nature neuroscience

Article

https://doi.org/10.1038/s41593-023-01382-9

Organizing memories for generalization in complementary learning systems

1 The Teacher-Student-Notebook framework

We consider a setting in which an agent experiences examples of a relationship in the environment in the form of P pairs of activity patterns $\{x^\mu, y^\mu\}, \mu = 1, \cdots, P$. Given the *input* activity vector of dimension N, $x^\mu \in R^N$, the agent must both memorize the associated scalar *output* activity y^μ and develop the ability to predict outputs for new inputs. For example, x^μ might represent activity in visual cortex in response to an event like seeing a bird, and y^μ might represent activity in a higher association cortex derived from a caregiver's speech: "Look, a bird!" The agent wishes to memorize what was seen and said in this specific instance, as well as learn what birds look like more generally.

For any given event, there will be many such relationships to learn, which collectively encode many diverse features in the environment. For instance, while viewing the bird, other neural circuits may encode the spatial location of the event, the time of day, other objects in the scene, and so on. Our model first considers just one of these relationships, and we return to having multiple relationships in the later sections of this document. Our theory contains three neural network components, and we refer to it eponymously as the teacher-student-notebook framework. We now describe each of these components in sequence.

Teacher Network. The ground truth relationship between inputs and output is represented by a *teacher* network. It generates input-output pairs by first drawing an input vector, x, in which each element is *i.i.d.* normal with variance $1/N$, i.e., $x_i \sim \mathcal{N}(0, 1/N),\ i = 1, \cdots, N$. Thus, the norm of the input vector is one in expectation. Next, the teacher labels this input according to the rule

$$y = \bar{w}x + \epsilon, \tag{1}$$

where $\bar{w} \in R^{1\times N}$ are the teacher weights, and ϵ is the teacher output noise. In words, the teacher is simply a shallow linear network with output noise. We take the teacher weights to be *i.i.d.* normal with variance $\sigma^2_{\bar{w}}$, $\bar{w}_i \sim \mathcal{N}(0, \sigma^2_{\bar{w}}), i = 1, \cdots, N$, and the output noise is *i.i.d.* normal with variance σ^2_ϵ. Note that the noise varies across examples, but the weights are fixed for all examples.

A key parameter of this setting is the *signal-to-noise* ratio (SNR),

$$\mathcal{S} = \frac{\langle (\bar{w}x)^2 \rangle_{x,\bar{w}}}{\langle \epsilon^2 \rangle_\epsilon} = \frac{\sigma^2_{\bar{w}}}{\sigma^2_\epsilon}, \tag{2}$$

where $\langle \cdot \rangle_{x,\bar{w}}$ denotes the average over input patterns and teacher weights, and $\langle \cdot \rangle_\epsilon$ denotes the average over noise. This ratio measures the extent to which the teacher's output follows a systematic mapping between input and output. To fix the scale across different teachers, we consider the case where the variance of the teacher's output is one,

$$\langle y^2 \rangle_{x,\bar{w},\epsilon} = \sigma^2_{\bar{w}} + \sigma^2_\epsilon = 1, \tag{3}$$

such that the SNR fixes the variances,

$$\sigma_{\bar{w}}^2 = \tfrac{\mathcal{S}}{1+\mathcal{S}}, \tag{4}$$

$$\sigma_{\epsilon}^2 = \tfrac{1}{1+\mathcal{S}}. \tag{5}$$

Conceptually, the teacher provides a generative model of the environment. We emphasize that taking the teacher to be a simple neural network does not reflect an assumption that the real environment is either simple or a neural network. Rather, the teacher network can be thought of as containing the optimal synaptic weights for approximating the true generative model of the environment, which may reflect diverse causal processes, from the physics of the world to the neural circuits that generate input and output activity patterns. In this sense, the teacher is a useful abstraction, not a mechanistic theory of the environment. We discuss further interpretations of the teacher in Section 10.

Student Network. The goal of the *student* network is to learn to approximate the relationship defined by the teacher. Here we take the student to have the same architecture as the teacher, that is, it is a shallow linear network that receives an N-dimensional input, x, and produces a predicted output, $\hat{y}$, according to

$$\hat{y} = wx, \tag{6}$$

where $w \in R^{1\times N}$ are the student weights. These weights are learned using gradient descent on a loss function $\mathcal{L}(w)$

$$\tau \frac{d}{dt} w = -\frac{\partial}{\partial w}\mathcal{L}(w), \tag{7}$$

here formulated in continuous time (also known as gradient flow) with time constant τ. We take the loss function to be the mean squared error over the example patterns,

$$\mathcal{L}(w) = \frac{1}{P}\sum_{\mu=1}^{P}(y^\mu - \hat{y}^\mu)^2, \tag{8}$$

where $\mu = 1, \cdots, P$ indexes examples, y^μ is the scalar target output, and $\hat{y}^\mu$ is the scalar output prediction in response to input vector x^μ. As described in more detail subsequently, the target patterns that drive learning can have multiple sources–they may come directly from the teacher or from a memory of past examples.

The performance of the student can be measured in two ways. First, its predictions can be evaluated on the specific examples $\mu = 1, \cdots, P$ seen during training, which we refer to as the *memory* error $\mathcal{E}_m$ (also known as the *training error* in machine learning contexts),

$$\mathcal{E}_m = \frac{1}{P}\sum_{\mu=1}^{P}(y^\mu - \hat{y}^\mu)^2, \tag{9}$$

where here we have not indicated the time dependence of $\hat{y}^\mu$ for notational simplicity. Second, the student's predictions can be evaluated on novel input-output pairs drawn from the teacher, which we refer to as the *generalization* error $\mathcal{E}_g$ (also known as the *test error* in machine learning contexts),

$$\begin{aligned} \mathcal{E}_g &= \langle (y - \hat{y})^2 \rangle_{x,\epsilon}, \\ &= \langle (\bar{w}x + \epsilon - wx)^2 \rangle_{x,\epsilon} \\ &= \left\langle ((\bar{w} - w)x + \epsilon)^2 \right\rangle_{x,\epsilon} \\ &= \frac{1}{N} ||\bar{w} - w||_2^2 + \sigma_\epsilon^2, \end{aligned} \tag{10}$$

where $\langle \cdot \rangle_{x,\epsilon}$ denotes the average over the teacher input distribution and output noise distribution, and we have used the fact that these distributions are independent.

Notebook Network. Finally, the job of the *notebook* network is to faithfully memorize experienced patterns as attractors of neural network dynamics, making possible later recall and replay. We consider a notebook of M neurons recurrently connected through the $M \times M$ weight matrix J. The activity in this network is binary, $h \in \{0, 1\}^M$, and evolves according to

$$h(u) = f(Jh(u - 1) - \theta), \tag{11}$$

where f is the Heaviside step function, u denotes discrete time steps of synchronous activity propagation, and θ is a threshold that can have a fixed value or be dynamically adapted to maintain a desired sparsity of activity (as described subsequently).

The notebook represents memorized patterns as binary vectors of zeros and ones by embedding these vectors as fixed points of the dynamics in Eq. (11). In particular, to store input-output pairs, $\{x^\mu, y^\mu\},\ \mu = 1, \cdots, P$, the notebook first chooses P binary (0/1) vectors of length M, uniformly at random from the set of vectors with sparsity a (i.e., with exactly aM nonzero entries). These binary patterns of activity in the notebook act as distinctive neural codes to be associated with each pattern, and this notebook code is sometimes referred to as a memory index.

Stacking the binary patterns into the columns of the $M \times P$ matrix ξ, and similarly stacking the input and output patterns into the $N \times P$ and $1 \times P$ matrices X and Y respectively, the weights within the notebook and between the notebook and student are given through a Hebbian scheme,

$$
\begin{aligned}
J_{ij} &= \begin{cases} \left(\frac{(\xi-a)(\xi-a)^T}{Ma(1-a)} - \frac{\gamma}{Ma} \right)_{ij} & \text{for } i \neq j \\ 0 \quad \text{otherwise} \end{cases}, & (12) \\
U^{S_x \to N} &= (\xi - a) X^T, & (13) \\
U^{S_y \to N} &= (\xi - a) Y^T, & (14) \\
V^{N \to S_x} &= \frac{X(\xi^T - a)}{Ma(1-a)}, & (15) \\
V^{N \to S_y} &= \frac{Y(\xi^T - a)}{Ma(1-a)}. & (16)
\end{aligned}
$$

Here $U^{S_x \to N} \in R^{M \times N}$ and $U^{S_y \to N} \in R^{M \times 1}$ map from the student inputs x and output y to the notebook activity h, and the matrices $V^{N \to S_x} \in R^{N \times M}$ and $V^{N \to S_y} \in R^{1 \times M}$ perform the reverse mapping from the notebook activity back to the student input and output. For simplicity and tractability, we take all student neurons to be linear. The parameter γ in Eqn. 12 implements global all-to-all inhibition, which causes activity that is far from stored patterns to decay to a silent state [12]. In simulations, we take $\gamma = 0.6$, which lies in the range theoretically derived to stably store the intended patterns without spurious attractors in this model [12]. These pathways allow diverse interactions between notebook and student, and we describe a number of specific interaction patterns subsequently.

The mean subtraction and normalization in these updates have been chosen to aid performance, as derived subsequently in Section 5.1 for connections from notebook neurons. In essence, the notebook generates distinct, pattern-separated activity patterns, stabilizes these as attractors of its recurrent dynamics, and links these bidirectionally to the student's input and output neurons to facilitate later replay and reactivation.

2 Learning setting

The teacher-student-notebook framework can allow for diverse learning settings in which examples from the teacher arrive at different times and in different quantities. Here we usually characterize memorization and generalization performance in a simple setting: the *single-batch, high-dimensional* regime. That is, we consider a scenario where an organism receives P training experiences up front in a short time window, and memory and generalization performance are evaluated subsequently over longer periods of time. For instance, a human subject might learn a task in a single hour long session but then be tested after several weeks' delay, or a rodent might perform several trials in a water maze on one day and be tested on the next. In our framework, these P experiences are drawn *i.i.d.* from the teacher and constitute one single batch for learning and consolidation. For convenience, we can collect this batch of samples into the $N \times P$ matrix X with columns $x^\mu,\ \mu = 1, \cdots, P$, and the $1 \times P$ row vector Y with elements $y^\mu,\ \mu = 1, \cdots, P$.

Given abundant training experience ($P >> N$), many different learning schemes can converge to similar performance. However, real world learning is often severely data limited. Animals may receive only one or two foot shocks. A human subject may need to learn a new visual discrimination (possibly dependent on millions of pixels) from just a few blocks of training trials. Real world settings therefore place a premium on learning from limited experience. Moreover, neuronal networks in the brain are typically very large relative to the amount of training experience. Even a simple visual discrimination may engage a network of millions or billions of neurons interconnected by billions or trillions of adjustable synapses. To address this large network, limited data setting, we analyze the *high-dimensional* regime, in which the size of the student network and the number of training samples both tend to infinity ($N \to \infty, P \to \infty$), but their ratio $\alpha = P/N$ remains finite. The *load parameter* α is a key parameter of our setting, and it measures the amount of experience relative to the number of tunable synapses in the student network. For $\alpha < 1$, the network has more tunable parameters than training experiences, allowing analysis of highly overparametrized learning settings. For $\alpha >> 1$, the network has many more training experience than tunable parameters, reflecting the more standard classical regime of statistics.

While in this paper we emphasize this single-batch, high-dimensional learning setting, future work in the teacher-student-notebook framework could investigate more complex scenarios where examples continue to arrive over time.

3 Interaction policies & performance

The single-batch learning setting still allows diverse possible interaction policies between the modules in the teacher-student-notebook framework. These interaction policies specify which modules undergo learning, from what activity patterns (e.g., from the teacher, or from replay from the student), and which modules are used to answer queries for new experiences. We consider four interaction policies, meant to typify common approaches to learning and consolidation.

Online Student. Only the student is trained, without any replay. Each example drives one update of error-corrective learning and is never revisited. This strategy provides a reference point for performance of a system based on online gradient descent learning.

Online Notebook. Only the notebook is used. Each example is stored in the notebook with Hebbian updates, and predictions for novel inputs are generated using the notebook only. This strategy provides a reference point for performance of a system based on Hebbian memorization, without replay-guided learning.

Memory-optimized Replay. This strategy initially stores all experiences in the notebook and trains the student using notebook-driven reactivations until the student has fully memorized all examples. This is similar to the standard theory of systems consolidation.

Generalization-optimized Replay. This novel strategy, proposed in this work, initially stores all experiences in the notebook but only trains the student using notebook-driven reactivations as long as generalization performance improves.

The next four sections of the supplement sequentially characterize the memorization and generalization performance of each of these interaction policies.

4 Online Student Policy

In the online student policy, each example $x^\mu \in R^N, \mu = 1, \cdots, P$, in the batch is visited in order and a single step of error corrective gradient descent learning is applied with an example-dependent learning rate η^μ. In this section we characterize the expected generalization error dynamics under this scheme; to ensure a robust normative comparison to other policies, we derive the globally optimal learning rate function that maximizes generalization performance after all updates.

4.1 Generalization dynamics with example-dependent learning rate

Upon receiving each example $\mu = 1, \cdots, P$, the student weights are updated according to

$$w^{\mu+1} = w^\mu + \eta^\mu e^\mu x^{\mu^T}, \tag{17}$$

where $w^{\mu+1}$ is the weight vector resulting from the μth learning step, η^μ is the learning rate of this step, x^μ is the μth input example, and $e^\mu = y^\mu - \hat{y}^\mu$ is the error between the network's output and the target output for this example. We assume that the initial weights are zero, $w^1 = 0$. Using the teacher model, $y^\mu = \bar{w}x^\mu + \epsilon^\mu$, we have

$$\begin{aligned} w^{\mu+1} &= w^\mu + \eta^\mu \left(\bar{w}x^\mu + \epsilon^\mu - w^\mu x^\mu\right) x^{\mu^T} \\ &= w^\mu + \eta^\mu \left(\bar{w} - w^\mu\right) x^\mu x^{\mu^T} + \eta^\mu \epsilon^\mu x^{\mu^T}. \end{aligned} \tag{18}$$

In contrast to Eqn. (10), which expresses the generalization error $\mathcal{E}_g$ for a specific student and teacher, here we ask what the expected generalization error is for a randomly drawn teacher by averaging over the teacher weight distribution as well. That is, we track the expected generalization error $E_g = \langle \mathcal{E}_g \rangle_{\bar{w}}$, where the average is over the teacher weight distribution. In the high-dimensional regime, the generalization error is self-averaging, such that any specific realization closely tracks this expected generalization error, as will be verified by a close match between single simulations and the average dynamics we derive. The expected generalization error before example μ is

$$\begin{aligned}
E_g[\mu] &= \langle (y-\hat{y})^2 \rangle_{\bar{w},x,\epsilon,x_{1:\mu-1},\epsilon_{1:\mu-1}} \\
&= \langle \left(\left(\bar{w}-w^{\mu}\right) x + \epsilon \right)^2 \rangle_{\bar{w},x,\epsilon,x_{1:\mu-1},\epsilon_{1:\mu-1}} \\
&= \langle \left(\bar{w}-w^{\mu}\right) x x^T \left(\bar{w}-w^{\mu}\right)^T \rangle_{\bar{w},x,x_{1:\mu-1},\epsilon_{1:\mu-1}} \\
&\quad +2\langle \epsilon(\bar{w}-w^{\mu})x \rangle_{\bar{w},x,\epsilon,x_{1:\mu-1},\epsilon_{1:\mu-1}} + \langle \epsilon^2 \rangle_{\epsilon} \\
&= \frac{1}{N}\langle \|\bar{w}-w^{\mu}\|^2 \rangle_{\bar{w},x_{1:\mu-1},\epsilon_{1:\mu-1}} + \sigma_e^2, \qquad (19)
\end{aligned}$$

$x_{1:\mu}$ and $\epsilon_{1:\mu}$ denote history of training patterns and corresponding additive noise, respectively. We used that $\left(\bar{w}-w^{\mu}\right)x = x^T\left(\bar{w}-w^{\mu}\right)^T$ is a scalar, that ϵ is zero mean and independent of all other terms, and that x is multivariate normal with covariance matrix $\langle xx^T \rangle = \frac{1}{N}I$. Similarly, note that after example μ, the expected generalization error becomes

$$E_g[\mu+1] = \frac{1}{N}\langle \left\|\bar{w}-w^{\mu+1}\right\|^2 \rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} + \sigma_e^2. \qquad (20)$$

Substituting in Eqn. 18, we have

$$\begin{aligned}
E_g[\mu+1] &= \frac{1}{N}\left\langle \left\| \bar{w}-w^{\mu}-\eta^{\mu}(\bar{w}-w^{\mu})x^{\mu}x^{\mu^T} - \eta^{\mu}\epsilon^{\mu}x^{\mu^T} \right\|^2 \right\rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} + \sigma_e^2, \\
&= \frac{1}{N}\langle \left((\bar{w}-w^{\mu})(1-\eta^{\mu}x^{\mu}x^{\mu^T}) - \eta^{\mu}\epsilon^{\mu}x^{\mu^T} \right) \\
&\quad \times \left((\bar{w}-w^{\mu})(1-\eta^{\mu}x^{\mu}x^{\mu^T}) - \eta^{\mu}\epsilon^{\mu}x^{\mu^T} \right)^T \rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} + \sigma_e^2 \\
&= \frac{1}{N}\langle (\bar{w}-w^{\mu})(1-\eta^{\mu}x^{\mu}x^{\mu^T})^2(\bar{w}-w^{\mu})^T \rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} \\
&\quad -\frac{2}{N}\langle \eta\epsilon^{\mu}x^{\mu^T}(1-\eta^{\mu}x^{\mu}x^{\mu^T})(\bar{w}-w^{\mu})^T \rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} \\
&\quad +\frac{1}{N}\langle (\eta^{\mu})^2(\epsilon^{\mu})^2 x^{\mu^T}x^{\mu} \rangle_{x_{\mu},\epsilon_{\mu}} + \sigma_e^2. \qquad (21)
\end{aligned}$$

