hierarchy, which is composed of multiple cortical regions. As a result, several important issues concerning the neural mechanisms of PL remain unresolved after decades of research.

First, which cortical areas undergo modifications and which of the changes causally drive PL? While behavioral specificity of PL [9, 10] points to an important role for plasticity in early sensory areas, single-unit response properties in visual areas V1 and V2 show only minor changes after PL [1, 2]. In addition, PL induces significant changes to single-neuron properties in intermediate to late stages of visual processing, such as V4 [3, 4, 7, 8], LIP [11], and IT [12, 13].

Second, what are the functional consequences of the observed changes? Analysis of changes in neuronal responses after PL indicates improved accuracy of the neural coding of the trained stimuli [6–8]. This appears to be inconsistent with the behavioral finding that PL does not transfer to a different task even when using the same stimuli [14–16]. The Reverse Hierarchy Theory [17] proposes that PL is initially driven by learning in high areas, which results in less specific learning; modifications of lower areas follow if the task is difficult, as for instance in fine perceptual discrimination tasks, leading to more specific learning. Analysis of a reduced model of perceptual learning has lent support for this theory [18]. However, recent experimental and computational studies questioned these predictions [6, 7, 19], providing evidence of changes in primary sensory areas already in the early stages of PL. On the other hand, experiments in random dot visual motion discrimination tasks found that PL is correlated with changes in decision-making areas (e.g., LIP) but not sensory areas (e.g., MT) [11, 20]. From a theoretical perspective, the hierarchical nature of the underlying sensory system implies that there is an enormous degeneracy of possible synaptic weight matrices that solve the task of PL.

Most existing theories of PL assume changes only in either the weights of the readout from a fixed sensory array [21–23] or the input layer to a single cortical circuit [19]. Such "shallow" models are inconsistent with the sensory hierarchy in the brain and do not address the neural correlates of PL in multiple cortical regions.

In the present work, we directly addressed the issue of PL in a deep network by studying PL of a fine discrimination task in a deep neural network (DNN) model of the sensory hierarchy [24–26]. As learning dynamics in DNNs are in general challenging to study [27–35], we developed a mean-field theory of information propagation in the model at the limit of

large numbers of neurons in every layer and large number of training examples. The theory reveals that during the perceptual task, the DNN effectively behaves like a deep *linear* neural network. This considerably simplifies the theoretical analysis of the space of solutions, as well as the emergent changes in neural representations. Surprisingly, We found that modifications of synaptic weights in the first level of the hierarchy are both sufficient and necessary for PL. To address the degeneracy of the space of solutions, we developed a *normative* theory of PL. Specifically, we postulated that in the brain, learning dynamics are constrained by a normative "minimum perturbation (MP)" principle, which favors weight matrices with minimal changes relative to their pre-learning values. Interestingly, MP learning induces changes in weights and neural representations in all layers of the networks, except for the readout weight vector. MP learning predicts changes to tuning properties of cortical neurons that are consistent with experimental observations and suggests that signal amplification, not noise reduction, is the primary driver of PL. Our theory makes the readily testable prediction that PL can simultaneously lead to positive and negative transfer to different untrained stimuli. Finally, we found that MP learning can be implemented through slow gradient-descent learning. Overall, leveraging the large size of the network involved in PL, we have developed a statistical mechanics theory of PL in deep neural networks which provides mechanistic and normative understanding of several important empirical findings of PL. This work complements recent theoretical studies of learning in deep networks [27–35], contributing to the understanding of learning and computation in these important architectures.

Our deep network model of PL is described in Section II. The mean field analysis is summarized in Section III. Section IV presents the MP principle and analyzes PL with minimum perturbation. Section V analyzes the use of gradient descent to learn MP plasticity. A discussion of the implications for the field of perceptual learning is presented in Section VI.

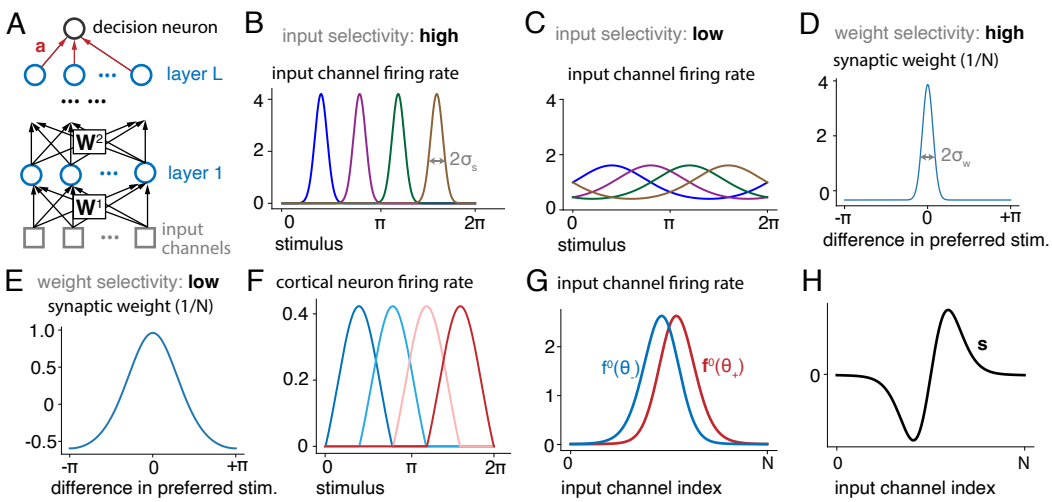


FIG. 1. **Model of perceptual learning.**

**A** Diagram of the deep network model of the sensory hierarchy. The output of input channels (gray squares) is passed through $L$ layers of cortical neurons (blue circles) before getting read out by a linear readout ($\boldsymbol{a}$).

**B** Example tuning curves of input channels. Each curve represents a channel with a different preferred stimulus. The preferred stimuli of input channels uniformly tile $[0, 2\pi]$. This panel shows the regime of high input selectivity and hence narrow tuning curves.

**C** Same as B, but for the scenario of low input selectivity.

**D** Example feedforward weight structure before learning. Weights connecting neurons with similar preferred stimuli tend to be excitatory (positive) and strong while those connecting neurons with dissimilar preferred stimuli tend to be weak and inhibitory (negative). This panel shows the regime of high weight selectivity.

**E** Same as D, but for low weight selectivity.

**F** Bell-shaped tuning curves of input channels and the initial weight patterns lead to bell-shaped tuning curves for all cortical neurons before PL.

**G** Noise-averaged activity of the input channels ($\boldsymbol{f^0}(\theta)$) in response to the two presented stimuli, $\theta_\pm$. The difference between them is exaggerated here for illustration purposes.

**H** The signal $\boldsymbol{s}$ is in the direction of the difference between $\boldsymbol{f^0}(\theta_\pm)$.

## II. A DEEP NETWORK MODEL OF PL

### A. Input channels

We assume $N$ input channels (**Fig. 1A**, gray squares) representing a 1D stimulus. Neurons in the input channels are indexed by a preferred stimulus angle $\theta_i = \frac{i}{N}2\pi$ for the $i$th neuron. The collective response of input channels to a stimulus with angle $\theta$ is given by the $N$-dim vector

$$\boldsymbol{x}^0(\theta) = \boldsymbol{f}^0(\theta) + \boldsymbol{\epsilon}^0, \tag{1}$$

where $\boldsymbol{\epsilon}^0$ is i.i.d. Gaussian noise with zero mean and variance $\sigma^2$. The noise averaged response of each input neuron is given by a bell-shaped tuning curve centered on its preferred stimulus

$$f_i^0(\theta) = Z_s^{-1} \exp\left(\frac{\cos(\theta_i - \theta) - 1}{\sigma_s^2}\right), \tag{2}$$

where $Z_s$ ensures $\|\boldsymbol{f}^0(\theta)\| = \sqrt{N}$ and $\sigma_s$ controls the input selectivity, assumed to be the same for all channels (**Fig. 1C**). Tuning and noise properties of input channels are not affected by learning.

### B. Model architecture and pre-PL weights

Our model of the sensory system is a feedforward network with $L$ hidden layers and a linear readout from the top layer (**Fig. 1A**). Each hidden layer is composed of $N$ rectified linear (ReLU) neurons ("cortical neurons"). Let $\boldsymbol{x}^l(\theta)$ denote the noisy population response vector of neurons in layer $l$, and $\boldsymbol{f}^l(\theta)$ its average over noise. $\{\boldsymbol{x}^l(\theta)\}_{l=1,.,...,L}$ are recursively given by

$$\boldsymbol{x}^l(\theta) = \Phi(\boldsymbol{W}^l \boldsymbol{x}^{l-1}(\theta)), \tag{3}$$

where $\Phi(\cdot)$ is the element-wise rectified linear function. The linear behavioral readout $\boldsymbol{a}$ produces a scalar network output from activity in the last layer

$$r(\theta) = \boldsymbol{a}^T \boldsymbol{x}^L(\theta). \tag{4}$$

Pre-PL weights are modelled after feedforward synaptic connections between visual areas in the brain. We chose a circulant structure which is appropriate for propagating angular signals. Every neuron receives strong, excitatory input from neurons in the previous

layer with similar preferred stimuli and weak, inhibitory input from neurons with dissimilar preferred stimuli. Concretely, pre-PL weights $\{\boldsymbol{W}^l\}_{l=1,2,\ldots,L}$ are given by

$$W^l_{ij,\mathrm{pre}} = Z_w^{-1} \exp\left(\frac{\cos(\theta_i - \theta_j) - 1}{\sigma_w^2}\right) + b_w, \tag{5}$$

where $Z_w$ is chosen such that each row of $\boldsymbol{W}^l_{\mathrm{pre}}$ has norm $1/\sqrt{N}$ (i.e. each weight is $O(N^{-1})$). This normalization ensures that the input to any hidden neuron is of magnitude $O(1)$. The offset $b_w$ is chosen such that each row sums to 0. The parameter $\sigma_w$ controls selectivity of the pre-learning weights; small $\sigma_w$ leads to a high-selectivity weight structure where a few weights dominate the input (**Fig. 1D**) and vice versa (**Fig. 1E**). As a result of the input tuning curves and the feedforward weight structure, all cortical neurons are tuned to the 1D stimulus and have bell-shaped tuning curves before learning (**Fig. 1F**).

### C. Fine-discrimination task

We focus on learning a fine discrimination task, where one out of two similar visual stimuli is presented to the subject, who must correctly indicate which one is presented. In our model, the task consists of discriminating two values of the stimulus, $\theta_\pm = \theta_{\mathrm{tr}} \pm \delta\theta$, where the center stimulus $\theta_{\mathrm{tr}}$ is called the trained stimulus and $\delta\theta \sim O(N^{-1/2})$. This choice of scaling ensures that the total signal-to-noise ratio (SNR) in the input layer is $O(1)$. In each trial, one of $\theta_\pm$ is presented and generates a noisy activation of the input array (Eq. 1, **Fig. 1G**). In each trial, the decision neuron activity $r$ indicates whether the input comes from the $\theta_+$ stimulus or from $\theta_-$ with $r > 0$ or $r < 0$, respectively. Stimuli are presented with equal probability; the optimal performance in the task is thus given by performing maximum likelihood discrimination (MLD [21]). Importantly, since the noise is Gaussian, the task can be performed optimally by a linear discriminator reading out directly from the input channels and using weights parallel to the signal (**Fig. 1H**), defined as the unit vector

$$\boldsymbol{s} = \left(\boldsymbol{f}^0(\theta_+) - \boldsymbol{f}^0(\theta_-)\right) \|\boldsymbol{f}^0(\theta_+) - \boldsymbol{f}^0(\theta_-)\|^{-1}. \tag{6}$$

Thus the output in this scenario equals $\boldsymbol{s}^T\boldsymbol{x}^0(\theta)$ which leads to optimal performance in this setup [36].

### D. Pre-PL readout

We assume the pre-PL value of the readout weight vector $\boldsymbol{a}_{\text{pre}}$ to be optimized for this task when reading out the pre-PL top-layer representations. Thus, we initialize the pre-PL readout such that it minimizes the loss function between the network readout and the optimal output (see below and S.M. I). The rationale for non-random initialization of the readout weights is to provide the network with well-above-chance but generally suboptimal performance (as shown below, it is suboptimal because the top-layer representations may be suboptimal). In the context of animal experiments, this mimics the situation where animals understand the task but have not yet acquired the expert skills required for near optimal performance.

### E. Learning

We model the process of PL as modifying weights in order to minimize the discrimination error. Since the optimal output for this task is given by $\boldsymbol{s}^T\boldsymbol{x}^0(\theta)$ it is convenient to use a mean-squared error objective function

$$E(\Theta) = \langle(\boldsymbol{a}^T\boldsymbol{x}^L(\theta, \boldsymbol{W}^1, ..., \boldsymbol{W}^L) - \boldsymbol{s}^T\boldsymbol{x}^0(\theta))^2\rangle_{\theta=\theta_{\text{tr}}\pm\delta\theta,\boldsymbol{\epsilon}^0}. \tag{7}$$

where $\Theta = (\boldsymbol{W}^1, \boldsymbol{W}^2, ..., \boldsymbol{W}^L, \boldsymbol{a})$ denotes the vector of all weights of the networks and angular brackets denote averaging over the two stimuli and noise. This cost function measures the deviation of the input-output function of the system from the optimal one.