The term linear in the noise again vanishes, and we note that $\langle x^{\mu^T}x^{\mu} \rangle_{x_\mu} = \langle \|x^{\mu}\|^2 \rangle_{x_\mu} = 1$. Therefore, the last term's expectation is $\frac{(\eta^{\mu})^2\sigma_e^2}{N}$, and

$$\begin{aligned}
E_g[\mu+1] &= \frac{1}{N}\langle (\bar{w}-w^{\mu})(1-\eta^{\mu}x^{\mu}x^{\mu^T})^2(\bar{w}-w^{\mu})^T \rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} \\
&\quad +\left(1+\frac{(\eta^{\mu})^2}{N}\right)\sigma_e^2 \\
&= \frac{1}{N}\langle \|\bar{w}-w^{\mu}\|^2 \rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} - \frac{2}{N}\langle (\bar{w}-w^{\mu})\eta^{\mu}x^{\mu}x^{\mu^T}(\bar{w}-w^{\mu})^T \rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} \\
&\quad +\frac{1}{N}\langle (\bar{w}-w^{\mu})(\eta^{\mu}x^{\mu}x^{\mu^T})^2(\bar{w}-w^{\mu})^T \rangle_{\bar{w},x_{1:\mu},\epsilon_{1:\mu}} \\
&\quad +\left(1+\frac{(\eta^{\mu})^2}{N}\right)\sigma_e^2 \qquad (22)
\end{aligned}$$

To simplify this expression, it's convenient to note that

$$\begin{aligned}
\langle (x^\mu x^{\mu^T})^2_{ij} \rangle_{x^\mu} &= \sum_k \langle x^\mu_i x^\mu_k x^\mu_k x^\mu_j \rangle_{x^\mu} \\
&= \sum_k (2\langle x^\mu_i x^\mu_k \rangle_{x^\mu} \langle x^\mu_k x^\mu_j \rangle_{x^\mu} + \langle x^\mu_i x^\mu_j \rangle_{x^\mu} \langle (x^\mu_k)^2 \rangle_{x^\mu}) \\
&= \sum_k \frac{1}{N^2}(2\delta_{ik}\delta_{jk} + \delta_{ij}) \\
&= \frac{2}{N^2}\delta_{ij} + \frac{1}{N}\delta_{ij},
\end{aligned} \tag{23}$$

where the second line follows from Wick's theorem for Gaussian moments, and δ_{ij} is the Kronecker delta. This implies $\langle (x^\mu x^{\mu^T})^2 \rangle = \frac{2}{N^2}I + \frac{1}{N}I$, and together with $\langle x^\mu x^{\mu^T} \rangle = \frac{1}{N}I$, our expression for $E_g[\mu+1]$ becomes,

$$\begin{aligned}
E_g[\mu+1] &= \left[1 - \frac{2\eta^\mu}{N} + (\eta^\mu)^2 \frac{2+N}{N^2}\right] \frac{1}{N} \langle \|\bar{w} - w^\mu\|^2 \rangle_{\bar{w}, x_{1:\mu}, \epsilon_{1:\mu}} \\
&\quad + \left(1 + \frac{(\eta^\mu)^2}{N}\right)\sigma_e^2.
\end{aligned} \tag{24}$$

This weight norm is related to the generalization error by Eqn. 19, which enables a recursive equation for the generalization error

$$\begin{aligned}
E_g[\mu+1] &= \left[1 - \frac{2\eta^\mu}{N} + (\eta^\mu)^2 \frac{2+N}{N^2}\right] \left(E_g[\mu] - \sigma_e^2\right) \\
&\quad + \left(1 + \frac{(\eta^\mu)^2}{N}\right)\sigma_e^2 \\
&= \left[1 - 2\frac{\eta^\mu}{N} + \left(\frac{\eta^\mu}{N}\right)^2 (2+N)\right] E_g[\mu] \\
&\quad + \left[1 + \frac{(\eta^\mu)^2}{N} - 1 + \frac{2\eta^\mu}{N} - (\eta^\mu)^2 \frac{2+N}{N^2}\right] \sigma_e^2 \\
&= \left[1 - 2\frac{\eta^\mu}{N} + \left(\frac{\eta^\mu}{N}\right)^2 (2+N)\right] E_g[\mu] \\
&\quad + \left[\frac{2\eta^\mu}{N} - \frac{2(\eta^\mu)^2}{N^2}\right] \sigma_e^2 \\
&= \left[1 - 2\frac{\eta^\mu}{N} + \left(\frac{\eta^\mu}{N}\right)^2 (2+N)\right] E_g[\mu] \\
&\quad + 2\left[\frac{\eta^\mu}{N}\left(1 - \frac{\eta^\mu}{N}\right)\right] \sigma_e^2.
\end{aligned} \tag{25}$$

Now passing to the limit $N >> 1$, we have

$$E_g[\mu+1] = \left[1 - \frac{\eta^\mu(2-\eta^\mu)}{N}\right] E_g[\mu] + 2\frac{\eta^\mu}{N}\sigma_e^2. \tag{26}$$

We then enter the high-dimensional regime where $\alpha = \mu/N$ and consider the new continuous variables $E_g(\alpha) \approx E_g[\alpha N]$ for the generalization error and $\eta(\alpha) \approx \eta^{\alpha N}$ for the learning rate[1]. We wish to calculate an equivalent differential equation,

$$\frac{dE_g(\alpha)}{d\alpha} = \lim_{d\alpha \to 0} \frac{E_g(\alpha + d\alpha) - E_g(\alpha)}{d\alpha}, \tag{27}$$

where we take $d\alpha = 1/N$, which is infinitesimal in the limit $N \to \infty$, to approximate the increment provided by a single new example. Thus

$$\begin{aligned} \frac{E_g(\alpha + d\alpha) - E_g(\alpha)}{d\alpha} &= N(E_g[\alpha N + 1] - E_g[\alpha N]) \\ &= -\eta^{\alpha N}(2 - \eta^{\alpha N})E_g[\alpha N] + 2\eta^{\alpha N}\sigma_e^2. \end{aligned} \tag{28}$$

We thus have the ordinary linear differential equation

$$\frac{d}{d\alpha}E_g(\alpha) = -\eta(\alpha)(2 - \eta(\alpha))E_g(\alpha) + 2\eta(\alpha)\sigma_e^2. \tag{29}$$

The solution can be found through the method of integrating factors. In particular, we define

$$H(\alpha) = \int_0^\alpha \eta(\alpha')(2 - \eta(\alpha'))d\alpha', \tag{30}$$

and find

$$E_g(\alpha) = E_g(0)e^{-H(\alpha)} + 2\sigma_e^2 e^{-H(\alpha)} \int_0^\alpha \eta(\tau)e^{H(\tau)}d\tau. \tag{31}$$

4.2 Optimal online learning rate

Equation (31) yields the expected generalization error for arbitrary learning rate functions. To ensure a fair normative comparison to other methods, we now compute the optimal learning rate as a function of example. We again begin by considering a discrete sequence of examples, and we will take the high-dimensional limit at the end.

Let $\eta^{*,\mu}$ denote the learning rate schedule that minimizes the expected generalization error on example $T = \alpha N$. Also let

$$f(x, \eta) = \left[1 - \frac{\eta(2 - \eta)}{N}\right]x + 2\frac{\eta}{N}\sigma_e^2 \tag{32}$$

be the discrete dynamics update from Eqn. (26), that is, the generalization error on example $\mu + 1$ if the generalization error on example μ is x and the learning rate used on example μ is η.

[1] To avoid confusion, recall that $\eta^{\alpha N}$ notates the learning rate for example $\alpha N \approx \mu$, and this notation is not meant to imply an exponential learning rate schedule.

At the penultimate example before the deadline, $T-1$, because there is only one update left, the best learning rate is given by greedily optimizing f,

$$\eta^{*,T-1} = \mathrm{argmin}_{\eta} f(E_g[T-1], \eta). \tag{33}$$

We directly perform the minimization by differentiating with respect to η and setting this derivative to zero,

$$\begin{aligned}
\frac{\partial}{\partial \eta} f(x, \eta) &= \eta x/N - (2-\eta)x/N + 2\sigma_e^2/N, \\
0 &= 2\eta^* x - 2x + 2\sigma_e^2, \\
\eta^*(x) &= 1 - \frac{\sigma_e^2}{x},
\end{aligned} \tag{34}$$

which yields the optimal update of

$$\eta^{*,T-1} = 1 - \frac{\sigma_e^2}{E_g[T-1]}. \tag{35}$$

The final generalization error as a function of the penultimate generalization error, x, is thus

$$\begin{aligned}
g(x) &\equiv \min_{\eta} f(x, \eta) \\
&= f(x, \eta^*(x)) \\
&= \left[1 - \frac{\left(1 - \frac{\sigma_e^2}{x}\right)(1 + \frac{\sigma_e^2}{x})}{N} \right] x + 2\frac{1 - \frac{\sigma_e^2}{x}}{N}\sigma_e^2 \\
&= \left[1 - \frac{\left(1 - \frac{\sigma_e^4}{x^2}\right)}{N} \right] x + 2\frac{1 - \frac{\sigma_e^2}{x}}{N}\sigma_e^2 \\
&= (1 - 1/N)x - \frac{\sigma_e^4}{Nx} + 2\frac{\sigma_e^2}{N}.
\end{aligned} \tag{36}$$

Differentiating with respect to x, we have

$$\frac{d}{dx} g(x) = 1 - 1/N + \frac{\sigma_e^4}{Nx^2}, \tag{37}$$

which is strictly positive for $N \geq 1, x > 0$. This indicates that the function $g(x)$ is strictly increasing, meaning that larger generalization errors at the penultimate step directly translate into larger generalization errors at the deadline.

Let $v_\mu(x)$ denote the optimal final generalization error on example T, starting from an error of x at step μ and choosing the optimal learning rate thereafter. We have shown that $v_{T-1}(x) = g(x)$, and it is strictly increasing. Now for the inductive step, assume that $v_{\mu+1}(x)$ is strictly increasing. Then

$$\begin{aligned}
\eta^{*,\mu} &= \mathrm{argmin}_{\eta} v_{\mu+1}(f(x, \eta)) \\
&= \mathrm{argmin}_{\eta} f(x, \eta).
\end{aligned} \tag{38}$$

Therefore the optimal learning rate is again selected by greedily minimizing $f(x, n)$. Finally, we note that $v_\mu(x) = v_{\mu+1}(g(x))$ is the composition of strictly increasing functions, and therefore strictly increasing. This establishes the inductive hypothesis and yields the optimal learning rate function for all examples

$$\eta^{*,\mu} = 1 - \frac{\sigma_e^2}{E_g[\mu]}. \tag{39}$$

In the high-dimensional regime, the optimal learning rate is thus

$$\eta^*(\alpha) = 1 - \frac{\sigma_e^2}{E_g(\alpha)}. \tag{40}$$

Inserting this optimal learning rate function back into Eqn. (29) yields the following optimal generalization error dynamics,

$$\frac{d}{d\alpha} E_g(\alpha) = 2\sigma_e^2 - E_g(\alpha) - \frac{\sigma_e^4}{E_g(\alpha)}. \tag{41}$$

5 Online Notebook Policy

In the online notebook policy, each example is stored in the notebook according to the Hebbian scheme in Eqns. (12)-(16). The notebook is then used to make predictions even for novel inputs, by allowing the notebook to converge to an attractor and reading off the predicted output.

In particular, an input x arriving at the student from the teacher can be used to seed recurrent pattern completion in the notebook, by letting $h(0) = f(U^{S_x \to N} x)$ and then running the notebook dynamics. In the simulations in the main text, rather than run the recurrent dynamics to convergence, we use the pattern obtained after 9 updates. At each update, the neurons are ranked by net input and the threshold θ is chosen so that the top aM are active (in the case of ties, slightly more neurons can be active). After the network dynamics have settled on some pattern $\tilde{\xi}$, a predicted output can be generated (using just the notebook) as $\tilde{y} = V^{N \to S_y} \tilde{\xi}$.

This section shows that, in the high-dimensional setting considered here, the notebook attains low memorization error (i.e., error on already-experienced examples) but is incapable of generalization.

5.1 Hebbian learning rule scale factor and offset

The memorization ability of recurrent attractor networks, as well as the performance of Hebbian plasticity rules in mapping from notebook activity patterns to student activity patterns, is known to depend on the statistics of the patterns and the specific form of the learning rule used to configure the weights [19, 5, 6, 37]. We begin by justifying the scaling and subtractive offsets in Eqns. (12)-(16), typically as an approximate implementation of the pseudoinverse learning rule given our sparse pattern statistics.

5.1.1 Recurrent weights

The job of the notebook is to faithfully memorize example patterns as attractors of neural network dynamics. The pseudoinverse learning rule is a flexible mechanism to memorize these patterns, wherein the $M \times M$ matrix of recurrent notebook connections would be

$$J = \xi\xi^{+} = \xi(\xi^{T}\xi)^{-1}\xi^{T}, \tag{42}$$

where ξ^{+} is the pseudoinverse of ξ, and we assumed that $P \leq M$. Suppose that the neural network dynamics have the form $h(u) = f(Jh(u-1))$, where h is the pattern of notebook activity. Assuming that $f(0) = 0$ and $f(1) = 1$ (e.g., f may be linear, threshold-linear, or binary), then these weights would successfully memorize all P patterns as steady-states of the network dynamics. In particular, note that

$$f(J\xi) = f(\xi(\xi^{T}\xi)^{-1}\xi^{T}\xi) = f(\xi) = \xi, \tag{43}$$

so that the network dynamics map each memorized pattern back onto itself[2]. It is instructive to expand the pseudoinverse weights in terms of the stored patterns,

$$J_{ij} = \sum_{\mu=1}^{P}\sum_{\nu=1}^{P} \xi_{i\mu}(\xi^{T}\xi)^{-1}_{\mu\nu}\xi_{j\nu}. \tag{44}$$

This reveals a practical problem with the pseudoinverse learning rule, as the storage prescription for each pattern depends on the other stored patterns through the inverse pattern correlation, $(\xi^{T}\xi)^{-1}_{\mu\nu}$.

The Hopfield model can be viewed as a solution to this problem that assumes simple random statistics for ξ in order to simplify the necessary structure of the learning rule. In particular, suppose that each memory randomly assigns aM neurons to the 1-state and $(1-a)M$ neurons to the 0-state. Thus, a quantifies the fraction of 1-states in the memorized patterns, and we refer to a as the sparseness parameter. We also assume that the memorized patterns are statistically independent from each other. These statistics imply that

$$\begin{aligned}\langle(\xi^{T}\xi)_{\mu\nu}\rangle_{\xi} = \sum_{i=1}^{M}\langle\xi_{i\mu}\xi_{i\nu}\rangle_{\xi} &= \sum_{i=1}^{M}(a\delta_{\mu\nu} + a^{2}(1-\delta_{\mu\nu})) \\ &= Ma^{2} + Ma(1-a)\delta_{\mu\nu},\end{aligned} \tag{45}$$

where $\langle\cdot\rangle$ now denotes the average over notebook patterns. In matrix notation, this implies that

$$\langle\xi^{T}\xi\rangle_{\xi} = Ma(1-a)I_{P} + Ma^{2}1_{P}1_{P}^{T}, \tag{46}$$

[2] Note that this argument neglected the threshold present in the notebook dynamics. This choice reflects the fact that we want the argument to carry over to the linear student neurons that we will consider in the next section. Nevertheless, pseudoinverse weights also work for thresholded notebook neurons in the typical case that $f(-\theta) = 0$ and $f(1-\theta) = 1$.

where I_P is the $P \times P$ identity matrix, and 1_P is the P-vector of ones. This form allows us to use the Sherwood-Morrison formula,

$$\left(A + uv^T\right)^{-1} = A^{-1} - \frac{A^{-1}uv^TA^{-1}}{1 + v^TA^{-1}u}, \tag{47}$$

with $A = Ma(1-a)I_P$, $u = Ma^2 1_P$, and $v = 1_P$ to obtain

$$\begin{aligned}\langle \xi^T\xi\rangle_\xi^{-1} &= \frac{1}{Ma(1-a)}I_P - \frac{Ma^2/(Ma(1-a))^2 1_P 1_P^T}{1 + MPa^2/(Ma(1-a))} \\ &= \frac{1}{Ma(1-a)}I_P - \frac{1/(M(1-a)^2)}{1 + Pa/(1-a)}1_P 1_P^T \\ &= \frac{1}{Ma(1-a)}I_P - \frac{1}{M(1-a)^2 + MPa(1-a)}1_P 1_P^T \\ &\approx \frac{1}{Ma(1-a)}I_P - \frac{1}{M^2\beta a(1-a)}1_P 1_P^T, \end{aligned} \tag{48}$$

where the final approximation used $P = \beta M$, $\beta = O(1)$, and $M \gg 1$. The Hopfield model approximates the pseudoinverse learning rule by replacing $(\xi^T\xi)^{-1}$ by $\langle \xi^T\xi\rangle_\xi^{-1}$. To see what this means, we need to do a bit more algebra:

$$J \approx \xi\langle \xi^T\xi\rangle_\xi^{-1}\xi^T = \frac{\xi\xi^T}{Ma(1-a)} - \frac{(\xi 1_P)(\xi 1_P)^T}{M^2\beta a(1-a)}. \tag{49}$$

The Hopfield model also approximates $\xi 1_P$ by $\langle \xi 1_P\rangle = Pa1_M$, where 1_M is the M-vector of ones, such that

$$J \approx \frac{\xi\xi^T}{Ma(1-a)} - \frac{a^2P^2 1_M 1_M^T}{M^2\beta a(1-a)} = \frac{1}{Ma(1-a)}\xi\xi^T - \frac{\beta a}{1-a}1_M 1_M^T. \tag{50}$$

To compare this to the Hopfield model, we first consider a general Hebbian weight matrix of the form,

$$J_{ij} = \sum_{\mu=1}^{P} B(\xi_{i\mu} - b)(\xi_{j\mu} - b), \tag{51}$$

where B and b are constants that scale and center the learning rule. Again using the approximation that $\xi 1_P \approx \langle \xi 1_P\rangle_\xi = Pa1_M$, we find

$$\begin{aligned} J &= B\xi\xi^T - Bb1_M 1_P^T\xi^T - Bb\xi 1_P 1_M^T + Bb^2 1_M 1_P^T 1_P 1_M^T \\ &\approx B\xi\xi^T + (-2abBP + b^2BP)1_M 1_M^T = B\xi\xi^T + bBP(-2a + b)1_M 1_M^T. \end{aligned} \tag{52}$$

Comparing Eqs. (50) and (52), we see that the two correspond when

$$B = \frac{1}{Ma(1-a)} \tag{53}$$

and

$$\begin{aligned} -\frac{\beta a}{1-a} &= bBP(-2a+b) = \frac{bP(b-2a)}{Ma(1-a)} = \frac{b\beta(b-2a)}{a(1-a)} \\ &\implies 0 = b^2 - 2ba + a^2 = (b-a)^2 \implies b = a. \end{aligned} \tag{54}$$

Therefore, the pseudoinverse rule can be approximated by the Hebbian rule,

$$J_{ij} = \sum_{\mu=1}^{P} \frac{(\xi_{i\mu} - a)(\xi_{j\mu} - a)}{Ma(1-a)}, \tag{55}$$

which is the weight matrix of the Hopfield model and the first term in Eqn. (12) of the notebook learning rules.

5.1.2 Notebook-to-student weights

Similar to the Hopfield storage prescription used to store binary indices as fixed points of the recurrent notebook dynamics, here we assume Hebbian connectivity between the notebook and student. In particular, we can form the $(N+1)\times P$ matrix Z by vertically stacking the matrices X and Y, such that Z represents the combined student input-output activity to be stored. We also define the $(N+1)\times M$ matrix V by vertically stacking the matrices $V^{N\to S_x}$ and $V^{N\to S_y}$, which represents the mapping from notebook activity to student activity. In this setting the relevant pseudoinverse learning rule for the weights from notebook to student neurons is

$$V = Z\xi^{+} = Z(\xi^T\xi)^{-1}\xi^T. \tag{56}$$

The same approximations used in the previous section lead to

$$V \approx \frac{Z\xi^T}{Ma(1-a)} - \frac{(Z1_P)(\xi 1_P)^T}{M^2\beta a(1-a)}. \tag{57}$$

Replacing $\xi 1_P$ by $\langle \xi 1_P \rangle = Pa1_M$, we find

$$V \approx \frac{Z\xi^T}{Ma(1-a)} - \frac{Za1_P 1_M^T}{Ma(1-a)} = \frac{Z(\xi^T - a1_P 1_M^T)}{Ma(1-a)}, \tag{58}$$

or

$$V_{ij} \approx \sum_{\mu=1}^{P} \frac{Z_{i\mu}(\xi_{j\mu} - a)}{Ma(1-a)}. \tag{59}$$

This is the Hebbian learning rule that we use to connect the notebook to the student for purposes of pattern reactivation (Eqns. (15)-(16)).

5.1.3 Student-to-notebook weights

We did not derive the student-to-notebook weights from a pseudoinverse rule. In particular, the associated pseudoinverse rule would depend on Z^+, which must be computed differently depending on whether $P \leq N+1$ or $P > N+1$. In contrast, we assumed that $P \leq M$ throughout, which allowed a unified expression for ξ^+. Moreover, memory recall means that the notebook will sometimes be activated by a subset of student neurons, so defining weights based on Z^+ may be inappropriate in some circumstances.

Nevertheless, the form of the student-to-notebook weights is justifiable. Define the $M \times (N+1)$ matrix U by horizontally stacking the matrices $U^{S_x \to N}$ and $U^{S_y \to N}$, which represents the mapping from student activity to notebook activity. It's useful to note that Eqs. (13)-(14) imply

$$\begin{aligned} \langle (UZ)_{i\mu} \rangle_Z &= \sum_\nu \sum_{j=1}^{N+1} \langle (\xi_{i\nu} - a) Z_{j\nu} Z_{j\mu} \rangle_Z = \sum_\nu (\xi_{i\nu} - a) \left(\sum_{j=1}^{N} \frac{1}{N} \delta_{\mu\nu} + \delta_{\mu\nu} \right) \\ &= 2(\xi_{i\mu} - a) \qquad (60) \end{aligned}$$

where the expectation is now over student patterns, and we noted that $\langle X_{j\mu} X_{j\nu} \rangle_X = \frac{1}{N} \delta_{\mu\nu}$ and $\langle Y_\mu Y_\nu \rangle_Y = \delta_{\mu\nu}$. Therefore, $\langle (UZ)_{i\mu} \rangle_Z < \xi_{i\mu}$ if $\xi_{i\mu} = 0$, and $\langle (UZ)_{i\mu} \rangle_Z > \xi_{i\mu}$ if $\xi_{i\mu} = 1$ and $a < 0.5$. Consequently, these weights are expected to seed the appropriate pattern in the binary notebook network with sparse memories. Similarly,

$$\langle (U^{S_x \to N} X)_{i\mu} \rangle_X = \xi_{i\mu} - a, \qquad (61)$$

so $\langle (UX)_{i\mu} \rangle_X < \xi_{i\mu}$ if $\xi_{i\mu} = 0$, and $\langle (UX)_{i\mu} \rangle_X > a$ if $\xi_{i\mu} = 1$ and $a < 0.5$. The input neurons are thus also expected to seed the appropriate pattern if $0 < \theta < a$, or if the threshold is dynamically chosen to maintain the desired spareness level.

5.2 Notebook memory error

With these Hebbian learning prescriptions in hand, we now characterize their performance. In this section, we consider the typical memory error by examining the statistics by which the notebook reactivates stored patterns of student activity. Previous studies of the Hopfield model [5, 6, 37] imply that large notebooks can accurately recall each random index if the number of stored patterns does not exceed the capacity of the network $P_c = \beta_c M$. Here we assume that $M \gg 1$ and $P < P_c$, such that erroneous index retrieval by the notebook is rare. Once a notebook memory index is accurately retrieved by the notebook's dynamics, the notebook can generate a predicted output using the Hebbian weights from notebook to student output ($V^{N \to S_y}$). The memory error of the notebook can thus be approximated as the typical error of this prediction. Real hippocampal networks likely exhibit active forgetting to enhance generalization or memory capacity [7, 32], and it would be interesting to consider alternate notebook models that incorporate forgetting effects in future work [28].

As in the previous section, let Z be a $(N+1)\times P$ matrix that groups together all input and output neuron responses for all memorized patterns. Then the notebook reactivated student pattern is

$$\hat{Z}_{i\mu} = \sum_{j=1}^{M} V_{ij}\xi_{j\mu} = \sum_{j=1}^{M}\sum_{\nu=1}^{P} \frac{1}{Ma(1-a)} Z_{i\nu}(\xi_{j\nu}-a)\xi_{j\mu}. \tag{62}$$

We first consider how well the notebook reactivates the student on average. In particular, averaging this expression over all possible notebook indices gives

$$\begin{aligned}\langle \hat{Z}_{i\mu}\rangle_{Z,\xi} \quad &= \tfrac{1}{Ma(1-a)} \textstyle\sum_{j=1}^{M}\sum_{\nu=1}^{P} Z_{i\nu}\langle(\xi_{j\nu}-a)\xi_{j\mu}\rangle_\xi \\ &= \tfrac{1}{Ma(1-a)} \textstyle\sum_{j=1}^{M}\sum_{\nu=1}^{P} Z_{i\nu}a(1-a)\delta_{\mu\nu} = Z_{i\mu}.\end{aligned} \tag{63}$$

Therefore, the Hebbian learning rule is unbiased, and it on average reactivates all student neuron responses accurately.

However, the randomness of notebook indices does cause notebook-driven student reactivations to fluctuate away from these average values. To determine the magnitude of notebook memory error quantitatively, first note that the memory error of the notebook is

$$\mathcal{E}_m = \frac{1}{P}\sum_{\mu=1}^{P}(Y_\mu - \hat{Y}_\mu)^2 = \frac{1}{P}\sum_{\mu=1}^{P}(Y_\mu^2 - 2Y_\mu\hat{Y}_\mu + \hat{Y}_\mu^2). \tag{64}$$

Averaging over possible notebook patterns, we find

$$E_m = \langle \mathcal{E}_m\rangle_\xi = \frac{1}{P}\sum_{\mu=1}^{P}(Y_\mu^2 - 2Y_\mu^2 + \langle \hat{Y}_\mu^2\rangle_\xi) = \frac{1}{P}\sum_{\mu=1}^{P}\mathrm{Var}(\hat{Y}_\mu). \tag{65}$$

This variance term can be written

$$\mathrm{Var}(\hat{Y}_\mu) = \left\langle \sum_{\nu=1}^{P}\sum_{j=1}^{M}\frac{Y_\nu(\xi_{j\nu}-a)}{Ma(1-a)}\xi_{j\mu}\sum_{\rho=1}^{P}\sum_{k=1}^{M}\frac{Y_\rho(\xi_{k\rho}-a)}{Ma(1-a)}\xi_{k\mu}\right\rangle_\xi - Y_\mu^2. \tag{66}$$

This expression shows that the exact value of the notebook training error depends on the specific realizations of the student outputs.

However, for practical purposes, it will be good enough to average Eq. (66) over possible student outputs, and noting that $\langle Y_\mu Y_\nu\rangle_Y = \delta_{\mu\nu}$, we find

$$\begin{aligned}\langle \mathrm{Var}(\hat{Y}_\mu)\rangle_Y =& \frac{1}{\left(Ma(1-a)\right)^2}\sum_{\nu=1}^{P}\sum_{j=1}^{M}\sum_{k=1}^{M}\langle(\xi_{j\nu}-a)\xi_{j\mu}(\xi_{k\nu}-a)\xi_{k\mu}\rangle_Y - 1 \\ =& \frac{1}{\left(Ma(1-a)\right)^2}\sum_{j=1}^{M}\sum_{k=1}^{M}\Big(\langle(\xi_{j\mu}-a)\xi_{j\mu}(\xi_{k\mu}-a)\xi_{k\mu}\rangle_Y \\ &+ \sum_{\nu\neq\mu}\langle(\xi_{j\nu}-a)\xi_{j\mu}(\xi_{k\nu}-a)\xi_{k\mu}\rangle_Y\Big) - 1.\end{aligned} \tag{67}$$

It is straightforward to evaluate the first expectation as

$$\begin{aligned}
&\langle(\xi_{j\mu}-a)\xi_{j\mu}(\xi_{k\mu}-a)\xi_{k\mu}\rangle_Y\\
=&\delta_{jk}(1-a)^2P(\xi_{j\mu}=1)+(1-\delta_{jk})(1-a)^2P(\xi_{j\mu}=1)P(\xi_{k\mu}=1|\xi_{j\mu}=1)\\
=&\delta_{jk}(1-a)^2a+(1-\delta_{jk})(1-a)^2a\frac{aM-1}{M-1}\\
=&\delta_{jk}a(1-a)^2+(1-\delta_{jk})\left(Ma^2(1-a)^2/(M-1)-a(1-a)^2/(M-1)\right). \quad (68)
\end{aligned}$$

Because $\mu \neq \nu$ in the second expectation of Eq. (67), it straightforwardly separates into the product of two terms:

$$\langle(\xi_{j\nu}-a)\xi_{j\mu}(\xi_{k\nu}-a)\xi_{k\mu}\rangle_Y=\langle(\xi_{j\nu}-a)(\xi_{k\nu}-a)\rangle_Y\,\langle\xi_{j\mu}\xi_{k\mu}\rangle_Y\,. \quad (69)$$

First,

$$\begin{aligned}
\langle(\xi_{j\nu}-a)\ (\xi_{k\nu}-a)\rangle_Y =&\ \delta_{jk}a(1-a)\\
&+(1-\delta_{jk})\Big((1-a)^2P(\xi_{j\nu}=1)P(\xi_{k\nu}=1|\xi_{j\nu}=1)\\
&-a(1-a)P(\xi_{j\nu}=1)P(\xi_{k\nu}=0|\xi_{j\nu}=1)\\
&-a(1-a)P(\xi_{j\nu}=0)P(\xi_{k\nu}=1|\xi_{j\nu}=0)\\
&+a^2P(\xi_{j\nu}=0)P(\xi_{k\nu}=0|\xi_{j\nu}=0)\Big)\\
=&\delta_{jk}a(1-a)+(1-\delta_{jk})\Big((1-a)^2a\frac{aM-1}{M-1}-a(1-a)a\frac{(1-a)M}{M-1}\\
&-a(1-a)(1-a)\frac{aM}{M-1}+a^2(1-a)\frac{(1-a)M-1}{M-1}\Big)\\
=&\delta_{jk}a(1-a)+(1-\delta_{jk})\Big(-a(1-a)^2\frac{1}{M-1}-a^2(1-a)\frac{1}{M-1}\Big)\\
=&\delta_{jk}a(1-a)-(1-\delta_{jk})\frac{a(1-a)}{M-1}. \quad (70)
\end{aligned}$$

Second,

$$\begin{aligned}
\langle\xi_{j\mu}\xi_{k\mu}\rangle_Y =&\delta_{jk}a+(1-\delta_{jk})P(\xi_{j\mu}=1)P(\xi_{k\mu}=1|\xi_{j\mu}=1)\\
=&\delta_{jk}a+(1-\delta_{jk})a\frac{aM-1}{M-1}. \quad (71)
\end{aligned}$$

Combining these two terms, we find,

$$\langle(\xi_{j\nu}-a)\xi_{j\mu}(\xi_{k\nu}-a)\xi_{k\mu}\rangle_Y=\delta_{jk}a^2(1-a)-(1-\delta_{jk})a^2(1-a)\frac{aM-1}{(M-1)^2} \quad (72)$$

for $\mu \neq \nu$. Plugging these expressions back into the expression for $\langle\mathrm{Var}(\hat{Y}_\mu)\rangle_Y$, we find

$$
\begin{aligned}
\langle \mathrm{Var}(\hat{Y}_\mu)\rangle_Y = & \frac{1}{\left(Ma(1-a)\right)^2}\sum_{j=1}^{M}\sum_{k=1}^{M}\Bigg(\delta_{jk}a(1-a)^2 \\
& + (1-\delta_{jk})\big(Ma^2(1-a)^2/(M-1) - a(1-a)^2/(M-1)\big) \\
& + \sum_{\nu\neq\mu}\left(\delta_{jk}a^2(1-a) - (1-\delta_{jk})a^2(1-a)\frac{aM-1}{(M-1)^2}\right)\Bigg) - 1 \\
= & \frac{1}{\left(Ma(1-a)\right)^2}\Big(Ma(1-a)^2 \\
& + M(M-1)\big(Ma^2(1-a)^2/(M-1) - a(1-a)^2/(M-1)\big) \\
& + (P-1)Ma^2(1-a) - (P-1)a^2(1-a)M(aM-1)/(M-1)\Big) - 1 \\
= & \frac{Ma(1-a)^2 + M^2a^2(1-a)^2 - Ma(1-a)^2 - M^2a^2(1-a)^2}{M^2a^2(1-a)^2} \\
& + (P-1)\frac{Ma^2(1-a) - a^3(1-a)M^2/(M-1) + a^2(1-a)M/(M-1)}{M^2a^2(1-a)^2} \\
= & (P-1)\frac{M-1-aM+1}{M(1-a)(M-1)} \\
= & \frac{P-1}{M-1}. \qquad (73)
\end{aligned}
$$

The proportionality of $\langle \mathrm{Var}(\hat{Y}_\mu)\rangle_Y$ to $P-1$ intuitively captures the interference of the Hebbian readout of memory μ from the other $P-1$ memories that contribute to $V^{N\to S_y}$.

Combining Eqs. (65) and (73), we find that the expected memory error of the notebook is simply

$$\langle \mathcal{E}_m\rangle_{\xi,Y} = \frac{P-1}{M-1}, \qquad (74)$$

Remarkably, note that this expression is independent of the notebook's sparseness. If $P = \beta M$ and $M \gg 1$, this implies that

$$\langle \mathcal{E}_m\rangle_{\xi,Y} \approx \beta. \qquad (75)$$

We thus see that the expected memorization error of the notebook scales with the number of memories stored in the system and can become significant when the loading is large. Note that this expression only makes sense if $\beta < \beta_c$, because we've assumed faithful index reactivation within the notebook itself.

5.3 Notebook generalization error

Next we examine the expected error when the notebook is used to predict the teacher output on a novel example. Because we operate the Hopfield network

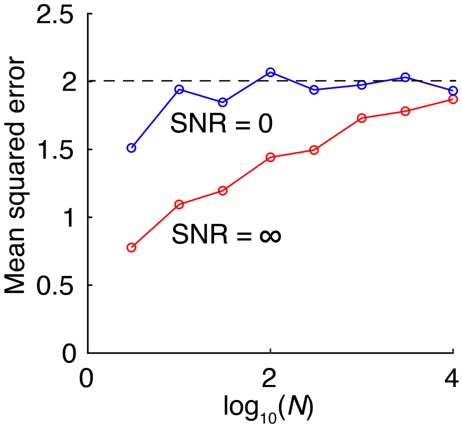


Figure S1: Numerical simulation of the generalization error of the nearest neighbor algorithm, for $\alpha = 1$ and SNR $\mathcal{S} = \infty$ (red) and $\mathcal{S} = 0$ (blue). As input dimension N approaches infinity, generalization error in both cases approaches 2, consistent with our analytical derivation.

below capacity, it successfully embeds all patterns as fixed points with relatively large basins of attraction, and the previous section showed that the student reactivation error is modest. For simplicity, we therefore model the notebook as a nearest neighbor algorithm that operates by returning the output associated with the nearest stored pattern for any given input.