## III. A MEAN-FIELD THEORY OF PL OF THE FINE DISCRIMINATION TASK

In this section, we describe our mean-field approach to studying sensory processing and PL in the deep, nonlinear network by approximating it with an equivalent linear network. We describe the approximation (Sec. III A) and the insights it provides into why pre-PL cortical representations can be suboptimal (Sec. III B). We discuss the space of possible solutions that this theory revealed in Sec. III C.

### A. Equivalent linear networks

First, we note that, during the fine discrimination task, signal and noise-induced fluctuations in the input to any neuron are small (they both scale as $N^{-1/2}$). This can be seen by considering the scaling relations $\delta\theta \sim O(N^{-1/2})$, $\sigma^2 \sim O(1)$ and $W^l_{ij} \sim O(N^{-1})$. At the large $N$ limit, we can expand activities of cortical neurons around their average inputs by writing (using $\odot$ to denote the Hadamard product and $\boldsymbol{f}^l$ to denote $\boldsymbol{f}^l(\theta_{\rm tr})$; this is similar to the approximation done in [37] for recurrent networks)

$$\boldsymbol{x}^l(\theta_\pm) = \Phi\left(\boldsymbol{W}^l \boldsymbol{x}^{l-1}(\theta_\pm)\right) \tag{8}$$

$$\approx \Phi\left(\boldsymbol{W}^l \boldsymbol{f}^{l-1}\right) + \Phi'\left(\boldsymbol{W}^l \boldsymbol{f}^{l-1}\right) \odot \left[\pm\delta\theta \boldsymbol{W}^l d_\theta \boldsymbol{f}^{l-1} + \boldsymbol{W}^l \boldsymbol{\epsilon}^{l-1}\right], \tag{9}$$

where

$$d_\theta \boldsymbol{f}^l = \frac{\left[\boldsymbol{f}^l(\theta_+) - \boldsymbol{f}^l(\theta_-)\right]}{2\delta\theta} \tag{10}$$

and $\boldsymbol{\epsilon}^l = \frac{1}{2}\left[\boldsymbol{x}^l(\theta_+) + \boldsymbol{x}^l(\theta_-) - \boldsymbol{f}^l(\theta_+) - \boldsymbol{f}^l(\theta_-)\right]$ are the signal-induced and noise-induced fluctuations in layer $l$, respectively. At large $N$, by the centra limit theorem, the components of noise are Gaussian (though correlated). For ReLU nonlinearity, the activation slope $\Phi'_i$ is 1 for an *active* neuron and zero for an *inactive* one. At the limit of large $N$, fluctuations in the input to each neuron are small compared to the mean. Thus, inactive neurons remain quiescent for most of the trials and do not contribute to the network output. For a similar reason, activities of active neurons are $[\boldsymbol{W}^l \boldsymbol{f}^{l-1}]_i$ i.e, they are *linear* functions of activities of neurons in the previous layer. Thus, we can define effective weight matrices , $\boldsymbol{W}^l_{\rm eff}$, as

$$\left(W^l_{\rm eff}\right)_{ij} = W^l_{ij}, \text{ iff } f^l_i > 0 \text{ and } f^{l-1}_j > 0 \tag{11}$$

and zero otherwise. Importantly, this approximation holds only during fine discrimination around a fixed $\theta_{\rm tr}$ because of the strong similarity between the different inputs. Inputs with angles very different from $\theta_{\rm tr}$ will be processed by different sets of effective weights. Given the analysis above, during the fine discrimination task around $\theta_{\rm tr}$ the input output function of the deep network is effectively linear,

$$r(\theta) = \boldsymbol{a}^T \boldsymbol{P} \boldsymbol{x}^{\boldsymbol{0}}(\theta) \tag{12}$$

where

$$\boldsymbol{P} = \boldsymbol{W}^L_{\rm eff} \boldsymbol{W}^{L-1}_{\rm eff} ... \boldsymbol{W}^1_{\rm eff}. \tag{13}$$

We call $P$ the *processing matrix* (**Fig. 2A**, right). We proceed to consider how the properties of $P$ affect task performance.

### B. Pre-learning suboptimal representations

Optimizing $\boldsymbol{a}_{\text{pre}}$ amounts to optimizing a linear readout from an input $\boldsymbol{P}\boldsymbol{x}^{\boldsymbol{0}}(\theta)$ which contains a signal and an additive (correlated) noise. In such a system, the probability of error under optimal readout is given by $H(\delta\theta\sqrt{J_L})$ where $H(x) = (2\pi)^{-1/2}\int_x^\infty e^{-z^2/2}dz$ and $J_L$ is the linear Fisher information [38]. It is defined as

$$J_L = (d_\theta \boldsymbol{f}^L)^T \boldsymbol{\Sigma}_L^{-1} d_\theta \boldsymbol{f}^L, \tag{14}$$

where the matrix $\boldsymbol{\Sigma}_L$ is the noise covariance matrix in the top layer. Given the linear approximation, it is given by $\boldsymbol{\Sigma}_L = \sigma^2 \boldsymbol{P}\boldsymbol{P}^T$ and the top layer signal is $d_\theta \boldsymbol{f}^L = \boldsymbol{P} d_\theta \boldsymbol{f}^0$. Even if $\boldsymbol{P}$ is low rank (see below), the (pseudo)-inverse $(\boldsymbol{P}\boldsymbol{P}^T)^{-1}$ is well defined when multiplied by $\boldsymbol{P}$. Hence,

$$J_L = \sigma^{-2}(d_\theta \boldsymbol{f}^0)^T \boldsymbol{P}^T (\boldsymbol{P}\boldsymbol{P}^T)^{-1} \boldsymbol{P} d_\theta \boldsymbol{f}^0. \tag{15}$$

Note that $\boldsymbol{P}^T(\boldsymbol{P}\boldsymbol{P}^T)^{-1}\boldsymbol{P}$ is a projection matrix. It is identity if $\boldsymbol{P}$ is full rank. Otherwise, it projects inputs onto the low-rank subspace spanned by its rows. Thus, Eq. 15 states that $\sigma^2 J_L$ is the squared norm of the projection of the signal vector $d_\theta \boldsymbol{f}^0 \propto \boldsymbol{s}$ onto the subspace spanned by $\boldsymbol{P}$. The network is optimal if $\boldsymbol{s}$ resides in the span of $\boldsymbol{P}$, yielding $J_L = J_0 = \sigma^{-2}\|d_\theta \boldsymbol{f}^0\|^2$.

We now ask whether the pre-PL weights are already optimal for the present task. We computed the singular value decomposition of the pre-PL $\boldsymbol{P}$ and found that it has a low rank structure (Fig. S2). There are two sources of the reduced rank of $\boldsymbol{P}$ depending on the system parameter regime: a "selective-input-unselective-weights" regime characterized by small $\sigma_s$ and large $\sigma_w$ (**Fig. 2B**, blue) and an "unselective-input-selective-weights" regime, characterized by large $\sigma_s$ and small $\sigma_w$ (**Fig. 2B**, red). In the selective-input-unselective-weights regime, the pre-PL network weights $\boldsymbol{W}^l_{\text{pre}}$ are low rank (even before rectification) due to the smoothness of circulant weights, implying that they project to subsequent layers only part of the signal in the input. In this regime, information loss occurs regardless of the rectification of representation neurons (**Fig. 2C, D**, blue line). On the other hand, in the unselective-input-selective-weights (large $\sigma_s$, small $\sigma_w$) regime, the

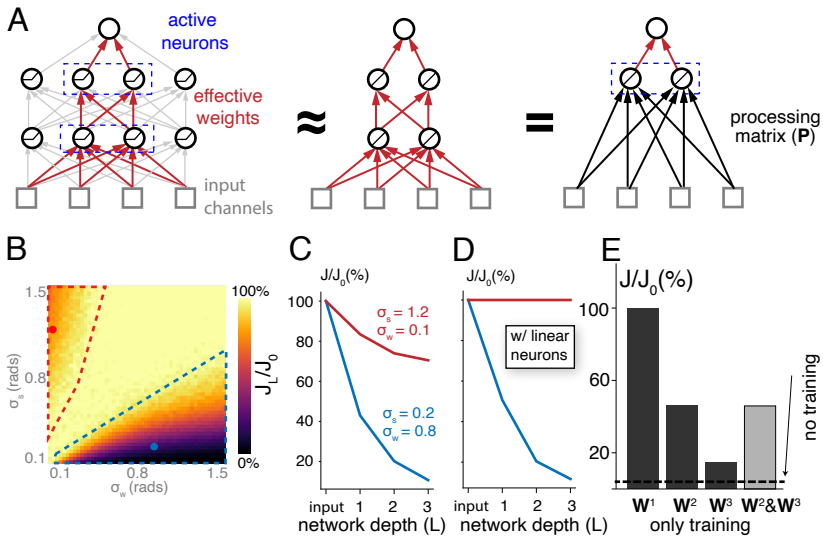


FIG. 2. **Suboptimal neural representations before learning.**

**A** Schematics showing the relationship between weights (all arrows, left), effective weights (red arrows, left and center), and the processing matrix (black arrows, right).

**B** Information for the trained stimulus in the last layer ($J_L$) divided by the input information ($J_0$), for different input and weight selectivity. The ratio is low for large $\sigma_s$, small $\sigma_w$(red polygon) or small $\sigma_s$, large $\sigma_w$ ( blue polygon). Dots: example parameters used in C, D. $N = 1000$ in all panels. $L = 1$ in this panel. See Fig. S1 for deeper networks.

**C** Information for the trained stimulus in the last layer of networks of different depths, divided by the input-layer information ($J_0$).

**D** Same as E, but assuming that all neurons are active.

**E** Best last-layer information achievable if plasticity is restricted to some weight matrices in a three-layer network. Dashed line: performance if no weight matrix is modified. Modifying any weight matrix improves the performance, but only modifying $\boldsymbol{W}^1$ is sufficient and necessary for optimizing it.

original weight matrices project the full signal. However, due to firing-rate rectification, a substantial fraction of the neurons are inactive for essentially all training stimuli. Thus, the effective weights are low rank. In this regime, the low-rank structure of $\boldsymbol{P}$ disappears if we remove neuronal rectification (**Fig. 2C, D**, red line). In both cases, the signal contains a substantial component perpendicular to the low-rank span of pre-PL $\boldsymbol{P}$, as evidenced by computing $J_L/J_0$ (**Fig. 2B**). Hence the pre-PL network exhibits suboptimal performance.

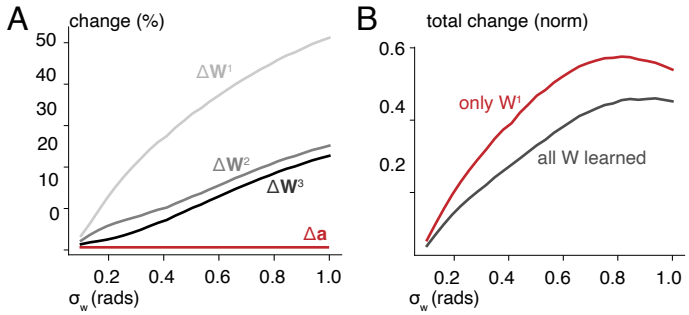


FIG. 3. **Minimum-perturbation plasticity of perceptual learning.**

**A** The magnitude of synaptic changes to each matrix and the readout vector $\boldsymbol{a}$, for networks initialized with different $\sigma_w$. Percent change is defined as the Frobenius norm of synaptic changes divided by that of the pre-PL weight matrix. In all panels, $\sigma_s = 0.2, L = 3, N = 1000$.

**B** Restricting learning to $\boldsymbol{W}^1$ leads to more network-wide perturbation (measured by the sum of matrix norms of $\Delta\boldsymbol{W}^1, \Delta\boldsymbol{W}^2, \Delta\boldsymbol{W}^3$) than unrestricted learning. In either case, the readout $\boldsymbol{a}$ is also allowed to learn but does not change significantly following PL.

### C. Space of solutions

We derived the following *necessary and sufficient condition* on post-learning effective weights that accomplish this goal (derivations in S.M. II; hereafter we use $\boldsymbol{W}^l$ to denote $\boldsymbol{W}^l_{\text{eff}}$ for brevity): for any values of $\boldsymbol{W}^2, ..., \boldsymbol{W}^L$ and $\boldsymbol{a}$ that satisfy $\tilde{\boldsymbol{a}} \equiv (\boldsymbol{W}^2)^T ... (\boldsymbol{W}^L)^T \boldsymbol{a} \neq \boldsymbol{0}$, the task can be performed optimally if and only if $\boldsymbol{W}^1$ satisfies

$$\boldsymbol{W}^1 = \|\tilde{\boldsymbol{a}}\|^{-2} \tilde{\boldsymbol{a}} \boldsymbol{s}^T + \boldsymbol{W}_\perp, \tag{16}$$

where $\boldsymbol{W}_\perp^T \tilde{\boldsymbol{a}} = 0$. This result implies that the learning problem can be solved for essentially arbitrary (non-zero) higher-layer weights as long as $\boldsymbol{W}^1$ is adjusted accordingly. Conversely, restricting the plasticity to higher-layer weights while freezing the first layer weights to their pre-PL values does not obey this condition; thus this is insufficient for optimal performance, as discussed earlier (**Fig. 2E**). This result underscores the critical role played by early sensory areas in PL.

## IV. LEARNING WHILE MINIMIZING NETWORK PERTURBATION

The large space of solutions makes it hard to predict the pattern of changes in the circuit induced by learning. To remove this degeneracy, we propose an optimality criterion, "minimum perturbation" (MP), that favors a solution with small perturbations to pre-PL weights. According to this criterion, the optimal weights are

$$\Delta\Theta = \Theta_{\text{post}} - \Theta_{\text{pre}} \tag{17}$$

$$\Theta_{\text{opt}} = \arg\min_{\Theta_{\text{post}}} \|\Delta\Theta\|^2 \text{ s.t. } E(\Theta_{\text{post}}) = 0, \tag{18}$$

where "post" indicates post-PL weights. This principle is motivated by the brain's need to maintain relatively stable representations while learning a new task. We analytically solved this optimization problem with Lagrange multipliers by first setting up the Lagrangian

$$\mathcal{L} = \sum_{l=1}^{L} \|\Delta \boldsymbol{W}^l\|^2 + \|\Delta \boldsymbol{a}\|^2 - \boldsymbol{\lambda}^T [\boldsymbol{W}_{\text{post}}^{1\,T} \boldsymbol{W}_{\text{post}}^{2\,T} ... \boldsymbol{W}_{\text{post}}^{L\,T} \boldsymbol{a}_{\text{post}} - \boldsymbol{s}]. \tag{19}$$

Extremizing the Lagrangian w.r.t. weight changes reveals a general rank-1 structure for MP $\Delta \boldsymbol{W}^l$

$$\Delta \boldsymbol{W}^l = (\boldsymbol{W}_{\text{post}}^{l+1\,T} \cdots \boldsymbol{W}_{\text{post}}^{L\,T} \boldsymbol{a}_{\text{post}})(\boldsymbol{W}_{\text{post}}^{l-1} \cdots \boldsymbol{W}_{\text{post}}^{1} \boldsymbol{\lambda})^T. \tag{20}$$

Solving the above equations requires introducing $2L$ scalar order parameters, which obey $2L$ self-consistent equations that need to be solved numerically. Expressions of the order parameters and self-consistent equations for $L = 1, 2, 3$, as well as the numerical procedure for solving the self-consistent equations, are given in S.M. III.

We discuss features of the solutions below.

**Distribution of MP plasticity**: We quantified the magnitude of MP modification to weights in each layer by computing $\|\Delta \boldsymbol{W}^l\| / \|\boldsymbol{W}_{\text{pre}}^l\|$ (**Fig. 3A**; see Fig. S5 for the unselective-input-selective-weights regime). The analysis reveals two salient features that are consistent across system parameters. First, MP plasticity predominantly affects lower-layer weights. Second, surprisingly, MP plasticity does not appreciably alter the readout (**Fig. 3A**, red line). This suggests that hidden-layer representation changes, rather than readout changes, drive performance improvement. Plasticity in higher-layer weights $\boldsymbol{W}^{l\geq 2}$ plays the important role of reducing the overall perturbation to the network. Indeed, if we restrict learning to $\boldsymbol{W}^{\mathbf{1}}$, the total perturbation is greater (**Fig. 3B**).

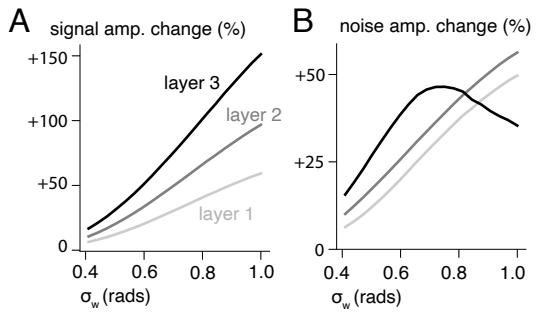


FIG. 4. **MP learning-induced changes to signal and noise.**

**A, B** PL-induced changes to signal (A) or noise (B) amplitude across layers for different weight selectivity. Changes are generally greater in higher layers and in networks with initial weights that are less selective (larger $\sigma_w$). In both panels, $\sigma_s = 0.4, \sigma_w = 1.0, N = 1000, L = 3$.

**Performance improvement is driven by signal amplification**: An important and long-standing debate in PL research is whether behavioral improvement is driven by signal amplification, noise suppression, or both [1, 3, 4, 7, 39–42]. To address this question within the framework of MP learning, we define the signal and noise contributions via $J_L = (\text{signal/noise})^2$, where the signal amplitude is $\|d_\theta \boldsymbol{f}^L\|$ and the noise amplitude is defined via $\text{noise}^{-2} = (d_\theta \boldsymbol{f}^L)^T \boldsymbol{\Sigma}_{L-1} d_\theta \boldsymbol{f}^L / \|d_\theta \boldsymbol{f}^L\|^2$. The network after MP learning exhibits a pronounced amplification of signal (**Fig. 4A**), with the effect being stronger in higher layers. Surprisingly, we found that PL also *amplifies* noise across all layers, although to a weaker extent than signal amplification (**Fig. 4B**; effects on noise correlation are shown in Fig. S4). Thus, MP learning improves perceptual performance by strengthening the signal rather than weakening the noise. This analysis also reveals that signal/noise changes are generally greater in higher layers even though weight changes are greater in lower layers, highlighting the difference between distribution of weight changes and distribution of representation changes.

**Impact on discrimination around untrained stimuli**: MP plasticity breaks the symmetry of pre-PL representations w.r.t. $\theta$, thus alters the representations of untrained stimuli. To assess how these changes affect the discrimination ability of angles around untrained values we define a normalized information gain, $[J_{L,\text{post}}(\theta) - J_{L,\text{pre}}(\theta)]/[J_{L,\text{post}}(\theta_{\text{tr}}) - J_{L,\text{pre}}(\theta_{\text{tr}})]$ for an untrained stimulus $\theta$. Our analysis revealed a rich, non-monotonic pattern of transfer arising from MP plasticity. Consistent with experimental findings, PL increases

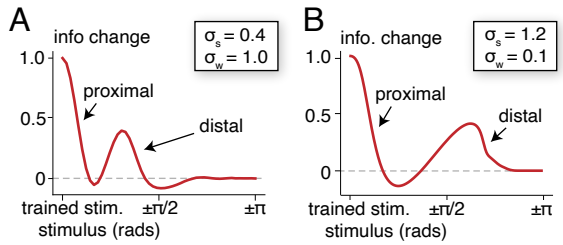


FIG. 5. **Transfer of PL to untrained stimuli.**

**A, B** Information changes in the last layer for different stimuli after PL, normalized by change for the trained stimulus. The change for the trained stimulus is 1 by definition. Information gain is prominent for stimuli close to the trained one ("proximal"), and those dissimilar from the trained one ("distal"). In all panels, $N = 1000, L = 3$, and the last layer is analyzed. For the selective-input-unselective-weights regime (**A**), $\sigma_s = 0.4, \sigma_w = 1.0$. For the unselective-input-selective-weights regime (**B**), $\sigma_s = 1.2, \sigma_w = 0.1$.

information for stimuli similar to the trained one ("proximal transfer", **Fig. 5**). In addition, PL also transfers to distal stimuli, where the distance between trained and test stimuli is intermediate ("distal transfer"). Importantly, PL can also decrease information for certain untrained stimuli (negative transfer), as revealed by the dips below 0 in **Fig. 5**. Finally, as expected, representations for stimuli far away from the trained one are unaffected by learning.

## V. MP LEARNING WITH GRADIENT DESCENT

So far, our analysis has focused on properties of MP plasticity without addressing the important question of *how* such plasticity is learned. We modeled the process of learning by studying gradient descent (GD), which has been shown to reproduce physiological features of PL in deep network models [18, 43]. We used GD to optimize $\Theta$ for a regularized loss function, defined as

$$E_{\text{reg}}(\Theta) = E(\Theta) + \lambda \|\Delta\Theta\|^2, \tag{21}$$

where the second term imposes a penalty on weight perturbation; strength of the penalty is controlled by the hyperparameter $\lambda$. We implemented GD by iterating $\Theta(t+1) = \Theta(t) - \eta \nabla_{\Theta(t)} E_{\text{reg}}(\Theta)$ until convergence, where $\eta$ is the learning rate. At convergence, larger $\lambda$ results in smaller weight perturbations but potentially suboptimal post-PL performance. To

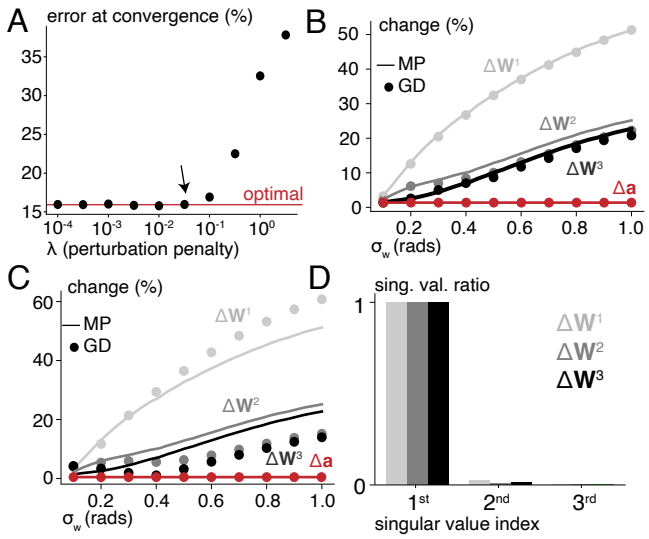


FIG. 6. **MP learning with gradient descent.**

**A** Discrimination error rate at convergence after regularized gradient descent with different perturbation penalty strength. Arrow: maximum $\lambda$ with optimal performance. In all panels $\sigma_s = 0.2, L = 3$; $\sigma_w = 0.8$ except in B, D.

**B** Magnitude of changes from regularized gradient descent (dots), compared against that of MP plasticity (line).

**C** Same as B, but for slow GD without explicit regularization.

**D** Leading singular values of of slow GD induced changes to weight matrices (normalized by the top singular value). That the first singular value is overwhelmingly large suggests that induced changes are close to rank-1.

realize MP learning, $\lambda$ should be as large as possible without making the final performance suboptimal (**Fig. 6A**). GD with such $\lambda$ results in weight changes that are fully consistent with MP plasticity (data not shown).

In the deep learning literature, it has been suggested that small changes in learned parameters can also be realized through implicit regularization by using small learning rates [44, 45]. We performed GD on the unreguarlized loss function, $E(\Theta)$, with a small learning rate $\eta$. The resultant plasticity agrees reasonably well with MP plasticity (**Fig. 6C**) in terms of magnitude. It also has the same salient features as MP plasticity: changes to the readout are negligible, weight changes are very close to being rank-1 (**Fig. 6D**) and the identity of active neurons does not change over learning (S.M. V), as is the case for MP

plasticity. These results point to the possibility that the slow progression of PL could be normatively explained as a mechanism to minimize perturbation during PL.

## VI. DISCUSSION

We have presented a theory of PL of fine discrimination in a deep network. The theory leverages similarity of all inputs relevant to the task, large network size and structured weight initialization to establish the effective linearity of the network input-output function during training and performance of the task. This input-output function is expressed by a processing matrix $\boldsymbol{P}$ which has been shown to be low rank, hence leading to a sub-optimal representation of the stimulus that cannot be resolved by adapting the readout weight only. We further derived the space of post-learning weights that resolve the suboptimality by fully spanning the task-relevant signal direction. Motivated by the brain's need to strike a balance between plasticity (acquiring new skills) and stability (preventing previously learned skills to be affected) in sensory areas [46], we propose the normative minimum perturbation principle that favors a specific solution. The favored solution, which we call MP plasticity, induces physiological and behavioral changes largely consistent with current experimental findings (for a detailed comparison, see S.M. VII). It also predicts that PL improves the sensory code for some untrained stimuli while degrades the representation of others, a readily testable prediction. We discuss some prominent features of MP plasticity and their implications for neural mechanisms of PL.

First, MP plasticity predominantly modifies the lowest-level weights while leaving the readout essentially unchanged. This points to the importance of involving low-level cortical areas in PL of fine discrimination, consistent with recent numerical experiments with deep convolutional networks [43]. That the readout is unchanged critically depends on our assumption that the pre-PL readout is already optimized w.r.t. pre-learning representations, in contrast to most neural network models of learning where the initial weights are random. We argue that random initialization is not biologically plausible when considering naturalistic tasks where subjects perform well above chance with little-to-no training. Note that while synaptic plasticity is greater in lower layers, the resultant representational changes are greater in *higher* layers.

Second, MP plasticity makes rank-1 modifications to weights. While rank-1 weight

changes are sufficient for optimizing neural representations for the trained task, such changes can be highly task-specific. To demonstrate this, we analyzed performance of the post-PL network on a width discrimination task where two stimuli with the same $\theta$ but different $\sigma_s$ are presented; PL does not improve this performance, despite the fact that width discrimination and angle discrimination involve the same mean stimulus (S.M. IV). Importantly, this absence of cross-task transfer reconciles the apparent inconsistency of the observed improved sensory representations by PL [1, 5, 6, 8] and the psychophysical findings that PL for one task did not transfer to another task using the same stimuli [14–16], which was interpreted as evidence that population codes for these stimuli did not improve [47]. Our results suggest that the improvement of representations does not equally benefit all tasks even if they share the same stimuli. Thus, cross-task transfer is not a reliable indicator of whether representations improve after PL.