In particular, let $x^\mu, \mu = 1, \cdots, P$ be the N-dimensional column vectors of stored inputs, and $y^\mu, \mu = 1, \cdots, P$ be the associated outputs. For a novel input $x \in R^N$, we find the nearest neighbor as

$$\mu^* \quad = \quad \mathrm{argmin}_\mu \left\| x - x^\mu \right\|^2 . \tag{76}$$

With the nearest neighbor identified, the prediction is $\hat{y} = y^{\mu^*} = \bar{w}x^{\mu^*} + \epsilon^{\mu^*}$. The expected generalization error is thus

$$\begin{aligned} E_g &= \langle (y-\hat{y})^2 \rangle_{x, x_{1:P}, \bar{w}, \epsilon, \epsilon_{1:P}} \\ &= \left\langle \left(\bar{w} \left(x - x^{\mu^*} \right) + \epsilon - \epsilon^{\mu^*} \right)^2 \right\rangle_{x, x_{1:P}, \bar{w}, \epsilon, \epsilon_{1:P}} \\ &= \left\langle \left(x - x^{\mu^*} \right)^T \bar{w}^T \bar{w} \left(x - x^{\mu^*} \right) \right\rangle_{x, x_{1:P}, \bar{w}} \\ &\quad +2\langle (\epsilon - \epsilon^{\mu^*}) \bar{w} (x - x^{\mu^*}) \rangle_{x, x_{1:P}, \bar{w}, \epsilon, \epsilon_{1:P}} + \langle (\epsilon - \epsilon^{\mu^*})^2 \rangle_{\epsilon, \epsilon_{1:P}}, \end{aligned} \tag{77}$$

We used that $\bar{w} \left(x - x^{\mu^*} \right) = \left(x - x^{\mu^*} \right)^T \bar{w}^T$ is a scalar. This form allows us to evaluate the expectation over teacher weights as $\langle \bar{w}^T \bar{w} \rangle_{\bar{w}} = \sigma_{\bar{w}}^2 I$. Also noting that the noise is uncorrelated with everything else, we find

$$\begin{aligned} E_g &= \sigma_{\bar{w}}^2 \left\langle \left(x - x^{\mu^*}\right)^T \left(x - x^{\mu^*}\right) \right\rangle_{x, x_{1:P}} + 2\sigma_e^2 \\ &= \sigma_{\bar{w}}^2 \left\langle \left\| x - x^{\mu^*} \right\|_2^2 \right\rangle_{x, x_{1:P}} + 2\sigma_e^2, \end{aligned} \tag{78}$$

where the prefactor of 2 in the noise term reflects the independent fluctuations of ϵ and ϵ^{μ^*}. We note that $x - x^\mu \sim \mathcal{N}(0, \frac{2}{N} I)$ for all μ, and so $z^\mu = \sqrt{\frac{N}{2}}(x - x^\mu) \sim \mathcal{N}(0, I)$. By the Gaussian Annulus Theorem (see e.g., Thm 2.9, pg 15 of [10]),

$$\begin{aligned} P\left(\left| \|z^\mu\| - \sqrt{N} \right| \geq \beta\right) &\leq 3e^{-c\beta^2}, \\ P\left(\left| \sqrt{\frac{N}{2}} \|x - x^\mu\| - \sqrt{N} \right| \geq \beta\right) &\leq 3e^{-c\beta^2}, \\ P\left(\left| \|x - x^\mu\| - \sqrt{2} \right| \geq \beta\sqrt{2/N}\right) &\leq 3e^{-c\beta^2}, \end{aligned} \tag{79}$$

where $c > 0$ is a constant independent of N. By the union bound, the probability that one or more patterns among the $\mu = 1, \cdots, \alpha N$ fails to concentrate is no more than

$$P\left(\bigcup_{\mu=1}^{\alpha N} \left\{ \left| \|x - x^\mu\| - \sqrt{2} \right| \geq \beta\sqrt{2/N} \right\}\right) \leq 3\alpha N e^{-c\beta^2}. \tag{80}$$

Therefore the minimum over all patterns will fail to concentrate with probability no more than

$$P\left(\left| \left\| x - x^{\mu^*} \right\| - \sqrt{2} \right| \geq \beta\sqrt{2/N}\right) \leq 3\alpha N e^{-c\beta^2}. \tag{81}$$

Choosing $\beta = N^{1/4}$ we have,

$$P\left(\left| \left\| x - x^{\mu^*} \right\| - \sqrt{2} \right| \geq \sqrt{2} N^{-1/4}\right) \leq 3\alpha N e^{-c\sqrt{N}}, \tag{82}$$

such that as $N \to \infty$, the minimum concentrates near $\sqrt{2}$ with probability one. Substituting back into the expression for the expected generalization error, in the high dimensional limit with high probability we have

$$E_g = 2\sigma_{\bar{w}}^2 + 2\sigma_e^2. \tag{83}$$

For our standard scaling where $\sigma_{\bar{w}}^2 + \sigma_e^2 = 1$, the error is therefore 2 regardless of the SNR. We note that this result applies in the high-dimensional limit where $N, P \to \infty$ and their ratio is $\alpha = P/N$. In finite size simulations, the generalization error can modestly differ, as shown in Fig. S1.

In essence, in the high-dimensional regime, the nearest neighbor is typically very far away from the new sample, such that generalization fails completely.

In fact, it is so poor that always predicting zero would be better (attaining generalization error of 1 rather than 2 for our setting). This finding strongly motivates the need for a trained student, but we note that notebook-mediated generalization could be better in different settings where, for instance, input examples arise from a low number of clusters [16].

6 Memorization-optimized Replay Policy

In the memorization-optimized replay policy, each example is stored in the notebook according to the Hebbian scheme in Eqns. (12)-(16). These patterns can then be reactivated offline to drive learning. In the simulations reported in the main text, offline notebook reactivations undergo a two-step retrieval process:

1. A random binary pattern is used to seed the reactivation event. Starting at this random state, the notebook updates through the recurrent dynamics 9 times synchronously to retrieve a stored pattern. On each update, the threshold θ is chosen to enforce a sparsity of a (up to ties, which can cause slightly more neurons to be active). Without this adaptive threshold, a silent attractor dominates retrieval.

2. The notebook then uses the retrieved pattern from (1) to seed a second round of pattern completion using a fixed threshold $\theta = -0.15$, which in combination with the global inhibition parameter $\gamma = 0.6$ provides good retrieval alongside the possibility of retrieving a silent state (see [12] for detailed derivation of performance as a function of these parameters). This two step process enables retrieval of patterns that are not forced to have a fixed sparseness, and a "silent state" attractor can be retrieved when the seeding pattern lies far away from any of the encoded patterns.

This models a simple form of replay. Supposing that the notebook pattern at convergence is $\tilde{\xi}$, the student input and target output are then reconstructed based on the Hebbian connectivity as $\tilde{x} = V^{N \to S_x}\tilde{\xi}$ and $\tilde{y} = V^{N \to S_y}\tilde{\xi}$. This provides an $\{\tilde{x}, \tilde{y}\}$ sample from which the student can learn using gradient descent.

The policy is memory-optimized, in the sense that this replay continues indefinitely, such that all samples stored in the notebook are eventually learned by the student. This section characterizes the memory and generalization performance of the student resulting from this replay process. If reactivations perfectly reconstructed the stored examples, this replay strategy would be similar to 'batch' learning strategies in machine learning, in which the same stored dataset is repeatedly revisited to update network weights. However, errors in reactivation could in principle degrade the learning process. In Section 6.1 we show that although reactivations introduce errors, remarkably, these errors are correlated in such a way that learning still proceeds like batch learning from perfectly recalled examples up to a rescaling of the learning rate. Using this fact, in Section 6.2 we provide the expected memory and generalization errors, based on results known in prior work [22, 4].

In this policy, both the notebook and student learn potentially beneficial information, and in principle either could be used to answer a specific query for a point x. We take the normative assumption that the best system is selected to make the prediction. Often, this means that the output for a previously stored input will be predicted by the notebook, while that for a novel input will be predicted by the student. However, in Section 6.1 we show that there are conditions under which the student memory error in fact surpasses the notebook, and the student would be used to make predictions for previously stored inputs.

6.1 Accurate learning despite errors in reactivation

How do reactivation errors influence learning dynamics in the student? One hint that learning from reactivations can be effective comes from Fig. 2 of the main text. Given that the notebook is specifically designed to rapidly store memories, it often has a lower memory error than the student. Surprisingly, however, Figs. 2a-h of the main paper show that the student's training error can fall below that of the notebook. How could it be that the student learns to accurately produce a memory that was imperfectly memorized by the notebook? Our key theoretical observation is that although the notebook imperfectly activates the output of the student, it also imperfectly activates the inputs of the student. These errors are correlated between input and output neurons in a way that does not harm student learning. We demonstrate this fact in this section.

Reactivations have subtly different statistics to the original samples. In particular, when the notebook settles on a pattern ξ^μ (one column of the matrix ξ) that was associated with an original sample x^μ, y^μ from the teacher, this results in reactivated student activity input and output patterns $\tilde{x}^\mu = V^{N\to S_x}\xi^\mu$ and $\tilde{y}^\mu = V^{N\to S_y}\xi^\mu$, respectively. Horizontally concatenating the input and output reactivations into the matrices $\tilde{X} \in R^{N\times P}$ and $\tilde{Y} \in R^{1\times P}$, this reactivation leads the weights in the student network to change (in the reactivated gradient direction) by the amount,

$$\begin{aligned}\tilde{\Delta}_\mu w_i &= -\lambda\frac{\partial}{\partial w_i}\left(\sum_j w_j\tilde{X}_{j\mu} - \tilde{Y}_\mu\right)^2 = -2\lambda\left(\sum_j w_j\tilde{X}_{j\mu} - \tilde{Y}_\mu\right)\tilde{X}_{i\mu} \\ &= -2\lambda\left(\sum_j w_j\tilde{X}_{i\mu}\tilde{X}_{j\mu} - \tilde{X}_{i\mu}\tilde{Y}_\mu\right). \qquad (84)\end{aligned}$$

Therefore, the change expected from gradient descent learning with a random notebook index is

$$\langle\tilde{\Delta}_\mu w_i\rangle_\xi = -2\lambda\left(\sum_j w_j\langle\tilde{X}_{i\mu}\tilde{X}_{j\mu}\rangle_\xi - \langle\tilde{X}_{i\mu}\tilde{Y}_\mu\rangle_\xi\right). \qquad (85)$$

To evaluate these expectations, we form the matrix $\tilde{Z}$ by vertically stacking $\tilde{X}$ and $\tilde{Y}$, then note that

$$\langle \tilde{Z}_{i\mu}\tilde{Z}_{j\mu}\rangle_\xi = \frac{1}{M^2a^2(1-a)^2}\sum_{\nu=1}^{P}\sum_{k=1}^{M}\sum_{\rho=1}^{P}\sum_{l=1}^{M} Z_{i\nu}Z_{j\rho}\langle(\xi_{k\nu}-a)\xi_{k\mu}(\xi_{l\rho}-a)\xi_{l\mu}\rangle. \tag{86}$$

When $\mu \neq \nu \neq \rho$, the statistical independence of memories allows us to factor out $\langle \xi_{k\nu} - a\rangle$, which is zero and causes the whole term to vanish. Similarly, we get no contributions if $\mu \neq \rho \neq \nu$. This implies that both the ν and ρ indices must either pair with each other or with μ, and the only terms that contribute are thus $\nu = \rho = \mu$ and $\nu = \rho \neq \mu$.

$$\begin{aligned}\langle \tilde{Z}_{i\mu}\tilde{Z}_{j\mu}\rangle_\xi = \frac{1}{M^2a^2(1-a)^2}\sum_{k=1}^{M}\sum_{l=1}^{M}\Big(&Z_{i\mu}Z_{j\mu}\langle(\xi_{k\mu}-a)\xi_{k\mu}(\xi_{l\mu}-a)\xi_{l\mu}\rangle \\ &+\sum_{\nu\neq\mu} Z_{i\nu}Z_{j\nu}\langle(\xi_{k\nu}-a)\xi_{k\mu}(\xi_{l\nu}-a)\xi_{l\mu}\rangle\Big).\end{aligned} \tag{87}$$

Both of these expectations have been calculated en route to calculating the notebook's training error. Plugging Eqs. (68) and (72) into the above expression, we find,

$$\begin{aligned}\langle \tilde{Z}_{i\mu}\tilde{Z}_{j\mu}\rangle_\xi =&\frac{1}{M^2a^2(1-a)^2}\sum_{k=1}^{M}\sum_{l=1}^{M}\Bigg(Z_{i\mu}Z_{j\mu}\big(\delta_{kl}a(1-a)^2 \\ &+(1-\delta_{kl})\big(Ma^2(1-a)^2/(M-1) - a(1-a)^2/(M-1)\big)\big) \\ &+\sum_{\nu\neq\mu} Z_{i\nu}Z_{j\nu}\left(\delta_{kl}a^2(1-a) - (1-\delta_{kl})a^2(1-a)\frac{aM-1}{(M-1)^2}\right)\Bigg) \\ =&\frac{1}{M^2a^2(1-a)^2}\Bigg(Z_{i\mu}Z_{j\mu}\big(Ma(1-a)^2 + M^2a^2(1-a)^2 - Ma(1-a)^2\big) \\ &+\sum_{\nu\neq\mu} Z_{i\nu}Z_{j\nu}\left(Ma^2(1-a) - Ma^2(1-a)\frac{aM-1}{M-1}\right)\Bigg) \\ =&Z_{i\mu}Z_{j\mu} + \sum_{\nu\neq\mu} Z_{i\nu}Z_{j\nu}\frac{(M-1)a - a(aM-1)}{(M-1)Ma(1-a)} \\ =&Z_{i\mu}Z_{j\mu} + \sum_{\nu\neq\mu}\frac{Z_{i\nu}Z_{j\nu}}{M-1}.\end{aligned} \tag{88}$$

Therefore,

$$\begin{aligned}\langle \tilde{\Delta}_\mu w_i\rangle_\xi &= -2\lambda\left(\sum_{j=1}^{M} w_j\left(X_{i\mu}X_{j\mu} + \sum_{\nu\neq\mu}\frac{X_{i\nu}X_{j\nu}}{M-1}\right) - X_{i\mu}Y_\mu - \sum_{\nu\neq\mu}\frac{X_{i\nu}Y_\nu}{M-1}\right) \\ &= \Delta_\mu w_i + \frac{1}{M-1}\sum_{\nu\neq\mu}\Delta_\nu w_i\end{aligned} \tag{89}$$

where $\Delta_\mu w_i$ is the weight update that would occur if the student were perfectly reactivated by the notebook pattern μ. Equivalently, $\Delta_\mu w_i$ is the weight update that would occur from online learning to the teacher's example. Importantly, all contributions to $\langle \tilde{\Delta} w_i \rangle$ are in the gradient direction of one of the teacher examples. Rearranging this expression slightly, we find:

$$\langle \tilde{\Delta}_\mu w_i \rangle_\xi = \left(1 - \frac{1}{M-1}\right) \Delta_\mu w_i + \frac{1}{M-1} \sum_{\nu=1}^{P} \Delta_\nu w_i. \tag{90}$$

Therefore, each notebook reactivation of pattern μ is equivalent to a mini-batch update for that particular pattern with effective learning rate $\lambda \left(1 - \frac{1}{M-1}\right)$, plus a batch update for all stored patterns with effective learning rate $\frac{\lambda}{M-1}$. Similarly, the learning expected by sequential notebook reactivation of all P patterns is

$$\langle \tilde{\Delta} w_i \rangle_\xi \equiv \sum_{\mu=1}^{P} \langle \tilde{\Delta}_\mu w_i \rangle_\xi = \left(1 + \frac{P-1}{M-1}\right) \sum_{\mu=1}^{P} \Delta_\mu w_i \tag{91}$$

This is equivalent to batch learning with an effective learning rate of

$$\tilde{\lambda} = \lambda \left(1 + \frac{P-1}{M-1}\right) \tag{92}$$

In sum, the notebook's imperfect reactivation patterns hurt notebook memory performance, but they do not harm the student's ability to learn from past memories if the learning rate is appropriately controlled.

6.2 Student memory and generalization error from replay

As shown in Sections 5.2 and 6.1, notebook reactivations closely recapitulate stored student activity patterns when run below a critical capacity, and reactivation errors are correlated in such a way as to preserve the relevant statistics for student learning. In this regime, when replay events are random and the learning rate is small, the student effectively learns from the whole batch of samples. Batch learning dynamics differ fundamentally from online learning dynamics, because in the batch setting the noise associated with each example is repeatedly revisited. This difference raises the danger of overfitting to the specific batch of stored data, rather than learning the general rule.