Finally, from the perspective of signal and noise, MP plasticity improves task performance by amplifying the signal. This result is inconsistent with [19], who found that amplification is not necessary for PL. Their conclusion may be confined to the regime where performance is dominated by neural noise, not input noise as in ours. Additionally, their plasticity model differs from ours in that it assumes circularly invariant weights both before and after learning, which forces a global change of synaptic weights. In contrast, in our model, PL plasticity is localized to the neurons responding to the stimulus (if we require post-PL weights to be circularly invariant in our model, post-PL tuning curves have very unnatural multi-modal shapes. See S.M. VIII). Finally, we note that our prediction of signal amplification stems from the fact that the readout layer remains essentially unchanged under MP learning. If the readout were adapted in ways that violate the MP principle, signal amplification is not always necessary (S.M. IX).

Our current theory can be extended in several interesting directions. Our plasticity model does not include a mechanism of unsupervised learning, namely, plasticity triggered by the mere exposure to the stimulus, independent of task. Thus, including considerations for task-irrelevant plasticity, observed in some PL studies [48, 49] is an interesting topic for future work. Additionally, it would be interesting to add to our architecture recurrent connections

within each layer and to consider the effect of neuronal noise on MP learning.

---


[1] A. Schoups, R. Vogels, N. Qian, and G. Orban, Nature **412**, 549 (2001).

[2] G. M. Ghose, T. Yang, and J. H. Maunsell, Journal of neurophysiology **87**, 1867 (2002).

[3] T. Yang and J. H. Maunsell, Journal of Neuroscience **24**, 1617 (2004).

[4] S. Raiguel, R. Vogels, S. G. Mysore, and G. A. Orban, Journal of Neuroscience **26**, 6589 (2006).

[5] Y. Gu, S. Liu, C. R. Fetsch, Y. Yang, S. Fok, A. Sunkara, G. C. DeAngelis, and D. E. Angelaki, Neuron **71**, 750 (2011).

[6] Y. Yan, M. J. Rasch, M. Chen, X. Xiang, M. Huang, S. Wu, and W. Li, Nature neuroscience **17**, 1380 (2014).

[7] M. Sanayei, X. Chen, D. Chicharro, C. Distler, S. Panzeri, and A. Thiele, Nature communications **9**, 1 (2018).

[8] A. M. Ni, D. A. Ruff, J. J. Alberts, J. Symmonds, and M. R. Cohen, Science **359**, 463 (2018).

[9] A. Karni and D. Sagi, Proceedings of the National Academy of Sciences **88**, 4966 (1991).

[10] A. Fiorentini and N. Berardi, Nature **287**, 43 (1980).

[11] C.-T. Law and J. I. Gold, Nature neuroscience **11**, 505 (2008).

[12] H. Z. Adab, I. D. Popivanov, W. Vanduffel, and R. Vogels, Journal of cognitive neuroscience **26**, 2187 (2014).

[13] H. O. de Beeck, J. Wagemans, and R. Vogels, Neuroscience **145**, 775 (2007).

[14] M. Ahissar and S. Hochstein, Proceedings of the National Academy of Sciences **90**, 5718 (1993).

[15] R. E. Crist, M. K. Kapadia, G. Westheimer, and C. D. Gilbert, Journal of neurophysiology **78**, 2889 (1997).

[16] M. Fahle, Vision research **37**, 1885 (1997).

[17] M. Ahissar and S. Hochstein, Trends in cognitive sciences **8**, 457 (2004).

[18] A. M. Saxe, *Deep linear neural networks: A theory of learning in the brain and mind* (Stanford University, 2015).

[19] V. R. Bejjanki, J. M. Beck, Z.-L. Lu, and A. Pouget, Nature neuroscience **14**, 642 (2011).

[20] S. A. Chowdhury and G. C. DeAngelis, Neuron **60**, 367 (2008).

[21] H. S. Seung and H. Sompolinsky, Proceedings of the national academy of sciences **90**, 10749 (1993).

[22] A. A. Petrov, B. A. Dosher, and Z.-L. Lu, Psychological review **112**, 715 (2005).

[23] B. A. Dosher, P. Jeter, J. Liu, and Z.-L. Lu, Proceedings of the National Academy of Sciences **110**, 13678 (2013).

[24] J. J. DiCarlo and D. D. Cox, Trends in cognitive sciences **11**, 333 (2007).

[25] D. L. Yamins and J. J. DiCarlo, Nature neuroscience **19**, 356 (2016).

[26] A. J. Kell and J. H. McDermott, Current opinion in neurobiology **55**, 121 (2019).

[27] J. Lee, Y. Bahri, R. Novak, S. S. Schoenholz, J. Pennington, and J. Sohl-Dickstein, arXiv preprint arXiv:1711.00165 (2017).

[28] A. Jacot, F. Gabriel, and C. Hongler, Advances in neural information processing systems **31** (2018).

[29] B. Bordelon and C. Pehlevan, arXiv preprint arXiv:2205.09653 (2022).

[30] Q. Li and H. Sompolinsky, Physical Review X **11**, 031059 (2021).

[31] A. M. Saxe, J. L. McClelland, and S. Ganguli, arXiv preprint arXiv:1312.6120 (2013).

[32] S. Mei, A. Montanari, and P.-M. Nguyen, Proceedings of the National Academy of Sciences **115**, E7665 (2018).

[33] L. Zdeborová, Nature Physics **16**, 602 (2020).

[34] Y. Bahri, J. Kadmon, J. Pennington, S. S. Schoenholz, J. Sohl-Dickstein, and S. Ganguli, Annual Review of Condensed Matter Physics **11** (2020).

[35] D. A. Roberts, S. Yaida, and B. Hanin, arXiv preprint arXiv:2106.10165 (2021).

[36] C. R. Rao, Journal of the Royal Statistical Society. Series B (Methodological) **10**, 159 (1948).

[37] J. Beck, V. R. Bejjanki, and A. Pouget, Neural computation **23**, 1484 (2011).

[38] P. Seriès, P. E. Latham, and A. Pouget, Nature neuroscience **7**, 1129 (2004).

[39] J. Poort, A. G. Khan, M. Pachitariu, A. Nemri, I. Orsolic, J. Krupic, M. Bauza, M. Sahani, G. B. Keller, T. D. Mrsic-Flogel, *et al.*, Neuron **86**, 1478 (2015).

[40] E. Zohary, S. Celebrini, K. H. Britten, and W. T. Newsome, Science **263**, 1289 (1994).

[41] H. Z. Adab and R. Vogels, Current biology **21**, 1661 (2011).

[42] J. Gold, P. Bennett, and A. Sekuler, Nature **402**, 176 (1999).

[43] L. K. Wenliang and A. R. Seitz, Journal of Neuroscience **38**, 6028 (2018).

[44] S. Gunasekar, J. Lee, D. Soudry, and N. Srebro, in *International Conference on Machine*

*Learning* (PMLR, 2018) pp. 1832–1841.

[45] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, Communications of the ACM **64**, 107 (2021).

[46] W. C. Abraham and A. Robins, Trends in neurosciences **28**, 73 (2005).

[47] B. Dosher and Z.-L. Lu, Annual Review of Vision Science **3**, 343 (2017).

[48] T. Watanabe, J. E. Nanez, and Y. Sasaki, Nature **413**, 844 (2001).

[49] A. R. Seitz and T. Watanabe, Vision research **49**, 2604 (2009).

[50] L. M. Vaina, J. W. Belliveau, E. B. Des Roziers, and T. A. Zeffiro, Proceedings of the National Academy of Sciences **95**, 12657 (1998).

[51] M. Sigman, H. Pan, Y. Yang, E. Stern, D. Silbersweig, and C. D. Gilbert, Neuron **46**, 823 (2005).

[52] J. F. Jehee, S. Ling, J. D. Swisher, R. S. van Bergen, and F. Tong, Journal of Neuroscience **32**, 16747 (2012).

[53] Y. Wang, W. Wu, X. Zhang, X. Hu, Y. Li, S. Lou, X. Ma, X. An, H. Liu, J. Peng, *et al.*, Frontiers in behavioral neuroscience **10**, 42 (2016).

[54] R. E. Crist, W. Li, and C. D. Gilbert, Nature neuroscience **4**, 519 (2001).

[55] G. H. Recanzone, M. M. Merzenich, W. M. Jenkins, K. A. Grajski, and H. R. Dinse, journal of Neurophysiology **67**, 1031 (1992).

[56] G. H. Recanzone, C. E. Schreiner, and M. M. Merzenich, Journal of Neuroscience **13**, 87 (1993).

[57] B. Pleger, A.-F. Foerster, P. Ragert, H. R. Dinse, P. Schwenkreis, J.-P. Malin, V. Nicolas, and M. Tegenthoff, Neuron **40**, 643 (2003).

[58] S. Schwartz, P. Maquet, and C. Frith, Proceedings of the National Academy of Sciences **99**, 17137 (2002).

[59] C. S. Furmanski, D. Schluppeck, and S. A. Engel, Current Biology **14**, 573 (2004).

[60] C. Schiltz, J. Bodart, S. Dubois, S. Dejardin, C. Michel, A. Roucoux, M. Crommelinck, and G. Orban, Neuroimage **9**, 46 (1999).

[61] A. A. Schoups, R. Vogels, and G. A. Orban, The Journal of physiology **483**, 797 (1995).

[62] M. Fahle and S. Edelman, Vision research **33**, 397 (1993).

[63] L.-P. Shiu and H. Pashler, Perception & psychophysics **52**, 582 (1992).

[64] P. E. Jeter, B. A. Dosher, A. Petrov, and Z.-L. Lu, Journal of vision **9**, 1 (2009).

## ACKNOWLEDGMENTS

The authors would like to thank Andrew Saxe and Ravid Ziv for very helpful discussions. This research was partially supported by the Swartz Program in Theoretical Neuroscience at Harvard University, the Gatsby Charitable Foundation, the National Institute of Neurological Disorders and Stroke (Grant No.1U19NS104653), and the National Science Foundation (Grant No.1806818). This paper is dedicated to the memory of Mrs. Lily Safra, a great supporter of brain research.

# Supplemental Materials for *A Minimum Perturbation Theory of Deep Perceptual Learning*

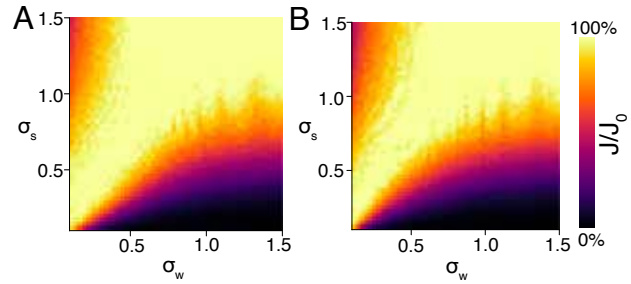


FIG. S1. Information loss ($J/J_0$) in untrained $L = 2$ (A) and $L = 3$ (B) networks. The $L = 1$ case is shown in Fig. 2B. The two panels share the color bar. In each case, there are two regimes corresponding to significant information loss. In all three cases, information loss is significant when the input is selective and the weights are unselective, or when the input is unselective and the weights are selective.

## I. READOUT INITIALIZATION

For brevity, we use $\boldsymbol{W}^l$ to refer to $\boldsymbol{W}^l_{\text{eff}}$ unless otherwise noted.

Under the linear approximations, the optimization problem in Eq. 7 can be solved in closed-form over $\boldsymbol{a}$. We can express the optimal $\boldsymbol{a}_0$ as

$$\boldsymbol{a}_0 = \langle \boldsymbol{P}\boldsymbol{x}^0(\theta)\boldsymbol{x}^0(\theta)^T\boldsymbol{P}^T\rangle^{\dagger}_{\theta=\theta_\pm,\boldsymbol{\epsilon}^0}\boldsymbol{P}\langle \boldsymbol{x}^0(\theta)\boldsymbol{x}^0(\theta)^T\rangle_{\theta=\theta_\pm,\boldsymbol{\epsilon}^0}\boldsymbol{s}, \tag{S1}$$

where $\boldsymbol{P}$ is defined in Eq. 13. This can be written more explicitly in term of the truncated singular value decomposition of $\boldsymbol{P} = \boldsymbol{A}\boldsymbol{\Lambda}\boldsymbol{B}$ as

$$\boldsymbol{a}_0 = \frac{1+\text{SNR}}{1+\text{SNR}/\|\boldsymbol{B}\boldsymbol{s}\|^2}\boldsymbol{A}\boldsymbol{\Lambda}^{-1}\boldsymbol{B}\boldsymbol{s}, \tag{S2}$$

where $\text{SNR} = (\delta\theta)^2\|d_\theta \boldsymbol{f}^0\|^2/\sigma^2$ is the input signal-to-noise ratio (SNR), which is set to 1 in simulations (see the specific values used in simulations in S.M. VI).

## II. SPACE OF SOLUTIONS

In this section we prove the sufficient and necessary condition (Eq. 16).

Formally, let $\tilde{\boldsymbol{a}} = \boldsymbol{a}^T\boldsymbol{W}^L\boldsymbol{W}^{L-1}...\boldsymbol{W}^2 \neq \boldsymbol{0}$. We claim

$$\boldsymbol{W}^{1^T}\tilde{\boldsymbol{a}} = \boldsymbol{s} \iff \boldsymbol{W}^1 = \|\tilde{\boldsymbol{a}}\|^{-2}\tilde{\boldsymbol{a}}\boldsymbol{s}^T + \boldsymbol{W}_\perp, \quad \boldsymbol{W}_\perp^T\tilde{\boldsymbol{a}} = \boldsymbol{0}. \tag{S3}$$

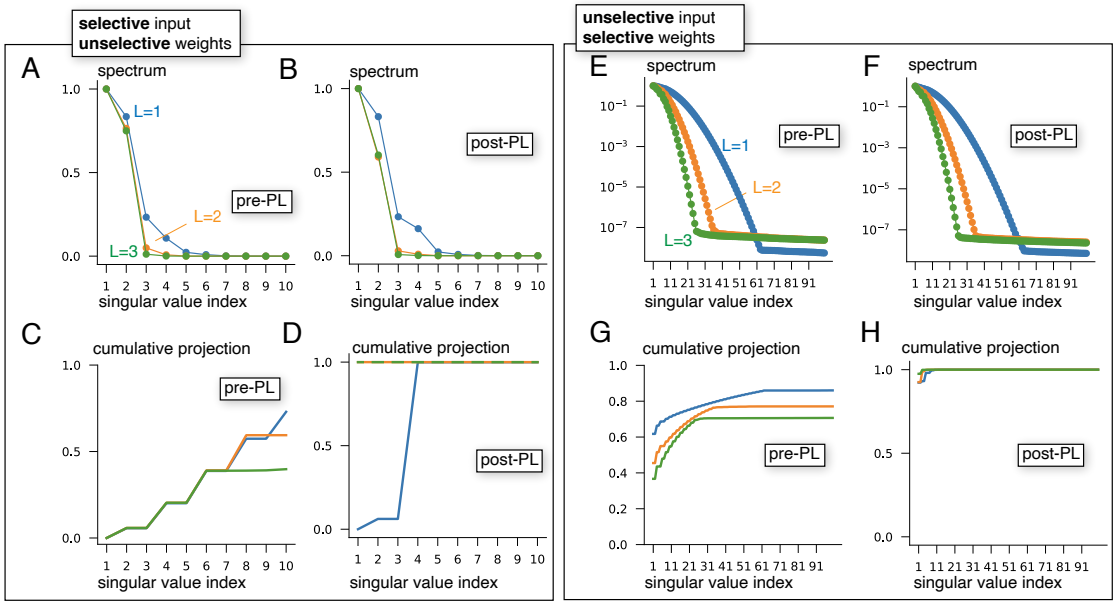


FIG. S2. Low-rankness of weight matrices.

(A) The spectrum of the product of effective weight matrices($\boldsymbol{P}$, see definition in Eq. 13 ) before PL. Each curve corresponds to a network of a different depth. All networks are in the selective-input-unselective-weights regime ($\sigma_s = 0.2, \sigma_w = 0.8$). $\boldsymbol{P}$ is of lower rank for deeper networks.

(B) Same as (A), but for networks after PL. Rank of $\boldsymbol{P}$ for networks post-PL is approximately the same as that in pre-PL networks.

(C) Sum of squared projection of the signal vector ($\boldsymbol{s}$) onto the top $n$ singular vectors of $\boldsymbol{P}$.

(D) Same as (C), but for networks after PL.

(E)(F)(G)(H) Same as (A)(B)(C)(D), respectively, but for networks in the unselective-input-selective-weights regime ($\sigma_s = 1.2, \sigma_w = 0.1$).

*Proof.* Let $\boldsymbol{W}^1 = \sum_i \lambda_i \boldsymbol{\alpha}_i \boldsymbol{\beta}_i^T$ be its singular value decomposition. Then let $\hat{\boldsymbol{\beta}}_i = \boldsymbol{\beta}_i - \gamma_i \boldsymbol{s}$ s.t. $\hat{\boldsymbol{\beta}}_i \perp \boldsymbol{s}$ and $\hat{\boldsymbol{\alpha}}_i = \boldsymbol{\alpha}_i - \epsilon_i \tilde{\boldsymbol{a}}$ s.t. $\hat{\boldsymbol{\alpha}}_i \perp \tilde{\boldsymbol{a}}$. We have

$$\boldsymbol{W}^1 = \sum_i \lambda_i \hat{\boldsymbol{\alpha}}_i \hat{\boldsymbol{\beta}}_i^T + \sum_i \lambda_i \gamma_i \hat{\boldsymbol{\alpha}}_i \boldsymbol{s}^T + \sum_i \lambda_i \gamma_i \epsilon_i \tilde{\boldsymbol{a}} \boldsymbol{s}^T + \sum_i \lambda_i \epsilon_i \tilde{\boldsymbol{a}} \hat{\boldsymbol{\beta}}_i^T. \tag{S4}$$

In order to satisfy $\boldsymbol{W}^{1^T} \tilde{\boldsymbol{a}} = \boldsymbol{s}$, the last term must be zero. In order to satisfy $\boldsymbol{W}^{\mathbf{1}^T} \tilde{\boldsymbol{a}} = \boldsymbol{s}$, the third term must be $\|\tilde{\boldsymbol{a}}\|^{-2} \tilde{\boldsymbol{a}} \boldsymbol{s}^T$.

Denote $\boldsymbol{W}_\perp = \sum_i \lambda_i \hat{\boldsymbol{\alpha}}_i \hat{\boldsymbol{\beta}}_i^T + \sum_i \lambda_i \gamma_i \hat{\boldsymbol{\alpha}}_i \boldsymbol{s}^T$,

$$\boldsymbol{W}^1 = \boldsymbol{W}_\perp + \|\tilde{\boldsymbol{a}}\|^{-2} \tilde{\boldsymbol{a}} \boldsymbol{s}^T. \tag{S5}$$

Since every component in $\boldsymbol{W}_\perp$ has a left vector perpendicular to $\tilde{\boldsymbol{a}}$, $\boldsymbol{W}_\perp^T \tilde{\boldsymbol{a}} = \boldsymbol{0}$.

## III. SOLUTIONS TO MINIMUM PERTURBATION (MP) LEARNING

In this section, we derive the expressions for MP plasticity, which are the solutions to the constrained optimization posed in Eq. 18, for networks of depths 1, 2, 3. These results are derived under the assumption that the same neurons are active in the pre-PL and post-PL network. The derived results are self-consistent with this assumption.

### A. L=1

Define the Lagrangian

$$\mathcal{L} = \|\Delta \boldsymbol{W}^1\|^2 + \|\Delta \boldsymbol{a}\|^2 - \boldsymbol{\lambda}^T \left[ \left(\Delta \boldsymbol{W}^1 + \boldsymbol{W}_0^1\right)^T (\Delta \boldsymbol{a} + \boldsymbol{a}_0) - \boldsymbol{s} \right], \tag{S6}$$

where $\boldsymbol{\lambda}$ is a vector of $N$ Lagrange multipliers. Extremizing the Lagrangian w.r.t. $\Delta \boldsymbol{W}$ and $\Delta \boldsymbol{a}$ yields

$$\Delta \boldsymbol{W}^1 = (\Delta \boldsymbol{a} + \boldsymbol{a}_0)\, \boldsymbol{\lambda}^T \tag{S7}$$

$$\Delta \boldsymbol{a} = \left(\Delta \boldsymbol{W}^1 + \boldsymbol{W}_0^1\right)^T \boldsymbol{\lambda}. \tag{S8}$$

Solve for $\boldsymbol{\lambda}$ to get

$$\boldsymbol{\lambda} = \left[ (1 - \|\boldsymbol{\lambda}\|^2)^{-1} \boldsymbol{W}_0^{1^T} \boldsymbol{W}_0^1 + \|\Delta \boldsymbol{a} + \boldsymbol{a}_0\|^2 \mathbb{I} \right]^{-1} \left[ \boldsymbol{s} - (1 - \|\boldsymbol{\lambda}\|^2)^{-1} \boldsymbol{W}_0^{1^T} \boldsymbol{a}_0 \right], \tag{S9}$$

where $\mathbb{I}$ is the $N$-dimensional identity matrix. Defining scalar order parameters

$$\alpha = (1 - \|\boldsymbol{\lambda}\|^2), \quad \beta = \|\Delta \boldsymbol{a} + \boldsymbol{a}_0\|^2, \tag{S10}$$

we have

$$\boldsymbol{\lambda} = \left[ \alpha^{-1} \boldsymbol{W}_0^{1^T} \boldsymbol{W}_0^1 + \beta \mathbb{I} \right]^{-1} \left[ \boldsymbol{s} - \alpha^{-1} \boldsymbol{W}_0^{1^T} \boldsymbol{a}_0 \right]. \tag{S11}$$

This expression can be plugged back into definitions of $\alpha, \beta$ to obtain two self-consistent equations for $\alpha, \beta$. Values of the order parameters can then be solved numerically, yielding $\alpha^*, \beta^*$. Plugging these back into expressions for $\Delta \boldsymbol{a}, \Delta \boldsymbol{W}^1$ gives the solution.

*a. Assuming fixed* $\boldsymbol{a}$ If we assume a fixed $\boldsymbol{a}$, the MP $\Delta \boldsymbol{W}^1$ can be given in closed form as

$$\Delta \boldsymbol{W}^1 = \frac{\boldsymbol{a}_0}{\|\boldsymbol{a}_0\|^2} \boldsymbol{s}^T - \frac{\boldsymbol{a}_0 \boldsymbol{a}_0^T \boldsymbol{W}_0^1}{\|\boldsymbol{a}_0\|^2}. \tag{S12}$$

### B. L=2

For $L = 2, 3$, applying the same analysis as above shows that changes to $\Delta \boldsymbol{a}$ are negligible. For simplicity, hereafter we assume $\Delta \boldsymbol{a} \approx 0$. This has the advantage of needing only $2L - 2$ order parameters (instead of $2L$).

Define the Lagrangian as

$$\mathcal{L} = \|\Delta \boldsymbol{W}^1\|^2 + \|\Delta \boldsymbol{W}^2\|^2 - \boldsymbol{\lambda}^T \left[ \left(\Delta \boldsymbol{W}^1 + \boldsymbol{W}_0^1\right)^T \left(\Delta \boldsymbol{W}^2 + \boldsymbol{W}_0^2\right)^T \boldsymbol{a}_0 - \boldsymbol{s} \right]. \tag{S13}$$

Extremizing yields

$$\Delta \boldsymbol{W}^1 = (\Delta \boldsymbol{W}^2 + \boldsymbol{W}_0^2)^T \boldsymbol{a}_0 \boldsymbol{\lambda}^T \tag{S14}$$

$$\Delta \boldsymbol{W}^2 = \boldsymbol{a}_0 \boldsymbol{\lambda}^T (\Delta \boldsymbol{W}^1 + \boldsymbol{W}_0^1)^T \tag{S15}$$

Solving for $\Delta \boldsymbol{W}^1, \Delta \boldsymbol{W}^2$ yields

$$\Delta \boldsymbol{W}^1 = (1 - \|\boldsymbol{a}_0\|^2 \|\boldsymbol{\lambda}\|^2)^{-1} \left[ \boldsymbol{W}_0^{2^T} \boldsymbol{a}_0 + \|\boldsymbol{a}_0\|^2 \boldsymbol{W}_0^1 \boldsymbol{\lambda} \right] \boldsymbol{\lambda}^T \tag{S16}$$

$$\Delta \boldsymbol{W}^2 = (1 - \|\boldsymbol{a}_0\|^2 \|\boldsymbol{\lambda}\|^2)^{-1} \boldsymbol{a}_0 \left[ \|\boldsymbol{\lambda}\|^2 \boldsymbol{W}_0^{2^T} \boldsymbol{a}_0 + \boldsymbol{W}_0^1 \boldsymbol{\lambda} \right]^T. \tag{S17}$$

Define scalar order parameters

$$\begin{aligned} u_1 &= \left\| \frac{1}{1 - \|\boldsymbol{a}_0\|^2 v_1} \left[ \boldsymbol{W}_0^{2^T} \boldsymbol{a}_0 + \|\boldsymbol{a}_0\| \boldsymbol{W}_0^1 \boldsymbol{\lambda} \right] \right\|^2 \\ v_1 &= \|\boldsymbol{\lambda}\|^2. \end{aligned} \tag{S18}$$

Plugging expressions for $\Delta \boldsymbol{W}^1, \Delta \boldsymbol{W}^2$ into the constraint equation and solve for $\boldsymbol{\lambda}$ to get

$$\boldsymbol{\lambda} = \left[ \|\boldsymbol{a}_0\|^2 \boldsymbol{W}_0^{1^T} \boldsymbol{W}_0^1 + (1 - \|\boldsymbol{a}_0\|^2 v_1) u_1 \mathbb{I} \right]^{-1} \left[ (1 - \|\boldsymbol{a}_0\|^2 v_1) \boldsymbol{s} - (\boldsymbol{W}_0^2 \boldsymbol{W}_0^1)^T \boldsymbol{a}_0 \right]. \tag{S19}$$

Plugging this expression back into Eqs.S18 yields two self-consistent equations of $u_1, v_1$. Other variables in these equations are all stationary in time. Therefore, one can numerically solve for $u_1, v_1$ to obtain expressions for $\Delta \boldsymbol{W}^1, \Delta \boldsymbol{W}^2$.