We therefore leverage known solutions to the batch learning dynamics of student-teacher models in our high-dimensional setting [22, 4]. The average memory error is (see Section 2 of [4])

$$E_m^{\mathrm{MO}}(t) = \frac{1}{\alpha} \int \rho^{MP}(\lambda) \left(\frac{1 + \lambda \mathcal{S}}{1 + \mathcal{S}} + \lambda \sigma_w^2 \right) e^{-\frac{2\lambda t}{\tau}} d\lambda + \left(1 - \frac{1}{\alpha}\right) \frac{1}{1 + \mathcal{S}} \mathbb{1}\{\alpha > 1\}, \tag{93}$$

and the generalization error is

$$E_g^{\mathrm{MO}}(t) = \int \rho^{\mathrm{MP}}(\lambda) \left[\left(\frac{\mathcal{S}}{1+\mathcal{S}} + \sigma_w^2 \right) e^{-\frac{2\lambda t}{\tau}} + \frac{1}{\lambda(1+\mathcal{S})}(1 - e^{-\frac{\lambda t}{\tau}})^2 \right] d\lambda + \frac{1}{1+\mathcal{S}}, \tag{94}$$

where the superscript MO stands for "memory-optimized," t here measures time in units of epochs, such that each stored example will be replayed once as t goes from 0 to 1, σ_w^2 denotes the initialization variance of the student weights, i.e.,, $w(0)_i \sim \mathcal{N}(0, \sigma_w^2)$, $\mathbb{1}\{\cdot\}$ is an indicator function that is 1 when the argument is true and zero otherwise, and the density $\rho^{\mathrm{MP}}(\cdot)$ denotes the Marchenko-Pastur distribution [23, 27], which describes the eigenvalue distribution of the input correlations XX^T in the high-dimensional regime. It has the form

$$\rho^{\mathrm{MP}}(\lambda) = \frac{1}{2\pi} \frac{\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)}}{\lambda} + \mathbb{1}\{\alpha < 1\}(1-\alpha)\delta(\lambda) \tag{95}$$

for $\lambda = 0$ or $\lambda \in [\lambda_-, \lambda_+]$, and is zero elsewhere. The distribution comprises a delta function spike at zero, corresponding to zero-variance input directions that occur when there are fewer samples than the input dimension (i.e.,, $\alpha < 1$), and a bulk with upper and lower limits, $\lambda_\pm = (\sqrt{\alpha} \pm 1)^2$, that depend on the load α. We set $\sigma_w^2 = 0$ for most analyses in the paper.

We call this strategy memory-optimized, because Eqn. (93) is strictly decreasing in time, so to optimize student memory, replay should be continued indefinitely. However, Eqn. (94) is non-monotonic. Thus, while sustained replay optimizes student memory, this strategy can degrade generalization. Most problematically, it causes catastrophic overfitting at the student capacity, which corresponds to the interpolation threshold where the training error can just reach zero at long times. For a shallow linear student, the capacity is reached when the number of samples is equal to the input dimension, $\alpha = 1$. Better performance can be obtained for larger and smaller α, a finding known as the *double descent* phenomenon [23, 22, 4, 8]. The behavior of this strategy for a range of SNRs and loads α is depicted in Supplementary Fig. S2a,c. While memorization performance is good throughout this space, generalization suffers for low SNRs and loads near one.

6.3 Weight norm dynamics

While memory and generalization error are two key measures of learning progress, we can also ask how the strength of student weights change throughout learning. This quantity could enable certain experimental links, for instance, as a proxy for functional connectivity in the context of the Sweegers et al. [35] experiment discussed in Section 11.

A straightforward modification to the derivation in Section 2.1 of [4] yields the time-dependent average student weight norm as

$$\begin{aligned} \langle ||w(t)||_2^2 \rangle_Z &= N \int \rho^{MP}(\lambda) \left[\sigma_w^2 e^{-2\lambda t/\tau} + \left(\sigma_{\bar{w}}^2 + \tfrac{\sigma_\epsilon^2}{\lambda} \right) \left(1 - e^{-\lambda t/\tau}\right)^2 \right] d\lambda, \\ &= N \int \rho^{MP}(\lambda) \left[\sigma_w^2 e^{-2\lambda t/\tau} + \tfrac{1+\lambda\mathcal{S}}{\lambda(1+\mathcal{S})} \left(1 - e^{-\lambda t/\tau}\right)^2 \right] d\lambda. \end{aligned} \tag{96}$$

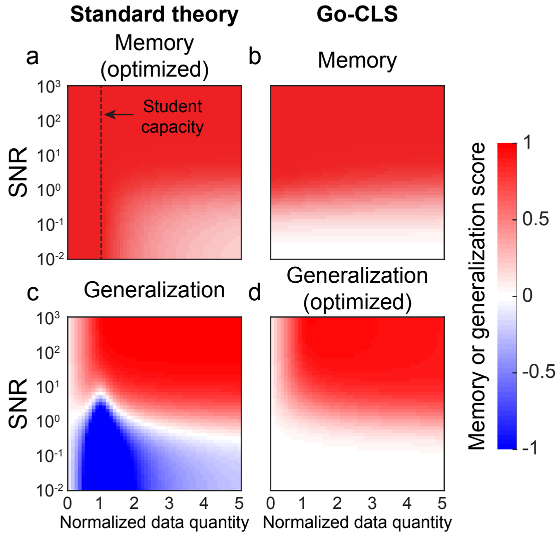


Figure S2: Heatmaps of student memorization performance (a, b) and generalization performance (c, d) as a function of SNR and α, when optimized for student memorization (a, c) or generalization (b, d).

Although we typically consider the case where $\sigma_w^2 = 0$, for large σ_w^2, this equation can describe an initial decrease in norm, followed by an increase in norm as weights align with the teacher.

6.4 Correlated training data and non-uniform memory reactivation

Here we numerically explore the effects of introducing input correlations and biased notebook sampling on the training and generalization error dynamics of the student.

In Section 6.1, the errors in notebook reactivation were caused by readout interference when retrieving the training patterns. This effectively introduced correlations in the training data set, and we were curious how correlations affected training dynamics more generally. When $\mathcal{S} = \infty$, increasing levels of correlation mainly made the generalization error decay slower (Fig. S3a). When $\mathcal{S} = 0.6$, correlated data caused more severe overfitting (Fig. S3b). Interestingly, while introducing correlations in training data generally increased the severity of overfitting, it did not the change our finding that the worst overfitting occurs when the normalized data quantity equals one ($\alpha = 1$) (Fig. S3c). These dynamics could potentially be studied analytically by replacing the Marchenko-Pastur distribution with the eigenvalue distribution of a random matrix ensemble containing uniform input correlations [41].

In Section 6.1, we also assumed that memories were reactivated with uni-

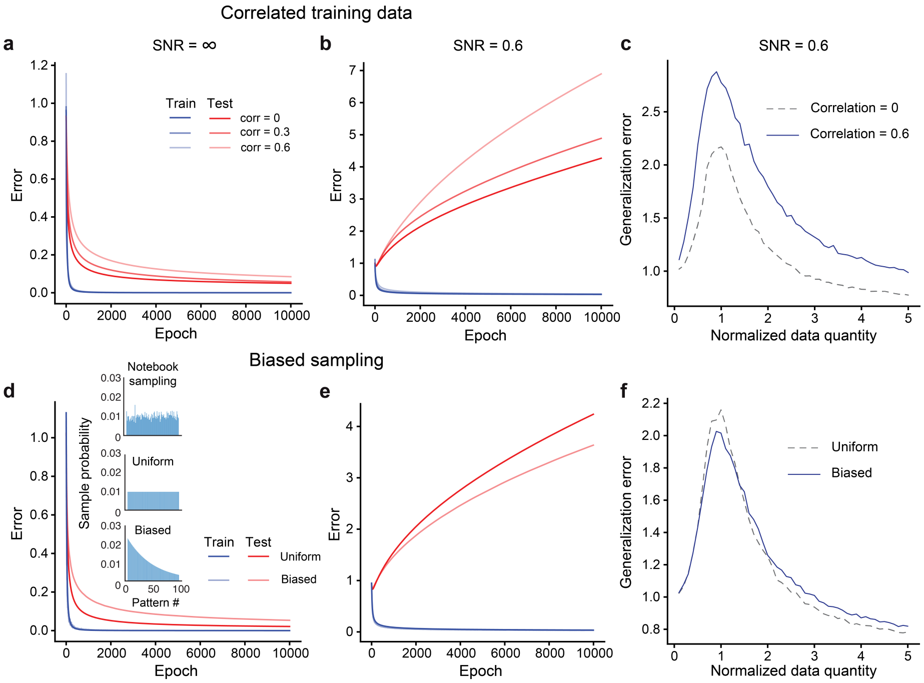


Figure S3: (a,b) Effect of introducing different levels of correlations among the training patterns on the training and generalization error dynamics for $\mathcal{S} = \infty$ and $\mathcal{S} = 0.6$, respectively. (c) Generalization error as a function of α for both i.i.d training data and correlated training data (generalization error was measured at epoch 2000). (d,e) Effect of biased sampling on training and generalization error dynamics for $\mathcal{S} = \infty$ and $\mathcal{S} = 0.6$, respectively. Biased sampling probabilities were chosen using an exponential function. Inset in d shows the sampling probabilities for training patterns when using the notebook's random reactivations (one example shown), uniformly sampling, and biased sampling. (f) Generalization error as a function of α for both uniformly sampled and biased sampled data. In all simulations, $P = N = 100$, and traces are the average of 50 independent runs.

form probability. This is a good model for the notebook we implemented (Fig. S3d, inset, top), but realistic memory-reactivation mechanisms might be more biased. We simulated non-uniform sampling of memories and found that biased sampling slowed down the rate of generalization error improvement on noiseless data with $P = N$ (Fig. S3d). On the other hand, it had a small effect in slowing down overfitting in the presence of noise (Fig. S3e). Similar to introducing correlations, biased sampling also did not change where worst overfitting occurred as a function of normalized data quantity (Fig. S3f). We observed subtle changes in the generalization performance under biased sampling compared to uniform sampling, as the trend of the change (i.e., increased or decreased rate of generalization error performance) switched sign as a function of normalized data quantity.

These results extend our main conclusion (i.e., overfitting is a problem when learning from a moderate amount of noisy data) to biased reactivation schemes and alternate noisy data ensembles. They additionally revealed that the severity of overfitting generally depends on more than the SNR and quantity of data, with more severe overfitting here appearing when the training data contained uniform input correlations. Future theoretical analyses could provide a fuller understanding on what other factors influence the dynamics and final results of systems consolidation.

7 Generalization-optimized Replay Policy

The generalization-optimized replay policy is similar to the memory-optimized replay policy. Samples are stored in the notebook and replayed to the student to drive learning. When it comes time to make a prediction, the system with the best error is used. The key difference, however, is that replay is not continued indefinitely. Instead, replay is terminated when generalization error stops improving and starts to worsen due to overfitting. That is, this strategy regulates replay to maximize generalization error.

7.1 Student memory and generalization error with regulated replay

In detail, this strategy continues replay until the optimal early stopping time t^*, defined as

$$t^* = \text{argmin}_t E_g^{\text{MO}}(t), \tag{97}$$

where $E_g^{\text{MO}}(t)$ is the memorization-optimized generalization error trajectory (i.e., unregulated trajectory) from Eqn. (94). The student memory and generalization error therefore have the piece-wise form

$$E_m^{\mathrm{GO}}(t) = \begin{cases} E_m^{\mathrm{MO}}(t) & t < t^* \\ E_m^{\mathrm{MO}}(t^*) & t \geq t^* \end{cases}, \tag{98}$$

$$E_g^{\mathrm{GO}}(t) = \begin{cases} E_g^{\mathrm{MO}}(t) & t < t^* \\ E_g^{\mathrm{MO}}(t^*) & t \geq t^* \end{cases}, \tag{99}$$

where the superscript GO stands for "generalization-optimized," and $E_m^{\mathrm{MO}}(t)$ denotes the memorization-optimized memory error trajectory from Eqn. (93). Crucially, under this regulated strategy, the student memory error can remain large indefinitely. Conversely, regulation avoids potentially catastrophic overfitting. The performance of the student under this strategy is depicted in Fig. S2b,d as a function of SNR $\mathcal{S}$ and load α. Finally, we note that weight norm dynamics have a similar piece-wise form for this strategy, such that the dynamics follow Eqn. 96 for $t < t^*$, at which point they stop.

7.2 Properties of early stopping

To gain a better understanding of this strategy, we can ask how the optimal stopping time depends on dataset parameters. While there is no closed form expression for t^*, some intuition can be obtained by computing the optimal stopping time for one fixed value of λ in the integral of Eqn. (94) (a strategy that would be exact if the MP distribution were a delta function at a single value of λ). The optimal stopping time is then (see Sec 2.2 of [4])

$$t^* = \frac{\tau}{\lambda} \log(\lambda \mathcal{S} + 1), \tag{100}$$

which shows that replay can continue longer for higher SNR relationships, though the relationship is logarithmic.

Early stopping is only one out of a variety of regularization strategies that can combat overfitting. Another possibility is to explicitly penalize large weight values. The L_2 regularization strategy sets the student weights according to

$$w^{L_2} = \mathrm{argmin}_w \left(\mathcal{E}_m(w) + \frac{\omega}{2} ||w||_2^2 \right), \tag{101}$$

where ω denotes the regularization strength. The optimal L_2 regularization strength for our setting is known to be inversely proportional to SNR, $\omega^{\mathrm{opt}} = 1/\mathcal{S}$ (see [2, 3, 4]). Further, for the specific teacher and student regression problem we consider here, this regularization is known to be Bayes optimal, such that no algorithm can outperform it [2, 3]. It therefore can serve as a normative standard of comparison for early stopping. Prior work has shown that, in our setting, early stopping closely approximates the effect of explicit L_2 regularization (see, e.g.,, Fig. 5a of [4]), providing a normative basis for the early stopping strategy.

Finally, we can exploit the similar performance of early stopping and optimal L_2 regularization to obtain an explicit (but approximate) expression for the performance of the generalization-optimized replay strategy after the early stopping time. In particular, for $t > t^*$ we have

$$\begin{aligned} E_g(t) &\approx E_g^{L_2} \quad \text{for } t \geq t^* \\ &= \frac{\mathcal{S}}{2(1+\mathcal{S})}\left(1-\alpha-1/\mathcal{S}+\sqrt{(1/\mathcal{S}+\alpha-1)^2+4/\mathcal{S}}\right) \\ &\quad +\frac{1}{1+\mathcal{S}}, \end{aligned} \tag{102}$$

where the latter step is the known generalization error of optimal L_2 regularization on this problem [2, 3, 4].

Using a similar approach, we can approximate the weight norm at the optimal stopping time as the weight norm of the optimal L_2 regularized solution (see Eqn. 66 [4]),

$$\begin{aligned} \left\langle \left\| w_{\text{opt}}^{L_2} \right\|_2^2 \right\rangle_Z &= \sigma_{\bar{w}}^2 \int \rho^{\text{MP}}(\lambda) \frac{\lambda}{\lambda + 1/\mathcal{S}} d\lambda, \\ &= \int \rho^{\text{MP}}(\lambda) \frac{\lambda \mathcal{S}^2}{(1+\mathcal{S})(1+\lambda\mathcal{S})} d\lambda, \end{aligned} \tag{103}$$

which we note limits to 1 as $\mathcal{S} \to \infty$ and 0 as $\mathcal{S} \to 0$, such that high-SNR relationships have larger weight norms than low-SNR relationships at the optimal stopping time.

8 Example of generalization non-limiting unpredictability

The main text provides several examples of generalization-limiting unpredictability, with the canonical example being a teacher with output noise. However, not all sources of unpredictability are generalization limiting. For example, suppose that the teacher generates noiseless data,

$$y = \bar{w}x, \tag{104}$$

but the student has internal noise in its input neurons that affects its predictions

$$\hat{y} = w(x + \eta). \tag{105}$$

Averaging over the input and noise distributions (but not the teacher weights), the generalization error of the student is

$$\mathcal{E}_g = \langle (y-\hat{y})^2 \rangle_{x,\eta} = \langle (y - \sum_i w_i(x_i+\eta_i))^2 \rangle_{x,\eta}$$

$$= \langle y^2 \rangle_x - 2\sum_i w_i \langle y(x_i+\eta_i)\rangle_{x,\eta} + \sum_i \sum_j w_i w_j \langle (x_i+\eta_i)(x_j+\eta_j)\rangle_{x,\eta}. \tag{106}$$

Assuming that x, y, and η are zero mean random variables, and that η is uncorrelated with x and y, this is equal to

$$\mathcal{E}_g = \sigma_y^2 - 2C_{yx}w + w^T(C_{xx} + C_{\eta\eta})w. \tag{107}$$

Setting the derivative with respect to w equal to zero,

$$0 = \frac{\partial \mathcal{E}_g}{\partial w} = -2C_{yx} + 2w^T(C_{xx} + C_{\eta\eta}) \tag{108}$$

we find that the student weights that optimize generalization are

$$w^* = (C_{xx} + C_{\eta\eta})^{-1}C_{xy} \tag{109}$$

In contrast, the teacher weights satisfy

$$C_{yx} = \bar{w}^T C_{xx} \Longrightarrow \bar{w} = C_{xx}^{-1}C_{xy}. \tag{110}$$

Since $w^* \neq \bar{w}$, the generalization-optimized student is statistically biased,

$$\langle \hat{y} \rangle_\eta - y = (w^* - \bar{w})^T x \neq 0, \tag{111}$$

and the generalization error is nonzero. The teacher is unpredictable by the student.

Nevertheless, this type of unpredictability does not require strongly regulated systems consolidation. For example, suppose that the notebook perfectly memorizes P input-output patterns of the student. Then, the memory error averaged over student neuron noise is

$$\begin{aligned}\langle \mathcal{E}_m \rangle_\eta &= \frac{1}{P}\sum_\mu \left\langle \left(y_\mu - \sum_i w_i(x_{i\mu} + \eta_i) \right)^2 \right\rangle_\eta \\ &= \frac{1}{P}\sum_\mu \left(y_\mu^2 - 2\sum_i w_i y_\mu x_{i\mu} + \sum_i \sum_j w_i w_j (x_{i\mu}x_{j\mu} + \langle \eta_i \eta_j \rangle_\eta) \right) \\ &= \frac{1}{P}\left(y^T y - 2yX^T w^T + w^T(XX^T + C_{\eta\eta})w \right).\end{aligned} \tag{112}$$

Setting the derivative with respect to w equal to 0,

$$0 = \frac{\partial \langle \mathcal{E}_m \rangle_\eta}{\partial w} = \frac{1}{P}\left(-2yX^T + 2w(XX^T + C_{\eta\eta}) \right) \tag{113}$$

we find that the weights minimizing the training error are

$$\hat{w} = yX^T(XX^T + C_{\eta\eta})^{-1}. \tag{114}$$

Noting that XX^T and yX^T are (proportional to) estimates of C_{xx} and C_{xy} given the P teacher examples, we see that this is the same basic form as the weights that minimize the generalization error.

In terms of the learning dynamics, the role played by eigenvalues of XX^T is now played by eigenvalues of $XX^T + C_{\eta\eta}$, which are lower bounded by the minimum eigenvalue of $C_{\eta\eta}$. For white noise, this is just σ_η^2. Overfitting was previously due to eigenvalues near 0, but those have now been shifted up to σ_η^2. The student input noise regularizes the learning process.

9 Means to regulate systems consolidation

Section 7 explained how generalization performance could be optimized by using the SNR of the teacher to regulate the amount of systems consolidation. In biology, the SNR is not known *a priori*, and the brain must decide for itself how to regulate consolidation. This section explores several plausible strategies that the brain could use to regulate systems consolidation. We emphasize strategies that serve to optimize generalization in the teacher-student-notebook framework.

9.1 Validation set approach

In the teacher-student-notebook framework, the notebook stores P teacher-generated examples at $t = 0$. We've so far assumed that the notebook reactivates all P examples with uniform probability to drive student learning. However, it's also possible for the notebook to divide its examples into separate training and validation sets. As before, the notebook could reactivate the training examples to drive student learning. However, if the validation examples are not reactivated to drive student learning, then they could instead be used to approximate the generalization error. Alternatively, a validation set can be used to train a separate, smaller student. A recent machine learning study shows that such a separately-trained model can be used to construct a score for ranking individual training sample's usefulness in improving generalization [29], which in turn can be used for regulating consolidation.

9.2 Maximum likelihood estimation

The above strategy used separate subsets of examples to drive learning and estimate the generalization error. Such a scheme could allow the brain the regulate systems consolidation by stopping student learning as soon as the generalization error begins to increase. An alternate strategy is to estimate the SNR of the teacher, $\mathcal{S}$, from the examples it provides, X and y. In this subsection, we calculate and characterize the maximum likelihood estimator,

$$\hat{\mathcal{S}} = \mathrm{argmax}_{\mathcal{S}} P(X, y|\mathcal{S}), \tag{115}$$

which is a statistically principled and asymptotically optimal unbiased estimator of $\mathcal{S}$. It will be convenient to replace the likelihood function, $P(X, y|\mathcal{S})$, with the log-likelihood function, $\log P(X, y|\mathcal{S})$, because the logarithm does not change the location of maximum,

$$\hat{\mathcal{S}} = \mathrm{argmax}_{\mathcal{S}} \log P(X, y|\mathcal{S}), \tag{116}$$

and it is often mathematically convenient to work with log-transformed functions. For example, by the product rule of probability, we have

$$P(X, y|\mathcal{S}) = P(X|\mathcal{S})P(y|X, \mathcal{S}) = P(X)P(y|X, \mathcal{S}), \tag{117}$$

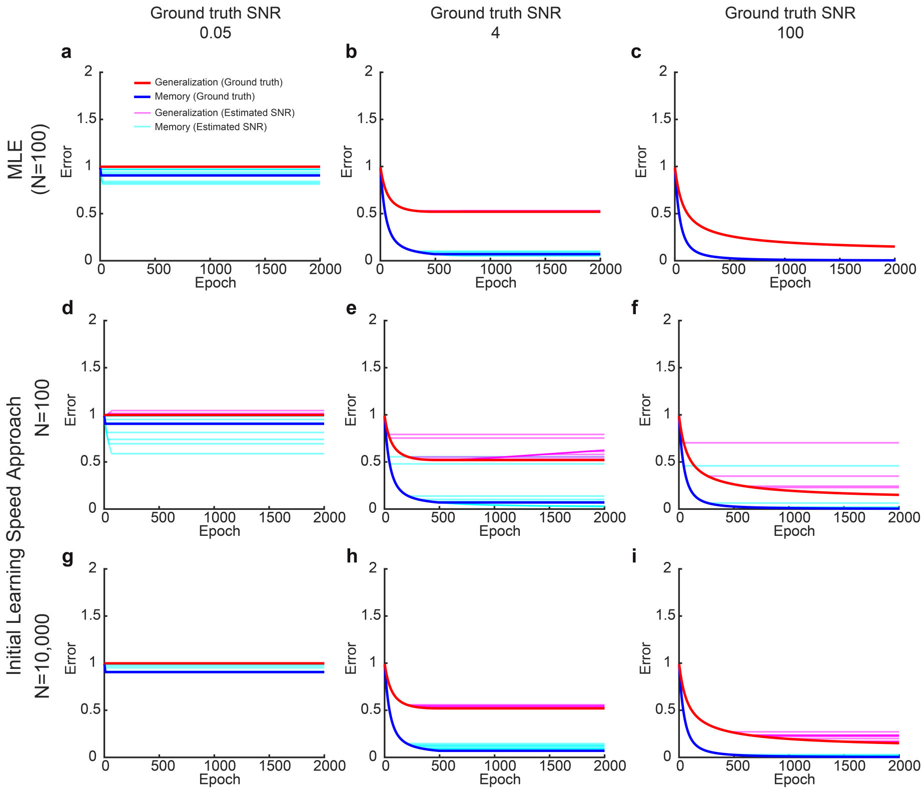


Figure S4: (a, b, c) Generalization and memorization error dynamics with early stopping computed according to the ground truth SNR and MLE-estimated SNR. $\mathcal{S} = \infty$, $\mathcal{S} = 4$, and $\mathcal{S} = 100$, respectively. $N = P = 100$. (d, e, f and g, h, i) Similar to panels a, b, c, but using the learning speed approach to estimate the SNR and corresponding early stopping times. Here the model dimensions were $N = P = 100$ for panels d, e, f and $N = P = 10,000$ for panels g, h, i. In all panels, 10 independent simulations were performed for estimating the SNR.

where we noted that $P(X)$ is independent of $\mathcal{S}$ by assumption. Therefore,

$$\log P(X, y|\mathcal{S}) = \log(P(X)) + \log(P(y|X, \mathcal{S})), \tag{118}$$

and

$$\hat{\mathcal{S}} = \mathrm{argmax}_{\mathcal{S}} \log(P(y|X, \mathcal{S})), \tag{119}$$

where we discarded $\mathcal{S}$-independent additive factors in the last step.

We next derive an expression for $P(y|X, \mathcal{S})$. It is convenient to first note that $P(y|X, \mathcal{S})$ is the marginal of $P(y, w|X, \mathcal{S})$ over w:

$$P(y|X, \mathcal{S}) = \int d^N w P(y, w|X, \mathcal{S}). \tag{120}$$

Again using the product rule of probability, we find

$$P(y, w|X, \mathcal{S}) = P(w|X, \mathcal{S})P(y|X, w, \mathcal{S}). \tag{121}$$

Both of these probability distributions are easy to specify. First, elements of w are i.i.d. distributed as $w_i \sim \mathcal{N}(0, \frac{\mathcal{S}}{1+\mathcal{S}})$ by assumption, so

$$P(w|X, \mathcal{S}) = P(w|\mathcal{S}) \propto \exp\left(-\frac{1+\mathcal{S}}{2\mathcal{S}} ww^T\right), \tag{122}$$

where the normalization constant was neglected for mathematical conciseness and will be put back in later. Second, note that elements of y are normally distributed with mean wX and variance determined by the noise, which is i.i.d. distributed as $\eta_\mu \sim \mathcal{N}(0, \frac{1}{1+\mathcal{S}})$. Therefore,

$$P(y|X, w, \mathcal{S}) \propto \exp\left(-\frac{1+\mathcal{S}}{2}\left(y - wX\right)\left(y - wX\right)^T\right), \tag{123}$$

where the neglected normalization constant will again be included later. Putting these two pieces together, we can easily see that $P(y, w|X, \mathcal{S})$ is a zero-mean multivariate Gaussian distribution with an $\mathcal{S}$-dependent covariance structure. In particular, the arguments of the exponential factors in Eqs. (122) and (123) combine to give

$$\begin{aligned} P(y, w|X, \mathcal{S}) \propto & \exp\left(-\frac{1+\mathcal{S}}{2\mathcal{S}} ww^T - \frac{1+\mathcal{S}}{2}\left(y - wX\right)\left(y - wX\right)^T\right) \\ =& \exp\Big(-1/2\Big((1+\mathcal{S})yy^T + w\left(\frac{1+\mathcal{S}}{\mathcal{S}} I + (1+\mathcal{S})XX^T\right)w^T \\ & - 2(1+\mathcal{S})yX^T w^T\Big)\Big), \end{aligned} \tag{124}$$

which shows that each term in the exponential is second-order in y and w. This allows us to use the general Gaussian integral formula to integrate over w,

$$\int d^N w \, \exp\left(-\frac{1}{2} wAw^T + B^T w^T\right) = \sqrt{\frac{(2\pi)^N}{\det A}} \exp\left(\frac{1}{2} B^T A^{-1} B\right), \tag{125}$$

with

$$A = \frac{1+\mathcal{S}}{\mathcal{S}} I + (1+\mathcal{S}) X X^T, B = (1+\mathcal{S}) X y^T. \tag{126}$$

Consequently, $P(y|X, \mathcal{S})$ is again a multivariate normal distribution, this time with exponential factor given by

$$\begin{aligned} P(y|X,\mathcal{S}) \propto \exp\Big(&- \frac{1+\mathcal{S}}{2} y y^T \\ &+ \frac{(1+\mathcal{S})^2}{2} y X^T \left(\frac{1+\mathcal{S}}{\mathcal{S}} I + (1+\mathcal{S}) X X^T \right)^{-1} X y^T \Big) \\ = \exp\Big(&- \frac{1+\mathcal{S}}{2} y y^T \\ &+ \frac{(1+\mathcal{S})}{2} y X^T \left(I/\mathcal{S} + X X^T \right)^{-1} X y^T \Big) \\ = \exp\Big(&- \frac{1}{2} y C^{-1} y^T \Big), \end{aligned} \tag{127}$$

where

$$C^{-1} = (1+\mathcal{S}) \Big(I - X^T \left(I/\mathcal{S} + X X^T \right)^{-1} X \Big) \tag{128}$$

is the inverse covariance matrix. Putting back in the normalization factors, we find

$$P(y|X,\mathcal{S}) = \frac{1}{\sqrt{(2\pi)^P \det C}} \exp\Big(- \frac{1}{2} y C^{-1} y^T \Big) \tag{129}$$

and thus

$$\hat{\mathcal{S}} = \text{argmax}_{\mathcal{S}} \left(-\frac{1}{2} \log \det C - \frac{1}{2} y C^{-1} y^T \right), \tag{130}$$

where C is a function of $\mathcal{S}$ and X.

It is convenient and conceptually clarifying to rewrite the covariance matrix in Eq. (128) in terms of the singular value decomposition of X,

$$X = U \Lambda^{1/2} V^T, \tag{131}$$

where U and V are orthogonal matrices and Λ is non-negative rectangular diagonal matrix. In particular, it implies that

$$\begin{aligned} C^{-1} =& (1+\mathcal{S}) \left(I - V \Lambda^{1/2} U^T \left(I/\mathcal{S} + U \Lambda^{1/2} V^T V \Lambda^{1/2} U^T \right)^{-1} U \Lambda^{1/2} V^T \right) \\ =& (1+\mathcal{S}) V \left(I - \Lambda^{1/2} \left(I/\mathcal{S} + \Lambda \right)^{-1} \Lambda^{1/2} \right) V^T \end{aligned} \tag{132}$$

and

$$C = \frac{1}{1+\mathcal{S}} V \left(I - \Lambda^{1/2} \left(I/\mathcal{S} + \Lambda \right)^{-1} \Lambda^{1/2} \right)^{-1} V^T \tag{133}$$

Using the fact that $\Lambda_{ab} = \lambda_a \delta_{ab}$ is a rectangular diagonal matrix, this expression can be significantly simplified by recognizing that

$$\begin{aligned}\left(I - \Lambda^{1/2}\left(I/\mathcal{S} + \Lambda\right)^{-1}\Lambda^{1/2}\right)^{-1}_{ab} &= \left(1 - \frac{\lambda_a}{1/\mathcal{S} + \lambda_a}\right)^{-1}\delta_{ab} \\ &= \left(\frac{1/\mathcal{S}}{1/\mathcal{S} + \lambda_a}\right)^{-1}\delta_{ab} = (1 + \lambda_a \mathcal{S})\delta_{ab}. \quad (134)\end{aligned}$$

In particular, we see that

$$C = V\frac{(I + \mathcal{S}\Lambda)}{1 + \mathcal{S}}V^T = \frac{I + \mathcal{S}X^T X}{1 + \mathcal{S}}. \quad (135)$$

We show in the main text that this estimation scheme accurately tracks the ground truth SNR (Fig. 2l). Figs. S4a-c additionally show that the estimated SNR leads to accurate early stopping times, generalization dynamics, and memorization dynamics.

9.3 Learning speed approach

Finally, we consider a simple heuristic based on the initial rate of improvement in a task. Intuitively, tasks with easy-to-memorize data points are those with an underlying pattern that supports generalization (i.e., are high SNR). To formalize this intuition in our setting, we consider the initial slope of the student memory error in Eq. (93),

$$\begin{aligned}\frac{d}{dt}E_m^{\mathrm{MO}}(0) &= \frac{1}{\alpha}\int \rho^{MP}(\lambda)\left(\frac{1 + \lambda\mathcal{S}}{1 + \mathcal{S}} + \lambda\sigma_w^2\right)\frac{d}{dt}e^{-\frac{2\lambda t}{\tau}}d\lambda\bigg|_{t=0} \\ &= -\frac{2}{\tau\alpha}\int \rho^{MP}(\lambda)\left(\frac{\lambda + \lambda^2\mathcal{S}}{1 + \mathcal{S}} + \lambda^2\sigma_w^2\right)d\lambda. \quad (136)\end{aligned}$$

For a given α, τ, and σ_w^2, this expression shows that the initial slope depends on the SNR, $\mathcal{S}$. Therefore, replay can be regulated by measuring the initial slope, a quantity immediately available to the agent, and reading off the associated SNR. The bottom panel of Fig. 2m illustrates this relationship for one set of parameters, which is approximately linear in the logarithm of the SNR, and Fig. 2n shows that the estimated SNR is reasonably accurate. In Figs. S4d-i, we additionally used the learning speed approach to estimate early stopping times, and we found that the resulting memorization and generalization dynamics could closely match those found using the ground truth SNR. We note that this agreement required that the model be sufficiently high dimensional, but our model remains small compared to the biological brain. This heuristic is representative more broadly of a class of approaches that monitor training trajectories as a way of estimating generalization performance.

10 Complex teachers

10.1 Linear Student

Here we show that a mismatch between the teacher and student, such that the teacher is deterministic but more complex than the student, is a form of generalization-limiting unpredictability that behaves similarly to observing a teacher with noise. Our derivation follows Appendix C of [4], but we include a derivation of this important result for completeness.

Suppose the teacher generates inputs independently from some distribution $x \sim p(x)$, and labels them using the possibly nonlinear function $y = g(x)$. The best possible linear student (i.e.,, the student trained on infinite data) will have weights

$$\hat{w}_{\text{OPT}} = C_{yx} C_{xx}^{+} \tag{137}$$

where $C_{yx} = \left\langle yx^T \right\rangle_{x,y}$ is the input-output correlation matrix, $C_{xx} = \left\langle xx^T \right\rangle_{x}$ is the input correlation matrix, and C_{xx}^{+} denotes its pseudoinverse. We can rewrite the teacher output as the prediction of this optimal student and a residual,

$$y = C_{yx} C_{xx}^{\dagger} x + \delta y, \tag{138}$$

where the residual, δy, is defined by this expression.

Next, we consider learning the student weights from a finite batch of data with P examples, given in matrices Y, X with examples in the columns. The student weights that minimize the training error are

$$\begin{aligned} \hat{w}_{\text{LS}} &= YX^T \left(XX^T\right)^{+} && (139) \\ &= \hat{w}_{\text{OPT}} + \delta Y X^T \left(XX^T\right)^{+} && (140) \end{aligned}$$

where we substituted Eq. 138 for Y, and $\delta Y = Y - C_{yx} C_{xx}^{+} X$ is the matrix of residuals. This formulation clearly separates contributions to the student weights into an optimal component and an overfitting component. Notably, the overfitting term $\delta Y X^T \left(XX^T\right)^{\dagger}$ has the same form as for additive noise. This noise is generally non-Gaussian, but the training and generalization errors equal those of a Gaussian teacher with mean and variance matched to δY.