### C. L=3

Setting up the Lagrangian and extremizing the variables to get

$$\Delta \boldsymbol{W}^1 = (\Delta \boldsymbol{W}^2 + \boldsymbol{W}_0^2)^T (\Delta \boldsymbol{W}^3 + \boldsymbol{W}_0^3)^T \boldsymbol{a}_0 \boldsymbol{\lambda}^T \tag{S20}$$

$$\Delta \boldsymbol{W}^2 = (\Delta \boldsymbol{W}^3 + \boldsymbol{W}_0^3)^T \boldsymbol{a}_0 \boldsymbol{\lambda}^T (\Delta \boldsymbol{W}^1 + \boldsymbol{W}_0^1)^T \tag{S21}$$

$$\Delta \boldsymbol{W}^3 = \boldsymbol{a}_0 \boldsymbol{\lambda}^T (\Delta \boldsymbol{W}^1 + \boldsymbol{W}_0^1)^T (\Delta \boldsymbol{W}^2 + \boldsymbol{W}_0^2)^T \tag{S22}$$

Solving these equations for $\Delta \boldsymbol{W}^l$ gives ($u_{1,2}, v_{1,2}$ being order parameters defined below)

$$\Delta \boldsymbol{W}^1 = (1 - v_1 u_2)^{-1} \left[ \boldsymbol{W}_0^{2^T} \boldsymbol{U} + u_2 \boldsymbol{W}_0^1 \boldsymbol{\lambda} \right] \boldsymbol{\lambda}^T \tag{S23}$$

$$\Delta \boldsymbol{W}^2 = (1 - v_1 u_2)^{-1} \boldsymbol{U} \left[ v_1 \boldsymbol{U}^T \boldsymbol{W}_0^2 + \boldsymbol{\lambda}^T \boldsymbol{W}_0^{1^T} \right] \tag{S24}$$

$$\Delta \boldsymbol{W}^3 = v_2 \boldsymbol{a}_0 \boldsymbol{U}^T + (1 - v_1 u_2)^{-1} \boldsymbol{a}_0 \left[ v_1 \boldsymbol{U}^T \boldsymbol{W}_0^2 + \boldsymbol{\lambda}^T \boldsymbol{W}_0^{1^T} \right] \boldsymbol{W}_0^{2^T}, \tag{S25}$$

where

$$\boldsymbol{U} = \boldsymbol{Q}^{-1} \left[ \boldsymbol{W}_0^{3^T} \boldsymbol{a}_0 + \|\boldsymbol{a}_0\|^2 (1 - v_1 u_2)^{-1} \boldsymbol{W}_0^2 \boldsymbol{W}_0^1 \boldsymbol{\lambda} \right] \tag{S26}$$

$$\begin{aligned} \boldsymbol{\lambda} = & \left[ u_1 \mathbb{I} + (1 - v_1 u_2)^{-1} u_2 \boldsymbol{W}_0^{1^T} \boldsymbol{W}_0^1 + \|\boldsymbol{a}_0\|^2 (1 - v_1 u_2)^{-2} (\boldsymbol{W}_0^2 \boldsymbol{W}_0^1)^T \boldsymbol{Q}^{-1} \boldsymbol{W}_0^2 \boldsymbol{W}_0^1 \right]^{-1} \\ & \left[ \boldsymbol{s} - (1 - v_1 u_2)^{-1} (\boldsymbol{W}_0^2 \boldsymbol{W}_0^1)^T \boldsymbol{Q}^{-1} (\boldsymbol{W}_0^3)^T \boldsymbol{a}_0 \right], \end{aligned} \tag{S27}$$

and

$$\boldsymbol{Q} = (1 - \|\boldsymbol{a}_0\|^2 v_2) \mathbb{I} - \|\boldsymbol{a}_0\|^2 (1 - v_1 u_2)^{-1} v_1 \boldsymbol{W}_0^{2^T} \boldsymbol{W}_0^2. \tag{S28}$$

We have defined four scalar order parameters to be solved numerically.

$$u_1 = \|(1 - v_1 u_2)^{-1} [\boldsymbol{W}_0^{2^T} \boldsymbol{U} + u_2 \boldsymbol{W}_0^1 \boldsymbol{\lambda}]\|^2 \tag{S29}$$

$$u_2 = \|\boldsymbol{U}\|^2 \tag{S30}$$

$$v_1 = \|\boldsymbol{\lambda}\|^2 \tag{S31}$$

$$v_2 = \|(1 - v_1 u_2)^{-1} [\boldsymbol{W}_0^1 \boldsymbol{\lambda} + v_1 \boldsymbol{W}_0^{2^T} \boldsymbol{U}]\|^2. \tag{S32}$$

### D. Numerical solvers of self-consistent equations

In each case discussed above, we seek to solve $k$ nonlinear equations of $k$ scalar variables numerically. There are many algorithms for this purpose. In general, convergence to the true solution is not guaranteed and depends on initial estimates. To obtain good initial estimates, we used a two-step procedure to solve the equations for each set of network parameters.

In the first step, we used an iterative algorithm defined in Algorithm1.

To aid convergence, we replaced all matrix inversions in the equations with pseudo-inverses (specifically, we only keep the 4 leading singular values and inverse them). These

Initialize estimates for order parameters, $\alpha^0, \beta^0$ ;

$i = 0$ ;

Initialize update factor $\gamma$ ;

Initialize convergence threshold $\epsilon$ ;

**while** $(\alpha^{i+1} - \alpha^i)^2 > \epsilon$ ***or*** $(\beta^{i+1} - \beta^i)^2 > \epsilon$ **do**

  $\alpha^{i+1}, \beta^{i+1} = \gamma * \text{pseudo-self-consistent equations}(\alpha^i, \beta^i) + (1-\gamma) * (\alpha^i, \beta^i)$;

  $i = i + 1$ ;

  **if** $i >$ *MaxIteration* **then**

    **break**

  **end**

**end**

**Algorithm 1:** Algorithm for solving self-consistent equations.

equations are referred to as "pseudo-self-consistent equations". We first used trial-and-error to find good initial estimates for a specific set of network parameters (e.g., $\sigma_s = 0.1, \sigma_w = 1$) and ran the algorithm until convergence. We then considered another pair of parameters that are close to the previous pair (e.g., $\sigma_s = 0.125, \sigma_w = 1$), using *final estimates* for $\sigma_s = 0.1, \sigma_w = 1$ as initial estimates for the new pair. We repeated this procedure recursively to cover all network parameter regimes of interest. After this step, we obtain solutions to the pseudo-self-consistent equations.

For the second step, we used `scipy.optimize.fsolve`, which implements a quasi-Newton method. For each set of network parameters, we used solutions to the pseudo-self-consistent equations as initial estimates for solutions to the true self-consistent equations. Upon convergence, we obtain solutions to the true self-consistent equations.

## IV. CROSS-TASK TRANSFER TEST

To test whether perceptual learning on the $\theta$ discrimination task transfers to a different task, we devised a $\sigma_s$ discrimination task ("width discrimination"). In this task, the network has to discriminate between two close-by values of $\sigma_s$, $\sigma_{s,\text{tr}} \pm \delta\sigma_s$ with the same $\theta$. We ensure that the two tasks have the same difficulty (i.e., the optimal accuracy is the same for both tasks) by choosing values of $\delta\sigma_s$ such that $\|\delta\theta d_\theta \boldsymbol{f}^0\| = \frac{1}{2}\|\boldsymbol{f}^0(\theta_{\text{tr}}; \sigma_s + \delta\sigma_s) - \boldsymbol{f}^0(\theta_{\text{tr}}; \sigma_s - \delta\sigma_s)\|$.

For typical network parameters that we considered, the performance on $\sigma_s$ discrimination is suboptimal prior to training, leaving room for learning. Finally, we assume that the network uses two separate linear readouts from the last layer, $\boldsymbol{a}$ for $\theta$ discrimination and $\boldsymbol{a'}$ for $\sigma_s$ discrimination. Importantly, the averaged stimuli in the $\sigma_s$ discrimination task and the $\theta$ discrimination task are both $\boldsymbol{f}^0(\theta_{\text{tr}})$.

To test for the presence of cross-task transfer, we consider the network with weights before and after MP perceptual learning and tested its width-discrimination performance with $\boldsymbol{a}'$.

Our results suggest that PL for $\theta$ discrimination does not transfer to $\sigma_s$ discrimination. This does not suggest that cross-task transfer *cannot* occur but merely provides an example where it *does not* occur despite extensive representational changes.

## V. STATIONARITY OF ACTIVE NEURONS UNDER GRADIENT DESCENT LEARNING

In this section, $\boldsymbol{W}^l$ refers to the full weight matrix, not the effective matrix $\boldsymbol{W}^l_{\text{eff}}$. Here we show that, under GD dynamics on the unregularized loss function, $\boldsymbol{W}^l\boldsymbol{f}^{l-1}(\theta_{\text{tr}})$ is in fact stationary in time for all $l$. Our overall strategy is to show that for any $l$, $\boldsymbol{W^l W^{l-1}}...\boldsymbol{W^1 f^0}$, which gives a vector of synaptic inputs for neurons in the $l$th layer, is stationary in time. We first show this in networks with one hidden layer before generalizing it to deeper networks.

For brevity, denote

$$h^l_i(\theta_\pm) = [\boldsymbol{W}^l\boldsymbol{x}^{l-1}(\theta_\pm)]_i, \tag{S33}$$

and write $\boldsymbol{f}^0 = \boldsymbol{f}^0(\theta_{\text{tr}})$. Input to the network can be written as

$$\boldsymbol{x}^0(\theta_\pm) = \boldsymbol{f}^0 \pm \delta\theta d_\theta \boldsymbol{f}^0 + \boldsymbol{\epsilon}^0. \tag{S34}$$

### A. Networks with one hidden layer

In this case, output of the network is given by

$$r_\pm \equiv \sum_i a_i \phi\left(\boldsymbol{W}_i^{1^T}\boldsymbol{x}^0(\theta_\pm)\right), \tag{S35}$$

where $\boldsymbol{W}^1_i$ is the $i$th row of $\boldsymbol{W}^1$. Rewrite Eq. 7 as

$$E(\boldsymbol{W}^1, \boldsymbol{a}) = \frac{1}{2}\langle (r_+ - \hat{r_+})^2 + (r_- - \hat{r_-})^2\rangle, \tag{S36}$$

where $\hat{r}_\pm = \boldsymbol{s}^T(\boldsymbol{f}^0 \pm \delta\theta d_\theta \boldsymbol{f}^0 + \boldsymbol{\epsilon}^0)$ and $\langle\cdot\rangle$ denotes average over noise.

Gradient for $\boldsymbol{W}_i^1$ is given by

$$\left\langle \nabla_{\boldsymbol{W}_i^1} E \right\rangle = \frac{1}{2}\left\langle \nabla_{\boldsymbol{W}_i^1}[(r_+ - \hat{r_+})^2] + \nabla_{\boldsymbol{W}_i^1}[(r_- - \hat{r_-}])^2 \right\rangle. \tag{S37}$$

For brevity, we only examine the first term (analysis of the second term is very similar).

$$\frac{1}{2}\left\langle \nabla_{\boldsymbol{W}_i^1}(r_+ - \hat{r_+})^2 \right\rangle = \left\langle (r_+ - \hat{r_+})\left(\frac{dr_+}{dx_i^1(\theta_+)}\right)\phi'(h_i^1(\theta_+))\boldsymbol{x}^0(\theta_+)\right\rangle \tag{S38}$$

$$= a_i\left\langle (r_+ - \hat{r_+})\phi'(h_i^1(\theta_+))\boldsymbol{x}^0(\theta_+)\right\rangle \tag{S39}$$

$$= a_i\left\langle (r_+ - \hat{r_+})\phi'(h_i^1(\theta_+))\right\rangle \boldsymbol{f}^0 + a_i\left\langle (r_+ - \hat{r_+})\phi'(h_i^1(\theta_+))\right\rangle \delta\theta d_\theta \boldsymbol{f}^0 + a_i\left\langle (r_+ - \hat{r_+})\phi'(h_i^1(\theta_+))\boldsymbol{\epsilon}^0\right\rangle. \tag{S40}$$

Applying Stein's lemma to the last term yields

$$\begin{aligned}\frac{1}{2}\langle \nabla_{\boldsymbol{W}_i^1}(r_+ - \hat{r_+})^2\rangle =& a_i\left\langle (r_+ - \hat{r_+})\phi'(h_i^1(\theta_+))\right\rangle \boldsymbol{f}^0 + a_i\left\langle (r_+ - \hat{r_+})\phi'(h_i^1(\theta_+))\right\rangle \delta\theta d_\theta \boldsymbol{f}^0 \\ &+ a_i\sigma^2\left\langle \phi'(h_i^1(\theta_+))\nabla_{\boldsymbol{\epsilon}^0}(r_+ - \hat{r_+})\right\rangle \\ &+ a_i\sigma^2\left\langle (r_+ - \hat{r_+})\phi''(h_i^1(\theta_+))\right\rangle \boldsymbol{W}_i^1.\end{aligned} \tag{S41}$$