10.2 Nonlinear student

In order to test if our theoretical conclusions generalize beyond the linear student setting in regression tasks, we extended our numerical experiments to nonlinear student networks solving real-world classification tasks. More specifically, we trained deep convolutional neural networks (CNNs) to classify images from the MNIST [24], CIFAR-10 [21], and Tiny ImageNet [15] data sets. We used a simple CNN for the MNIST dataset (2 {convolutional + pooling} layers followed by 3 fully connected layers) and ResNet-18 [18] for the CIFAR-10 and Tiny ImageNet datasets (see Fig. S5 associated code for simulation details). For CIFAR-10 and

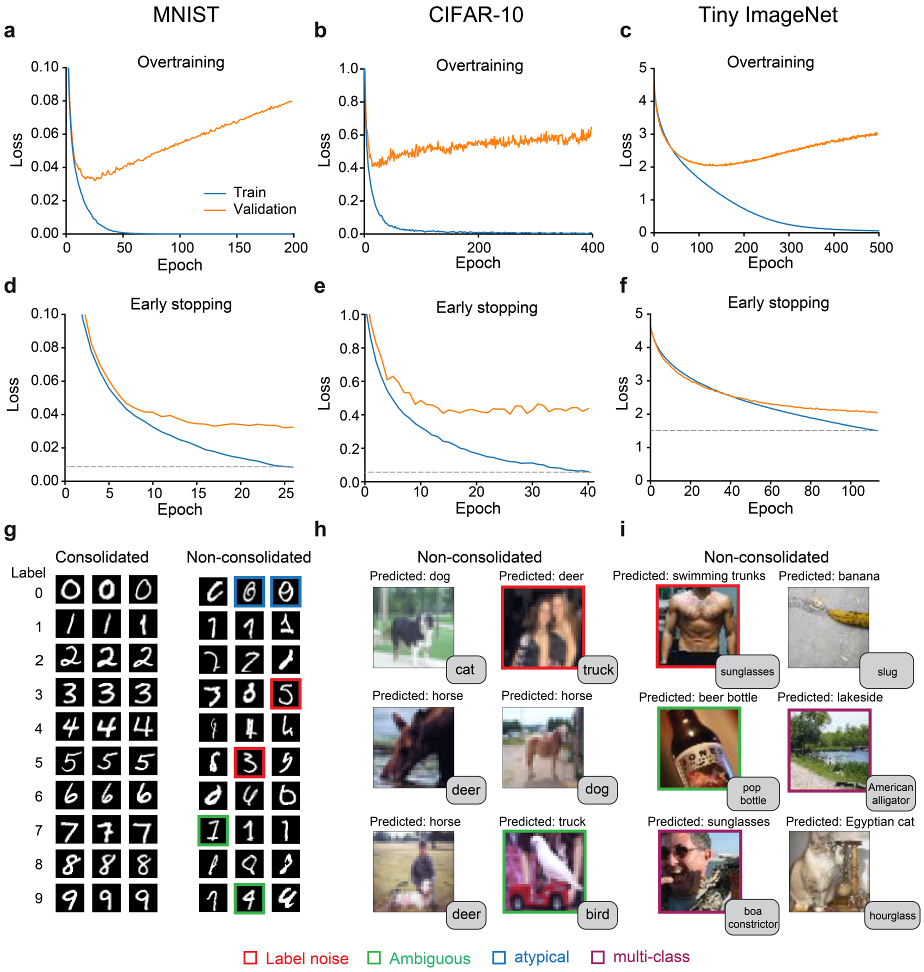


Figure S5: Deep Convolutional Neural Networks (CNN) trained on MNIST, CIFAR-10, and Tiny ImageNet show qualitative similar behavior of overfitting and memorization-generalization trade-off, compared to simpler linear models. (a-c) Deep CNNs can overfit to training data in all three datasets. (d-f) Overfitting can be prevented by early stopping. Gray dashed lines mark the non-zero training loss when performing early stopping. (g) Example training examples that are corrected classified (consolidated) (for g only) versus incorrectly classified (non-consolidated) after early stopping. For h and i, the text box at the bottom right corner of the image shows the ground truth label.

Tiny ImageNet training, data augmentations including random crop, rotation, and horizontal flip were introduced during training. Unaltered examples were used for evaluating training and testing performances.

Interestingly, we found that Go-CLS theory's essential conclusion still holds for these nonlinear teacher and student cases (Fig. S5). In particular, overtrained networks could achieve full memorization of the training data (100% training accuracy), but these models showed overfitting (Fig. S5a-c), reflected by the increased validation loss after reaching a minimum. Similar to the linear student network, early stopping was implemented by detecting when the validation loss stopped improving for 8-15 epochs, thereby preventing overfitting and resulting in a non-zero training loss (Fig. S5d-f). This reiterates the main message of Go-CLS theory: neural networks trained for perfect memory performance suffer in generalization, and generalization can be improved by regulating the consolidation process according to some regularization scheme (such as early stopping, dropout, or weight decay).

A novel aspect of these real-world data sets is that individual examples can have different noise characteristics. Examining the training examples that were correctly classified (consolidated) versus incorrectly classified due to early stopping (non-consolidated) provides interesting hints about the nature of the data that can harm generalization if fully consolidated. Taking the MNIST task as an example, the consolidated digits are easily recognizable as 'canonical' examples of each class (Fig. S5g, left), with variations that typically did not cause ambiguity (e.g., 7 vs 7 with a slash through it). By contrast, non-consolidated digits exhibited several distinct forms of unpredictability. First, there are clearly mislabeled data in the MNIST dataset (digits with red boxes). These digits act as noise during training, mirroring the "noisy teacher" discussed in the main paper (Fig. 5b). Second, many digits seemed to be written in ambiguous ways or atypical ways (e.g., examples of 0 with a slash through it). This suggests that more information is sometimes needed to determine the right label (i.e., about the writer's penmanship style), making these examples akin to the "partially observable teacher" (Fig. 5d). Similarly mislabeled and ambiguous data are also abundant for the unconsolidated images in CIFAR-10 and Tiny ImageNet datasets (S5h, i). These data sets additionally contain non-consolidated images that contain multiple distinct objects within the same scene. For example, one image contains both a cat and a hourglass, but the network's classification of "Egyptian cat" is counted as incorrect.

11 Comparison of Go-CLS theory to past experimental results

Go-CLS theory generates a diversity of amnesia curves that might help memory researchers explain the similar diversity found experimentally (Fig. S6). Researchers usually classify hippocampal amnesia dynamics according to whether memory deficits are similar for recent and remote memories (flat retrograde amnesia), more pronounced for recent memories (graded retrograde amnesia), or absent for both recent and remote memories (no retrograde amnesia) (Fig. S6a). Since real world experiences are composed of many elements that differ in their degree of predictability (Fig. 5j), our theory predicts that different components of human memory will consolidate to different degrees (Fig. 5k). In human memory research, patients with selective hippocampal damage indeed show retrograde amnesia reflecting diverse dynamics of systems consolidation [42, 17]. Some patients show graded retrograde amnesia consistent with the standard theory, while others either have flat retrograde amnesia or no retrograde amnesia [42] (Fig. S6b). Similarly diverse retrograde amnesia curves have been seen in rodent memory tasks (Fig. S6c-f). For example, hippocampal lesions can result in either graded or flat retrograde amnesia in different individuals performing the same task [39, 34, 40] (Fig. S6c-e), and individual animals can exhibit different types of amnesia on different tasks [40] (Fig. S6e,f).

Go-CLS theory recasts this wide range of experimental observations through the tuning of two parameters (Fig. 3e), the predictability of experience and the amount of prior consolidation. It is not yet possible to unambiguously specify these parameters for arbitrary real-world experiences and experimental memory tasks, but the empirical patterns are plausibly consistent with Go-CLS theory. For example, famous faces and facts about public events are generally reliable components of many life experiences, and one need not conjure up a specific past experience to remember what Barack Obama looks like or that the COVID-19 pandemic stunned the global economy. Memories of famous faces and facts about public events may thus represent content that is highly predictable across experiences, in which case Go-CLS would predict that they can be consolidated. Indeed, many patients can recall remote facts and famous faces without a functioning hippocampus [42] (Fig. S6b). In contrast, autobiographical memories combine idiosyncratic details about specific experiences in one's life that may not generalize to other experiences. For example, one often needs to think back to the original experience to remember the cake served at their child's birthday party or the songs played at their wedding. Because many incidental influences shape how complex real-life events unfold, remembering autobiographical memories may require the recall of content that is intrinsically unpredictable. Go-CLS predicts that unpredictable content will not be consolidated, and most patients cannot recall these memories without a hippocampus [42] (Fig. S6b). Along similar lines, the Morris water maze task consistently requires the hippocampus [40, 11, 31] (Fig. S6f), perhaps suggesting that rodents need to recall past experiences to reconstruct the detailed arrangement of environmental cues and platform positions [30, 33], both chosen arbitrarily by the experimenter. Rigorously assessing these *post hoc* interpretations will require theoretical and experimental progress on the algorithms used for predictability estimation.

Go-CLS theory can also generate diverse time courses for time-dependent generalization that mimic experimental diversity [39, 38, 14, 9]. For example, some mice showed increased fear responses to similar but not identical contexts in fear-conditioning experiments ("generalizers", Fig. S6g, red bars), while others maintained distinct behavioral responses over time ("discriminators", Fig.

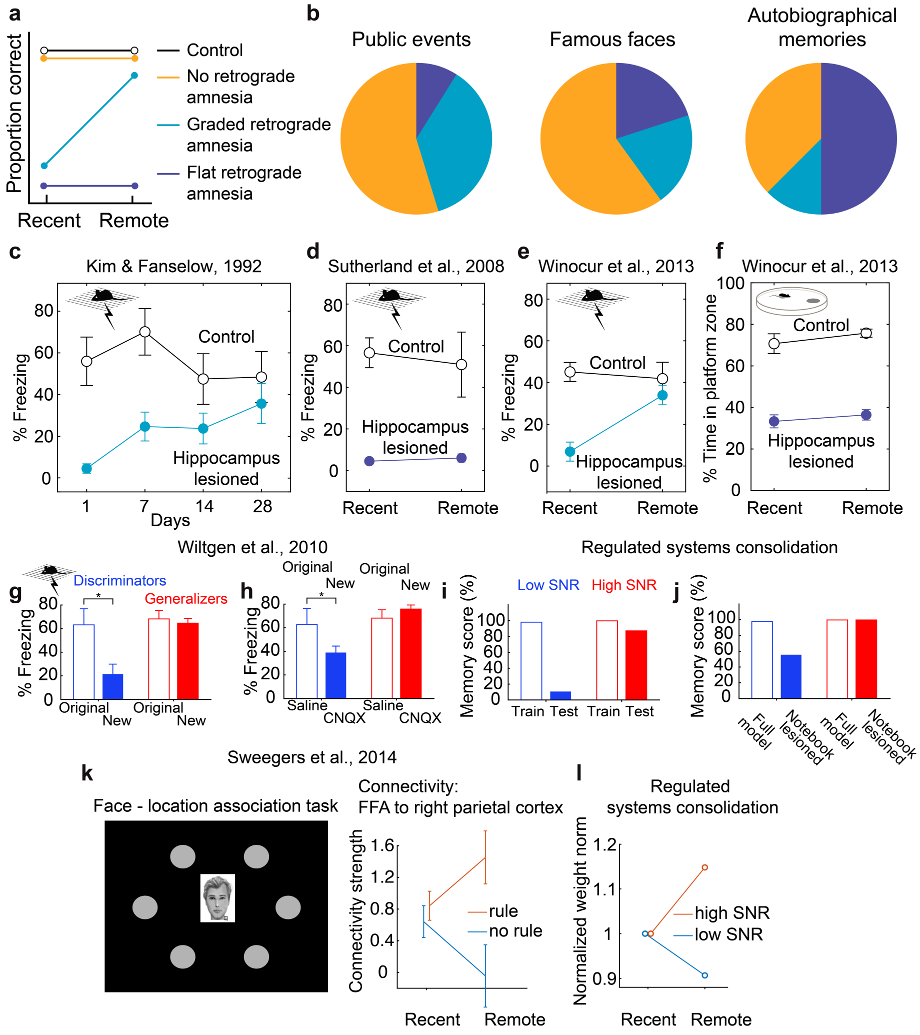


Figure S6: Caption continued on next page.

Figure S6: Diverse findings in memory research. (a) Schematic of retrograde amnesia curves. (b) Reports of retrograde amnesia in human patients with selective hippocampal damage show diverse dynamics. Figure adapted from Yonelinas et al., 2019 [42]. (c, d) Lesioning hippocampus in rodents can produce both graded and flat retrograde amnesia. Figure adapted from Kim & Fanselow, 1992 [20], Sutherland et al., 2008 [34]. Lesioning the hippocampus can result in graded (e) or flat (f) retrograde amnesia in the same animal performing different tasks (contextual fear conditioning and Morris water maze, respectively). Figure adapted from Winocur et al., 2013 [40]. (g) Discriminators can differentiate the original fear-conditioning context with another similar but novel context, whereas generalizers show similar amount of fear response to both contexts. (h) Silencing the hippocampus in mice 15 days after contextual fear conditioning differentially impact fear memory of the original context, depending on whether the animal show time-dependent fear generalization. panels g and h are adapted from Wiltgen et al., 2010 [39]. (i, j) Regulated systems consolidation can reproduce similar correlation between time-dependent generalization and reduced hippocampal dependence of memories. High SNR (1000) and low SNR (0.6) simulations based on analytical solutions are used to model the "generalizers" and "discriminator". 2000 total epochs are simulated with $N = P = 100$, notebook size $M = 5000$, and learnrate = 0.005. (k) Face-location association task with rules vs no rules show different time-dependent change in functional connectivity between cortical areas. Figure adapted from Sweegers et al., 2014 [35]. (l) Regulated systems consolidation shows similar connectivity changes over time, as reflected in the norm of the student's weights. Student weight w is drawn i.i.d. from $\mathcal{N}(0, 0.5)$, where the weights' non-zero initial condition reflect the brain's preexisting connectivity between these two regions. The student then learns from a high SNR teacher (SNR = 2) or a low SNR teacher (SNR = 0.05), while the weight norm is monitored through time (normalized to the initial norm). Note that a decrease in weight norm is expected on the low-SNR learning task, as a large weight norm generates substantial output variance that is uncorrelated with the teacher's noisy output. 2000 total epochs are simulated with $N = P = 100$, notebook size $M = 2000$, and learnrate = 0.015.

S3g, blue bars) [39]. Strikingly, only the discriminators required their hippocampus for memory recall of the original context (Fig. S6h). Although other interpretations are possible, it's intriguing that Go-CLS theory predicts that memory transfer and generalization improvement should be similarly correlated (Fig. 3f, S6i, S6j). In this interpretation, "discriminators" might judge fear conditioning to be an unpredictable experience that should not consolidate because this would cause maladaptive generalization. Their memories would thus be left in original form and be susceptible to strong retrograde amnesia (Fig. S6i, j, blue bars). In contrast, "generalizers" might infer that the experience is predictable, which would then lead to consolidation, weak retrograde amnesia, and learned generalizations (Fig. S6, red bars). This variability across individuals is possibly due to differences in each animal's regulation process (Fig. 2i-m) or feature encoding (Fig. 5i).

An experiment closely related to Go-CLS theory was performed by Sweegers et al. [35]. In their task design, healthy human participants had to associate specific faces with positions on a computer screen (Fig. S6k, left). Half of the locations were assigned faces through an unpredictable random process, whereas the other locations were assigned faces according to a hidden but fully reliable rule. The authors then used functional magnetic resonance imaging (fMRI) to assess how systems consolidation changed the functional connectivity between several brain areas during memory recall. They specifically asked whether functional connectivity patterns revealed statistical interactions between the association type (faces assigned to rule locations versus no-rule locations) and time (recall at recent versus remote time points). We subsequently refer to these statistical interactions as rule/time interactions.

Sweegers et al. found selective cortical recruitment that is consistent with the general premise of Go-CLS theory. More specifically, they detected statistically significant rule/time interactions in the functional connectivity from the hippocampus to a region containing parts of the anterior cingulate cortex (ACC) and medial prefrontal cortex (mPFC) [35]. This suggests that the time course of system consolidation depends on the predictability of the consolidated information. A *post hoc* analysis revealed similar rule/time interactions in the functional connectivity from this ACC/mPFC region to the fusiform face area (FFA). These findings collectively indicate enhanced ACC/mPFC connectivity for the rule locations at the remote time point, consistent with the idea that systems consolidation is regulated to recruit neocortical computations when the predictability of experience is high.

Intriguingly, Sweegers et al. also described a trend in their data that could quantitatively link their experimental paradigm to Go-CLS theory. In particular, when they investigated how functional connectivity from the fusiform face area (FFA) differed at recent and remote time points, they found that its connectivity to right parietal cortex increased for the rule-based locations and decreased for the no-rule locations (Fig. S6k, right) [35]. This result is expected from Go-CLS theory (Fig. S6l). The right parietal cortex is involved in spatial processing, and we interpret its functional connectivity from FFA in the teacher-student-notebook model as student weights used to predict neural activity coding location from neural activity coding faces. Our theory predicts that the predictability of the face-location relationship determines whether systems consolidation drives neocortical learning that links FFA to right parietal cortex. Indeed, these connections strengthened only when the face-location relationship was predictable. This empirical difference can be quantitatively captured by regulated systems consolidation (Fig. S6l). However, the authors did not test the statistical significance of this trend, because earlier analyses failed to detect any rule/time interactions in the functional connectivity from FFA. As such, the authors were worried that subsequent statistical tests would have inaccurately inflated p-values. It would be very interesting to see whether this trend is significant in a replication of Sweegers et al.

Predictable rules may similarly enhance consolidation in real-world situations. For example, a well-known study from Maguire et al. studied a licensed London taxi driver with bilateral hippocampal damage and found that this patient was only able to navigate London using its main arteries [26]. Consistent with the premise of Go-CLS, the authors suggested that the hippocampus is required for navigating "roads in a very unpredictable and irregular layout" but not those in a "predictable, regular (grid-like) layout" [26].

12 Experimentally testable prediction of Go-CLS

In this section, we provide some predictions of the Go-CLS theory and outline a framework for designing experiments to test these predictions. The core feature of these testable predictions is that subjects performing any task should consolidate information that is highly predictable, but not information that is unpredictable. Thus, any experimental design requires a way to vary the predictability of an input-output task and to measure the amount of consolidation that occurs during learning. Having accomplished this, it should be possible to design experiments that probe the mechanistic implementation of regulated consolidation, provided that the initial experiments are indeed consistent with Go-CLS. In theory, experiments can be done on any species capable of learning input-output tasks with variably predictable relationships. Here, we focus on mammalian species with hippocampal and neocortical brain regions (e.g.,, humans, non-human primates, rodents, etc.), as consolidation is presumed to occur through interactions between these two brain regions. Below, we provide a general recipe for designing such tasks. We presume that the details of experimental design will be determined by domain experts who wish to perform these types of experiments.

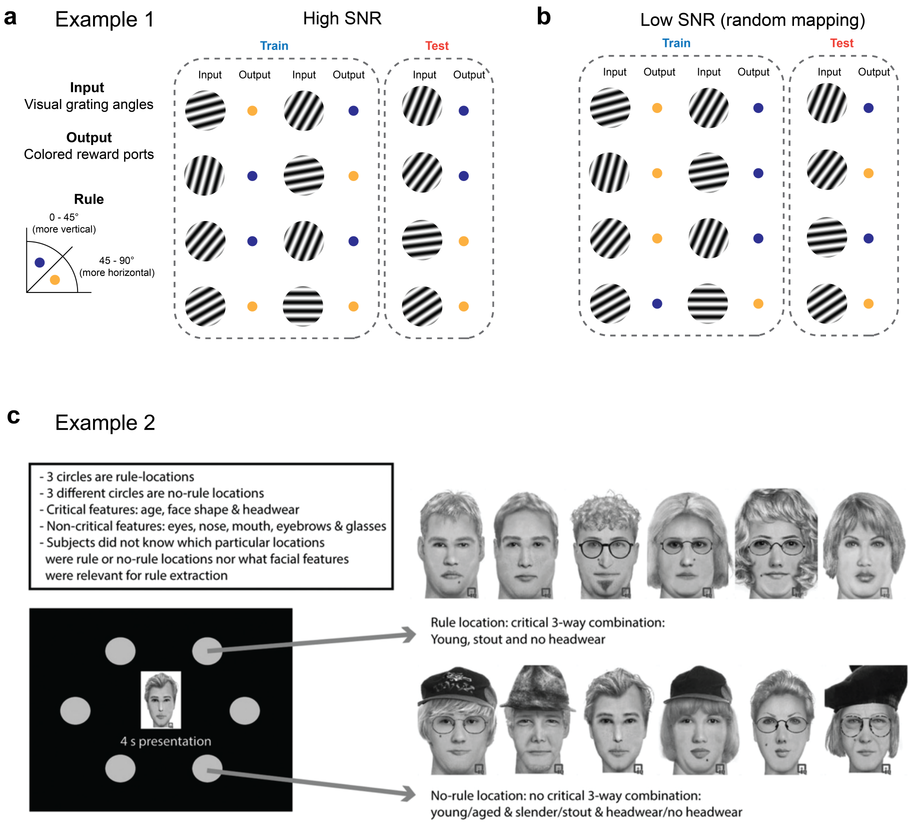
12.1 Testing the relationship between predictability and consolidation

The essential component is a behavioral task consisting of an input-output relationship designed by the experimenter and learned by the subject through experience. To design such a task, the experimenter must first choose inputs (e.g.,, visual cues), outputs (e.g.,, sound frequencies), an action to indicate the output (e.g.,, movement to target), a relationship between the inputs and outputs that can take predictable and unpredictable forms, and a reward (or removal of aversive stimulus) that serves as the driver for the subject to learn the input-output relationship. Predictable here means that if the subject sees either an old or a novel cue, it can use a rule learned from past trials to make a good prediction about how to respond correctly. The unpredictable version of the task lacks a systematic input-output relationship, and the only strategy for good performance is to memorize which cues are rewarded and which ones are not. Thus, we refer to the predictable and unpredictable versions of the task as *rule-based* and *memory-based* tasks, respectively. In addition to these two versions of the task, levels of generalization performance should be quantifiable by testing the performance on novel experiences following the same generative process. This is a critical piece of information on whether the animal is indeed consolidating predictable information in the manner we have originally designed.

Furthermore, there must be ways to infer the extent of systems consolidation. A key method is to determine if performance is impaired by hippocampal dysfunction (preferably reversibly) at remote time points. Our theory predicts the extent of memory preservation after hippocampus dysfunction should be correlated with the predictability of experience. It is important to ensure that the rule-based and memory-based tasks initially require similar hippocampal involvement, or else consolidation may not be required. This can be done by checking that hippocampal dysfunction impairs both task versions at recent time points. Manipulations such as lesions and reversible silencing (chemical, optogenetic, chemogenetic) can be applied to experimental animals (e.g.,, rodents) at different time points post-learning. This is more difficult in humans because of the constraints associated with using invasive or genetic tools for hippocampal perturbation, but it might be possible with functional imaging methods. For example, following systems consolidation of rule-based (highly predictable) experience (i.e.,, in a well-trained subject), memory recall is predicted to not require the hippocampus, and memory recall may thus engage the hippocampus more weakly. Strong neocortical activation must occur while the subject performs the task, and we expect higher within-neocortex functional connectivity. In contrast, weaker within-neocortex functional connectivity would be expected during performance of a memory-based (unpredictable) version of the task than the rule-based version, which relies more on reactivation of hippocampal memories to make predictions. Alternatively, perturbations of the hippocampus could be potentially achieved in humans through fMRI-guided transcranial magnetic stimulation (TMS) [36].

In the following, we provide two example experiments following the above-prescribed recipe:

Example 1: Visual gratings presented at various angles are used as the inputs and colored reward ports are used as outputs. Animals can indicate their choice by licking at a particular reward port (Fig. S7a). A relationship with varying degrees of predictability can be introduced to link the input and output. For example, the high predictability version of the relationship could be



- 3 circles are rule-locations
- 3 different circles are no-rule locations
- Critical features: age, face shape & headwear
- Non-critical features: eyes, nose, mouth, eyebrows & glasses
- Subjects did not know which particular locations were rule or no-rule locations nor what facial features were relevant for rule extraction

Rule location: critical 3-way combination:
Young, stout and no headwear

No-rule location: no critical 3-way combination:
young/aged & slender/stout & headwear/no headwear

Figure S7: Example experiments for testing Go-CLS. (a, b) Illustrations of a rule-based and a memory-based input-output task. (c) Task design for a face-location task used in Sweegers et al., 2014 [35].

that angles from 0° to 45° correspond to licking at the blue reward port, angles from 45° to 90° correspond to licking at the yellow reward port (Fig. S7a). For a task with low predictability, the mapping between angles and reward ports could be selected randomly (i.e.,, no systematically predictable relationship between angle and reward port, but the mapping could be memorized) (Fig. S7b). Our theory predicts that, over time, the hippocampus is no longer required to solve the rule-based (highly predictable) version of the task due to full consolidation (i.e.,, hippocampal silencing will not impair the correct performance of either the learned examples or the novel examples following the same rule). On the other hand, in the task with randomly established mapping, silencing the hippocampus would impair the animal's ability to recall previously memorized grating-to-reward-port mappings. Note that specific choices of such input/output pairs and their mapping rules would require considerations of the species-specific biology and ethology. For example, humans and non-human primates might be able to perform the angle-based task well, whereas mice might not have the visual acuity to differentiate nearby angles [1].

Example 2: An experiment performed by Sweegers et al. (Fig. S7c) is consistent with our task design recipe. Specifically, the inputs are human faces with certain features, the output is one of six locations on the screen to which the experimental subject needs to move a joystick to indicate the choice. The mapping was designed to be either highly predictable, mapping certain features to certain locations, or unpredictable, consisting of a random mapping between features and locations. The authors in this study measured systems consolidation using the strength of cortico-cortical coupling for remotely acquired associations compared to recent ones, indicated by functional connectivity (see Supplementary Material 11). fMRI-guided TMS could potentially be used in future studies to perturb the hippocampus for a more causal investigation of the relationship between task predictability and hippocampus dependence.

12.2 Experiments probing implementational mechanisms of Go-CLS

If experiments like the ones described above yield results that are consistent with the Go-CLS theory (i.e.,, show clear correlation between the level of systems consolidation and the task predictability), implementational mechanisms for how systems consolidation is regulated can then be studied. Experimental tools for investigating possible mechanisms will differ by species. In humans, non-invasive tools such as fMRI, electroencephalography (EEG) or magnetoencephalography (MEG) will be the predominate methods for monitoring brain activity. In animal model systems, finer-scale measurements can be performed in combination with the behavioral experiments described in the previous section. For example, electrical recordings using electrode arrays (e.g.,, tetrode arrays or neural pixel probes) or cellular-resolution calcium imaging can be used to record neural activity. The tools available in any of these species permit experiments to monitor changes in the way representations of the task change over time during learning in the hippocampus, neocortex, or both. Spontaneous neural activity during resting periods can also be recorded in any of these species to decode task-related representations [25] and monitor correlated changes in offline activities and the extent of consolidation. For model systems like rodents, optogenetic or chemogenetic tools can be used to causally identify different brain regions involved, perhaps even prior to recording/imaging, in order to focus those efforts on the most pertinent brain regions. Manipulations can be timed to interfere with learning during training sessions or during offline periods when consolidation is expected to occur. Similar experiments in humans could potentially be done through fMRI guided TMS or deep brain stimulation.

Once critical brain regions and appropriate manipulation times are identified, further refinement of experimental protocols can be used to dig deeper into mechanisms. For example, hippocampal sharp wave ripples (SWRs) associate with offline replay of prior experiences and have been shown to have a causal role in mediating systems consolidation [13]. It would therefore be natural to monitor these replay events (e.g.,, through electrophysiology in rodents or MEG or fMRI in humans [25]) after the subject learns either the rule-based or memory-based tasks. Replay content can then be decoded to see if rule-based experiences are replayed more than memory-based experiences. If so, this would be consistent with replay regulation as a mechanism for content-dependent regulation of systems consolidation.

If rule-based experiences are indeed preferentially replayed, it would then be possible to study the underlying mechanisms that selectively bias these replay events. One possibility is that brain regions outside of the hippocampus send inputs to the hippocampus to bias the content of replay, based on predictability. For example, in mice, the medial prefrontal cortex (mPFC) can influence hippocampal activity indirectly through the thalamic nucleus reuniens. mPFC has been implicated as a crucial brain region in rule-learning and cognitive control, and it is plausible that mPFC contains the predictability information needed to bias replay content. Two predictions follow from this hypothesis. First, neural activity in mPFC would be fundamentally different during offline replay of rule-based versus memory-based experiences. Second, silencing thalamic nucleus reuniens or mPFC (e.g.,, by chemogenetics) would disrupt the regulation of replay in a manner dependent on the predictability of experience. Other brain regions (e.g.,, ventral tegmental area) could be studied in a similar manner. Once such brain regions are identified in rodents, more detailed mechanistic experiments can be performed to map the cellular level connectivity, cell types, neurotransmitters, receptors, and other molecular processes involved in the regulation of systems consolidation.

On the other hand, it is also possible that hippocampal replay content and frequency are similar for experiences with differing predictability. In this case, an alternative hypothesis would be that regulation of systems consolidation occurs outside of the hippocampus. For example, replay events could be regulated so that neocortical regions respond differently depending on predictability of the content. This hypothesis is theoretically appealing because memories are composed of many relationships that differ in their predictability. To test this hypothesis, experimental techniques such as wide-field or mesoscopic calcium imaging (rodents) or fMRI (humans) can be used to monitor cortical-wide activity during periods of offline replay. Differential activation of the neocortical regions between the rule-based and the memory-based tasks would be informative about whether systems consolidation is regulated at the neocortical level.

References

[1] Mohammad Abdolrahmani, Dmitry R Lyamzin, Ryo Aoki, and Andrea Benucci. Attention decorrelates sensory and motor signals in the mouse visual cortex. *BioRxiv*, page 615229, 2021.

[2] M. Advani and S. Ganguli. An equivalence between high dimensional bayes optimal inference and m-estimation. *Advances in Neural Information Processing Systems*, 2016.

[3] M. Advani and S. Ganguli. Statistical mechanics of optimal convex inference in high dimensions. *Physical Review X*, 6(3):031034, 2016.

[4] M.S. Advani, A.M. Saxe, and H. Sompolinsky. High-dimensional dynamics of generalization error in neural networks. *Neural Networks*, 132:428–446, 2020.

[5] D.J. Amit, H. Gutfreund, and H. Sompolinsky. Storing infinite numbers of patterns in a spin-glass model of neural networks. *Physical Review Letters*, 55(14):1530–1533, 1985.

[6] D.J. Amit, H. Gutfreund, and H. Sompolinsky. Information storage in neural networks with low levels of activity. *Physical Review A*, 35(5):2293–2303, 1987.

[7] Alessio Attardo, James E Fitzgerald, and Mark J Schnitzer. Impermanence of dendritic spines in live adult ca1 hippocampus. *Nature*, 523(7562):592–596, 2015.

[8] M. Belkin, D. Hsu, S. Ma, and S. Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, August 2019.

[9] Joseph C Biedenkapp and Jerry W Rudy. Context preexposure prevents forgetting of a contextual fear memory: implication for regional changes in brain activation patterns associated with recent and remote memory tests. *Learning & Memory*, 14(3):200–203, 2007.

[10] A. Blum, J. Hopcroft, and R. Kannan. *Foundations of Data Science*. Cambridge University Press, 2020.

[11] Johan J Bolhuis, Caroline A Stewart, and Elma M Forrest. Retrograde amnesia and memory reactivation in rats with ibotenate lesions to the hippocampus or subiculum. *The Quarterly Journal of Experimental Psychology Section B*, 47(2b):129–150, 1994.

[12] J. Buhmann, R. Divko, and K. Schulten. Associative memory with high information content. *Physical Review. A, General Physics*, 39(5):2689–2692, March 1989.

[13] György Buzsáki. Hippocampal sharp wave-ripple: A cognitive biomarker for episodic memory and planning. *Hippocampus*, 25(10):1073–1188, 2015.

[14] Lycia D de Voogd, Yannick PJ Murray, Ramona M Barte, Anouk van der Heide, Guillén Fernández, Christian F Doeller, and Erno J Hermans. The role of hippocampal spatial representations in contextualization and generalization of fear. *Neuroimage*, 206:116308, 2020.

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[16] J.F. Fontanari. Generalization in a hopfield network. *J Phys France*, 51:2421–2430, 1990.

[17] Paul W Frankland, Cátia M Teixeira, and Szu-Han Wang. Grading the gradient: Evidence for time-dependent memory reorganization in experimental animals. *Debates in Neuroscience*, 1(2):67–78, 2007.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[19] J.J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, April 1982.

[20] Jeansok J Kim and Michael S Fanselow. Modality-specific retrograde amnesia of fear. *Science*, 256(5057):675–677, 1992.

[21] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[22] A. Krogh and J.A. Hertz. Generalization in a linear perceptron in the presence of noise. *Journal of Physics A: Mathematical and General*, 25:1135–1147, 1992.

[23] Y. LeCun, I. Kanter, and S.A. Solla. Eigenvalues of covariance matrices: Application to neural-network learning. *Physical Review Letters*, 66(18):2396, 1991.

[24] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[25] Yunzhe Liu, Matthew M Nour, Nicolas W Schuck, Timothy EJ Behrens, and Raymond J Dolan. Decoding cognition from spontaneous neural activity. *Nature Reviews Neuroscience*, 23(4):204–214, 2022.

[26] Eleanor A Maguire, Rory Nannery, and Hugo J Spiers. Navigation around london by a taxi driver with bilateral hippocampal lesions. *Brain*, 129(11):2894–2907, 2006.

[27] V.A. Marchenko and L.A. Pastur. Distribution of eigenvalues for some sets of random matrices. *Matematicheskii Sbornik*, 114:507–536, 1967.

[28] M Mézard, JP Nadal, and G Toulouse. Solvable models of working memories. *Journal de physique*, 47(9):1457–1462, 1986.

[29] Sören Mindermann, Jan Brauner, Muhammed Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. Prioritized training on points that are learnable, worth learning, and not yet learnt. *arXiv preprint arXiv:2206.07137*, 2022.

[30] Richard Morris. Developments of a water-maze procedure for studying spatial learning in the rat. *Journal of neuroscience methods*, 11(1):47–60, 1984.

[31] Amber C Ocampo, Larry R Squire, and Robert E Clark. Hippocampal area ca1 and remote memory in rats. *Learning & Memory*, 24(11):563–568, 2017.

[32] Blake A Richards and Paul W Frankland. The persistence and transience of memory. *Neuron*, 94(6):1071–1084, 2017.

[33] Blake A Richards, Frances Xia, Adam Santoro, Jana Husse, Melanie A Woodin, Sheena A Josselyn, and Paul W Frankland. Patterns across multiple memories are identified over time. *Nature neuroscience*, 17(7):981–986, 2014.

[34] Robert J Sutherland, Jamus O'Brien, and Hugo Lehmann. Absence of systems consolidation of fear memories after dorsal, ventral, or complete hippocampal damage. *Hippocampus*, 18(7):710–718, 2008.

[35] C.C.G. Sweegers, A. Takashima, G. Fernández, and L.M. Talamini. Neural mechanisms supporting the extraction of general knowledge across episodic memories. *NeuroImage*, 87:138–146, February 2014.

[36] Preston P Thakral, Kevin P Madore, Sarah E Kalinowski, and Daniel L Schacter. Modulation of hippocampal brain networks produces changes in episodic simulation and divergent thinking. *Proceedings of the National Academy of Sciences*, 117(23):12729–12740, 2020.

[37] M.V. Tsodyks and M.V. Feigelman. The enhanced storage capacity in neural networks with low-level activity. *Europhysics Letters*, 6(2), 1988.

[38] Brian J Wiltgen and Alcino J Silva. Memory for context becomes less specific with time. *Learning & memory*, 14(4):313–317, 2007.

[39] Brian J Wiltgen, Miou Zhou, Ying Cai, J Balaji, Mikael Guzman Karlsson, Sherveen N Parivash, Weidong Li, and Alcino J Silva. The hippocampus plays a selective role in the retrieval of detailed contextual memories. *Current biology*, 20(15):1336–1344, 2010.

[40] Gordon Winocur, Melanie J Sekeres, Malcolm A Binns, and Morris Moscovitch. Hippocampal lesions produce both nongraded and temporally graded retrograde amnesia in the same rat. *Hippocampus*, 23(5):330–341, 2013.

[41] Jianfeng Yao. A note on a marčenko–pastur type theorem for time series. *Statistics & probability letters*, 82(1):22–28, 2012.

[42] Andrew P Yonelinas, Charan Ranganath, Arne D Ekstrom, and Brian J Wiltgen. A contextual binding theory of episodic memory: systems consolidation reconsidered. *Nature Reviews Neuroscience*, 20(6):364–375, 2019.