Every $\langle\cdot\rangle$ contains the product of two random variables, which can be written as $\langle XY\rangle = \langle X\rangle\langle Y\rangle + cov(X,Y)$. At large $N$, it can be verified that each is dominated by $\langle X\rangle\langle Y\rangle$. Eliminating covariance terms and assuming that $\langle r_+ - \hat{r_+}\rangle = -\langle r_- - \hat{r_-}\rangle$ throughout training, Eq. S37 becomes

$$\langle \nabla_{\boldsymbol{W}_i^1} E\rangle \approx a_i\langle r_+ - \hat{r_+}\rangle\left\langle \phi'(h_i^1(\theta_+)) - \phi'(h_i^1(\theta_-))\right\rangle \boldsymbol{f}^0 \tag{S42}$$

$$+ a_i\langle r_+ - \hat{r_+}\rangle\left\langle \phi'(h_i^1(\theta_+)) + \phi'(h_i^1(\theta_-))\right\rangle \delta\theta d_\theta \boldsymbol{f}^0 \tag{S43}$$

$$+ a_i\sigma^2\left\langle \phi'(h_i^1(\theta_+)) + \phi'(h_i^1(\theta_-))\right\rangle (\boldsymbol{V} - \boldsymbol{s}) \tag{S44}$$

$$+ a_i\sigma^2\langle r_+ - \hat{r_+}\rangle\left\langle \phi''(h_i^1(\theta_+)) - \phi''(h_i^1(\theta_-))\right\rangle \boldsymbol{W}_i^1, \tag{S45}$$

where $\boldsymbol{V} = \boldsymbol{P}^{\boldsymbol{L}^T}\boldsymbol{a}$. Under the linear approximations, we assume $\left\langle \phi'(h_i^l(\theta_+)) - \phi'(h_i^l(\theta_-))\right\rangle = 0$. Thus,

$$\langle \nabla_{\boldsymbol{W}_i^1} E\rangle \approx 2a_i\langle r_+ - \hat{r_+}\rangle\left\langle \phi'(h_i^l(\theta_+))\right\rangle \delta\theta d_\theta \boldsymbol{f}^0 + 2a_i\sigma^2\left\langle \phi'(h_i^l(\theta_+))\right\rangle (\boldsymbol{V} - \boldsymbol{s}). \tag{S46}$$

$\boldsymbol{V}$ is perpendicular to $\boldsymbol{f}^0$ because otherwise the network would be a biased discriminator. In addition, $d_\theta \boldsymbol{f}^0 \perp \boldsymbol{f}^0$. Thus, $\langle \nabla_{\boldsymbol{W}_i^1} E\rangle \perp \boldsymbol{f}^0$ and $\boldsymbol{W}^1\boldsymbol{f}^0$ is stationary in time during learning.

### B. Generalization to deeper networks

To show that the averaged input is stationary in deeper networks, we first derive a general expression for the dynamics of $\boldsymbol{W}^l$ during training in a deep network. Dynamics for $\boldsymbol{W}^l$ can be derived from

$$\langle \nabla_{\boldsymbol{W}_i^l}(r_+ - \hat{r_+})^2\rangle = \Big\langle \frac{dr_+}{dx_i^l(\theta_+)}(r_+ - \hat{r_+})\phi'(h_i^l(\theta_+))\Big\rangle \boldsymbol{f}^{l-1} \tag{S47}$$

$$+ \Big\langle \frac{dr_+}{dx_i^l(\theta_+)}(r_+ - \hat{r_+})\phi'(h_i^l(\theta_+))\Big\rangle \delta\theta d_\theta \boldsymbol{f}^{l-1}$$

$$+ \Big\langle \frac{dr_+}{dx_i^l(\theta_+)}(r_+ - \hat{r_+})\phi'(h_i^l(\theta_+))\boldsymbol{\epsilon}^{l-1}\Big\rangle, \tag{S48}$$

where $\frac{dr_+}{dx_i^l(\theta_+)}$ is a random variable with $O(N^{-1/2})$ mean and fluctuation. Applying Stein's lemma to the last term and eliminating non-leading order terms to get (let $\tilde{a}_i^l = \left\langle \frac{dr_+}{dx_i^l(\theta_+)}\right\rangle$)

$$\langle \nabla_{\boldsymbol{W}_i^l}(r_+ - \hat{r_+})^2\rangle \approx \tilde{a}_i^l \langle r_+ - \hat{r_+}\rangle \langle \phi'(h_i^l(\theta_+))\rangle \boldsymbol{f}^{l-1}$$

$$+ \tilde{a}_i^l \langle r_+ - \hat{r_+}\rangle \langle \phi'(h_i^l(\theta_+))\rangle \delta\theta d_\theta \boldsymbol{f}^{l-1}$$

$$+ \tilde{a}_i^l \langle \phi'(h_i^l(\theta_+))\rangle \left(\boldsymbol{\Sigma}^{l-1}\tilde{\boldsymbol{a}}^{l-1} - \sigma^2 \boldsymbol{P}^{l-1}\boldsymbol{s}\right). \tag{S49}$$

Combine $\langle \nabla_{\boldsymbol{W}_i^l}(r_+ - \hat{r_+})^2\rangle$ and $\langle \nabla_{\boldsymbol{W}_i^l}(r_- - \hat{r_-})^2\rangle$ to get

$$\langle \nabla_{\boldsymbol{W}_i^l} E\rangle = 2\tilde{a}_i^l \big\langle \phi'(h_i^l(\theta_+))\big\rangle \left[\langle r_+ - \hat{r_+}\rangle \delta\theta d_\theta \boldsymbol{f}^{l-1} + \boldsymbol{\Sigma}_{l-1}\tilde{\boldsymbol{a}}^{l-1} - \sigma^2 \boldsymbol{P}^{l-1}\boldsymbol{s}\right]. \tag{S50}$$

Note that under mean-field approximations, $\tilde{\boldsymbol{a}}^{l-1} = (\boldsymbol{W}_{\text{eff}}^l)^T(\boldsymbol{W}_{\text{eff}}^{l+1})^T ... (\boldsymbol{W}_{\text{eff}}^L)^T \boldsymbol{a}$. We have

$$\langle \nabla_{\boldsymbol{W}_i^l} E\rangle = 2\tilde{a}_i^l \big\langle \phi'(h_i^l(\theta_+))\big\rangle \left[\langle r_+ - \hat{r_+}\rangle \delta\theta d_\theta \boldsymbol{f}^{l-1} + \sigma^2 \boldsymbol{P}^{l-1}\boldsymbol{V} - \sigma^2 \boldsymbol{P}^{l-1}\boldsymbol{s}\right]$$

$$= 2\tilde{a}_i^l \big\langle \phi'(h_i^l(\theta_+))\big\rangle \boldsymbol{P}^{l-1} \left[\langle r_+ - \hat{r_+}\rangle \delta\theta d_\theta \boldsymbol{f}^{0} + \sigma^2 \boldsymbol{V} - \sigma^2 \boldsymbol{s}\right]. \tag{S51}$$

We proceed to show that every component of $\langle \nabla_{\boldsymbol{W}_i^l} E\rangle$ is perpendicular to $\boldsymbol{f}^{l-1}$.

First, define a notion of parity for vectors. For an $N$-dimensional vector $\boldsymbol{v}$, it is odd if for all $j$, $v_{N/2-j} = -v_{N/2+j}$; we call it even if $v_{N/2-j} = v_{N/2+j}$. Without loss of generality, we consider the scenario where the input neuron preferring the trained stimulus has index $N/2$. It is easy to see that $d_\theta \boldsymbol{f}^0$ is an odd vector while $\boldsymbol{f}^0$ is an even vector. Furthermore, any odd vector is perpendicular to any even vector.

We make the ansatz that throughout training, $\boldsymbol{V}$ is an odd vector. If $\boldsymbol{V}$ was not odd (that is, it is even or the sum of even and odd vectors), its even component would be perpendicular to the signal $d_\theta \boldsymbol{f}^0$. This component would therefore be suboptimal because it

would contribute to noise without contributing to signal. Unlike other sources of suboptimality discussed in the main text, this component can easy be removed by making $\boldsymbol{a}$ an odd vector. Since we optimize $\boldsymbol{a}$ before learning, this component does not exist at the beginning of learning.

Since pre-PL weight matrices are circulant, it is easy to verify that they have the following property: if $\boldsymbol{v}$ is an odd/even vector, then $\boldsymbol{W}^l\boldsymbol{v}$ and $(\boldsymbol{W}^l)^T\boldsymbol{v}$ are also odd/even. Further, it can be verified that pre-PL effective weight matrices have the same property. We say that these matrices *preserve vector parity*. Product of matrices preserving vector parity preserves parity itself.

We now show that throughout learning, weight matrices and effective matrices preserve vector parity. Assume that at time $t$, weight matrices still preserve vector parity. Note that the gradient is a rank-1 matrix. The right vector of the gradient is an odd vector, since $d_\theta \boldsymbol{f}^0, \boldsymbol{V}, \boldsymbol{s}$ are all odd vectors, and the product matrix preserves parity. The left vector, which can be written as the element-wise product between an odd vector $\tilde{\boldsymbol{a}}^l$ and an even vector $\left\langle \phi'(\boldsymbol{h}^l(\theta_+)) \right\rangle$, is an odd vector. Therefore, for a small $\tau$, this matrix at time $t+\tau$ can be written as

$$\boldsymbol{W}^l_{t+\tau} = \boldsymbol{W}^l_t + \tau \boldsymbol{o}_1 \boldsymbol{o}_2^T, \tag{S52}$$

where $\boldsymbol{o}_{1,2}$ are odd vectors. This new matrix will again preserve vector parity since (letting $\boldsymbol{o}$ denote any odd vector)

$$\boldsymbol{W}^l_{t+\tau}\boldsymbol{o} = \boldsymbol{W}^l_t\boldsymbol{o} + \tau \boldsymbol{o}_1 \boldsymbol{o}_2^T \boldsymbol{o}, \tag{S53}$$

is a sum of two odd vectors and therefore still an odd vector and (letting $\boldsymbol{e}$ denote any even vector)

$$\boldsymbol{W}^l_{t+\tau}\boldsymbol{e} = \boldsymbol{W}^l_t\boldsymbol{e} + \tau \boldsymbol{o}_1 \boldsymbol{o}_2^T \boldsymbol{e} = \boldsymbol{W}^l_t\boldsymbol{e}, \tag{S54}$$

which is even. Since weight matrices preserve parity at initialization, we can show by induction that they do so throughout learning.

We now return to Eq. S51. Since weight matrices at all times preserve vector parity, the mean response in layer $l-1$, $\boldsymbol{f}^{l-1} = \boldsymbol{W}^{l-1}_{\text{eff}}\boldsymbol{W}^{l-2}_{\text{eff}}...\boldsymbol{W}^{1}_{\text{eff}}\boldsymbol{f}^0$ is always even.

In addition, $\langle \nabla_{\boldsymbol{W}^l_i} E \rangle$ is always odd because $\boldsymbol{P}^{l-1}$ always preserve parity. Therefore, $\langle \nabla_{\boldsymbol{W}^l_i} E \rangle$ is always perpendicular to $\boldsymbol{f}^{l-1}(\theta_{\text{tr}})$.

| Variable | Value | Comments |
|---|---|---|
| [0.5ex] Number of neurons per layer ($N$) | 1000 | All layers have the same number. |
| Learning rate ($\eta$) | 0.001 | |
| Input noise variance ($\sigma^2$) | 0.01 | |
| $\delta\theta$ (in rads) | $\approx 0.0009$ | Adjusted so that signal-to-noise ratio in input is 1. |
| Minibatch size | 50 | |

TABLE S1. Default numerical hyperparameters.

## VI. NUMERICAL SIMULATIONS OF GRADIENT DESCENT LEARNING

Gradient descent with minibatches was used for all weights in the network ($\boldsymbol{W}^l$ and $\boldsymbol{a}$), implemented with `pytorch`. The algorithm is terminated after the error estimated with $50,000$ examples is less or equal to the optimal level for more than 5 times. Default hyperparameters used in simulations are tabulated in Table S1. We chose relatively small $\delta\theta$ and $\sigma^2$ so that the large $N$ effects are already apparent with $N = 1000$, conserving computational resources. Python codes are available at `https://github.com/hzshan/perceptual_learning`.

## VII. DETAILED COMPARISONS WITH EXPERIMENTAL EVIDENCE

Our finding that PL is driven by improved sensory coding is consistent with the observed PL-induced changes to sensory representations in several electrophysiological experiments [1–8, 39–41] and functional imaging studies [50–53] across different model systems and tasks. Some of these studies reported representational changes that are closely related to behavioral improvements [6–8, 39], consistent with our predictions. Furthermore, we predict that for a fine discrimination task, processing between layers is approximately linear and thus all information present in an area is accessible to a linear decoder, as reported by [6]. However, our theory is inconsistent with studies that found little to no neural plasticity correlates of PL in sensory areas [2, 11, 20, 54]. Such inconsistency may arise from different task conditions and analysis methods. First, while MP synaptic changes are stronger in early layers (**Fig. 3A**), changes to neuronal tuning properties are predicted to be more prominent in higher layers (**Fig. 4**). This is consistent with experiments reporting bigger changes in monkey V4

than those in V1 and V2 following orientation discrimination PL [1–4]. This may explain why some experiments failed to find significant changes in early sensory areas [2, 54]. Second, our theory predicts that PL-induced changes to tuning curves are localized near $\theta_{\mathrm{tr}}$ and may cause some tuning curves to lose their pre-PL bell shapes. Thus, studies (e.g., [2, 54]) that excluded non-bell-shaped neurons from analysis or fitted bell-shaped functions to tuning curves may fail to detect these localized changes. Third, for motion direction discrimination in moving random dot patterns [11, 20], the relevant primary sensory layer may be MT rather than neurons in V1 with smaller receptive fields. Under this interpretation, changes in the readout layer from MT should be sufficient to yield an optimal performance.

The *types* of changes to sensory representations predicted by MP learning are largely consistent with experimental observations. At a single-cell level, MP learning causes localized sharpening of tuning curves and thus amplified signal strength, as observed in experiments [1, 3, 4, 7, 39–41]. In addition, PL is predicted to decrease the number of neurons preferring $\theta_{\mathrm{tr}}$, as reported in [2–4]. On a population level, MP learning decreased mean noise correlation (Fig. S4), consistent with findings from simultaneous recordings of multiple units [5–8]. We predict that mean firing rates of neurons responding to $\theta_{\mathrm{tr}}$ will be unaffected by learning, consistent with [5, 11, 41, 52, 54], while some others found increased [3, 50, 51, 55–59] or decreased activation [2, 60]. While these inconsistencies may arise from differences in tasks and setups, our theory indicates that increased/decreased activation does not play a causal role in PL [6, 19].

In terms of psychophysics, our theory predicts a rich pattern of cross-stimuli transfer. While PL transfers to stimuli highly similar to $\theta_{\mathrm{tr}}$ as expected, it causes performance for intermediate stimuli ("proximal", **Fig. 5**) to drop below pre-PL levels. Indeed, some experiments report worse-than-baseline performance when subjects are tested on untrained stimuli following PL [61–63]. In addition, PL transfers to stimuli further away ("distal") from $\theta_{\mathrm{tr}}$. A more systematic examination of how observed cross-stimuli transfer depends on stimulus similarity can further test our theory. In deriving our results, we have assumed a high-precision scenario where both signal and noise are small. A sufficiently large signal (i.e., low precision) and/or noise invalidate our assumption that a fixed subset of neurons are active throughout learning as well as during responses to the trained stimuli. PL under such conditions likely involves a broader subset of neurons and weights, potentially explaining why it leads to broader cross-stimuli transfer than high-precision PL does [64].

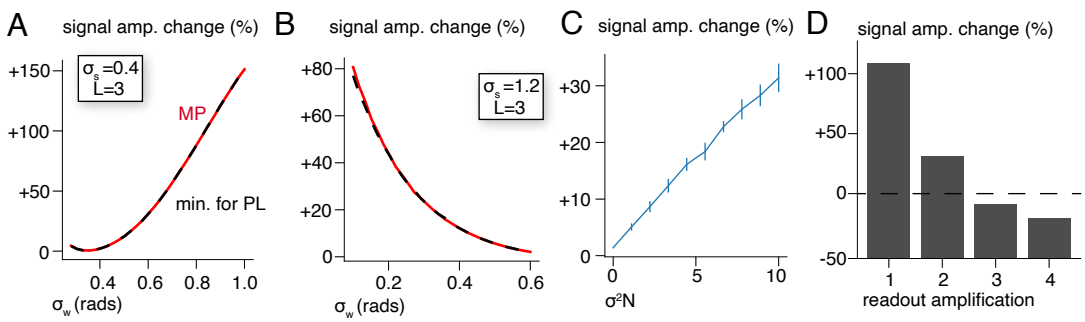


FIG. S3. (A,B) comparison between the MP-induced signal amplification and the minimal signal amplification required for PL. (A) shows results in the selective-input-unselective-weights regime; (B) shows results in the unselective-input-selective-weights regime. (C) Average signal amplification induced by "soft" MP learning, where weights are allowed to fluctuate around the MP weights. $\sigma^2$: magnitude of fluctuation. Errorbars show standard deviations over 10 independent samples. Results are taken from a 1-layer network with $\sigma_s = 0.4, \sigma_w = 1.0$. (D) Signal amplification induced by MP learning if we amplify the pre-PL readout weights. Results are taken from the last layer in a network with $\sigma_s = 0.4, \sigma_w = 1.0, L = 2$.

## VIII. CIRCULANT SOLUTIONS TO PERCEPTUAL LEARNING

For networks with one layer and a fixed $\boldsymbol{a}$, we seek a circulant $\boldsymbol{W}^1_{\text{circ}}$ such that ${\boldsymbol{W}^1_{\text{circ}}}^T \boldsymbol{a} = \boldsymbol{s}$. Note that all circulant matrices can be diagonalized as $\boldsymbol{W}^1_{\text{circ}} = \boldsymbol{F}\boldsymbol{\Lambda}_W\boldsymbol{F}^T$, where singular vectors in $\boldsymbol{F}$ are the Fourier bases. We then use

$$(\Lambda_W)_{ii} = \frac{(\boldsymbol{F}^T\boldsymbol{s})_i}{(\boldsymbol{F}^T\boldsymbol{a})_i + \epsilon_{\text{circ}}}, \tag{S55}$$

where the regularizer $\epsilon_{\text{circ}}$ is chosen to be as large as possible without significantly increasing $\|{\boldsymbol{W}^1_{\text{circ}}}^T\boldsymbol{a} - \boldsymbol{s}\|$. We found that such $\boldsymbol{W}^1_{\text{circ}}$, computed numerically, leads to multi-modal tuning curves, which are unrealistic results (data not shown).

## IX. IS SIGNAL AMPLIFICATION REQUIRED FOR PL?

### A. The minimal signal amplitude in the last layer required for PL

Both MP learning and gradient-descent learning lead to negligible changes to the readout vector $\boldsymbol{a}$. Assuming the readout to be fixed, we ask whether it is possible to complete

PL without increasing the signal amplitude in last-layer representations by computing the minimal signal amplitude in the last layer required for PL.

For PL, the necessary and sufficient condition in Eq. 16 translates to analogous conditions on the post PL value of the product matrix $\boldsymbol{P}^L$, i.e., $\boldsymbol{P}^L = \boldsymbol{u}\boldsymbol{s}^T + \boldsymbol{P}_\perp$, $\boldsymbol{P}_\perp \boldsymbol{s} = \boldsymbol{0}, \boldsymbol{P}_\perp^T \boldsymbol{a} = \boldsymbol{0}$. In addition, in order to minimize the loss function(Eq. 7), $\boldsymbol{u}^T\boldsymbol{a} = 1$. The squared signal amplitude in the last layer is given by

$$\|d_\theta \boldsymbol{f}^L\big|_{\theta_{\rm tr}}\|^2 = \|\boldsymbol{P}^L d_\theta \boldsymbol{f}^0\big|_{\theta_{\rm tr}}\|^2 = \|\boldsymbol{u}\|^2 \|d_\theta \boldsymbol{f}^0\big|_{\theta_{\rm tr}}\|^2 + \|\boldsymbol{P}_\perp d_\theta \boldsymbol{f}^0\|^2. \tag{S56}$$

This is minimized (under the constraint that $\boldsymbol{u}^T\boldsymbol{a} = 1$) by $\boldsymbol{u} = \|\boldsymbol{a}\|^{-2}\boldsymbol{a}$ and $\boldsymbol{P}_\perp = \boldsymbol{0}$. Thus, the minimal post-PL signal amplitude is

$$\min \|d_\theta \boldsymbol{f}^L\big|_{\theta_{\rm tr}}\| = \|\boldsymbol{a}\|^{-1} \|d_\theta \boldsymbol{f}^0\big|_{\theta_{\rm tr}}\|. \tag{S57}$$

As shown in Fig. S3A,B, the minimal post-PL signal amplitude is larger than the pre-PL one across parameters. Thus, signal amplification is indeed necessary for PL, assuming a fixed $\boldsymbol{a}$. We also note that the signal amplitude after MP learning is close to the minimal level.

### B. PL with an amplified readout vector

A reasonable extension of our model is to consider the case where the direction of the readout vector stays the same but its amplitude is increased, namely ($\boldsymbol{a}_{\rm post} = (1+c)\boldsymbol{a}_{\rm pre}$). We computed MP changes to weight matrices for several values of $c$ (for networks with $L = 2$) and find that if $c$ is sufficiently large, MP learning may even lead to a *decreased* signal amplitude (Fig. S3C).

### C. Learning under a soft MP constraint

Compared to a non-MP post-PL solution to PL, does MP learning lead to smaller signal amplitudes? We consider a "soft" MP constraint where changes to the weights fluctuate in the space of solutions around MP changes. Concretely, for networks with one-layer and a fixed $\boldsymbol{a}$, we sample solution $\boldsymbol{W}^1$ with

$$\boldsymbol{W}^1_{\text{soft MP}} = \boldsymbol{W}^1_{\rm pre} + \Delta \boldsymbol{W}^1_{\rm MP} + \left(\mathbb{I} - \frac{\boldsymbol{a}\boldsymbol{a}^T}{\|\boldsymbol{a}\|^2}\right)\boldsymbol{E}, \tag{S58}$$

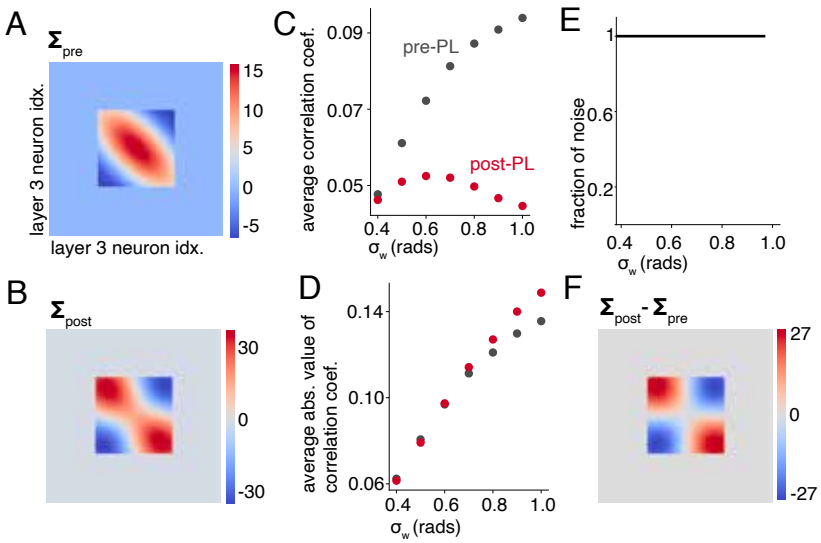


FIG. S4. Structure of noise correlation in layer 3 of a $L = 3$ network when $\theta_{\rm tr}$ is presented. (A) Normalized* pre-PL noise covariance matrix. (B) Post-PL covariance. (C) Averaged pair-wise Pearson's correlation coefficients before (red) and after PL (black), for different $\sigma_w$. (D) Same as C, averaged absolute values of coefficients are shown. (E) Fraction of PL-induced changes to covariance ($\boldsymbol{\Sigma}_{\rm post} - \boldsymbol{\Sigma}_{\rm pre}$) that project on the signal direction. (F) Visualization of PL-induced changes to covariance.

*Covariance matrices are multiplied by $N/\sigma^2 \|\boldsymbol{f}^0\|^2 \|\boldsymbol{f}^3\|^{-2}$, where $\boldsymbol{f}^0, \boldsymbol{f}^3$ are noise-averaged population response vectors in the input layer and layer 3, respectively. This scaling makes each element of the matrix $O(1)$ and adjusts for different activity levels in different layers.

where $E_{ij} \sim \mathcal{N}(0, \sigma^2_{\text{soft MP}}/N)$ and $\Delta \boldsymbol{W}^1_{\rm MP} = \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|^2}\boldsymbol{s}^T - \frac{\boldsymbol{a}\boldsymbol{a}^T \boldsymbol{W}^1_{\rm pre}}{\|\boldsymbol{a}\|^2}$ is the MP changes to $\boldsymbol{W}^1$ (derived under S.M. III).

Every sampled $\boldsymbol{W}^1$ solves PL. We computed the post-PL signal amplitude for various $\sigma^2_{\text{soft MP}}$. $\sigma^2_{\text{soft MP}} = 0$ is the MP changes (Fig. S3D). We found that MP learning does indeed lead to a smaller signal amplitude than the average "soft" MP learning.

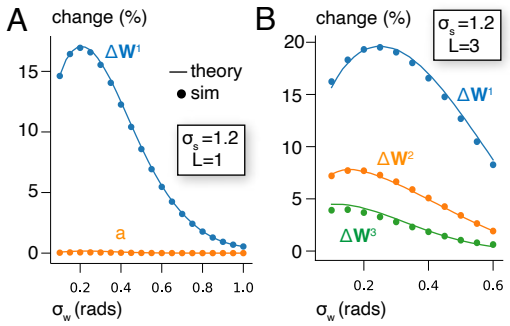


FIG. S5. Comparing magnitude of MP learning changes ("theory") and changes induced by slow gradient descent ("sim") in the unselective-input-selective-weights regime. (A) L=1 network. (B) L=3 network. Learning rate is $10^{-3}$.