Distance-Dependent Connectivity Shapes Continual Learning by Controlling Representational Drift

Chiu-Chang Cheng[1] **Ching-Lung Hsu**[2,3,4,5,*] **Ya-Ning Chang**[1,*] **Chao-Hung Wang**[1,6,*]

[1]Miin Wu School of Computing, National Cheng Kung University
[2]Institute of Biomedical Sciences (IBMS), Academia Sinica
[3]Neuroscience Program of Academia Sinica (NPAS)
[4]Department of Life Science, College of Life Science, National Taiwan University
[5]Graduate Institute of Life Sciences, National Defense Medical University
[6]Academy of Innovative Semiconductor and Sustainable Manufacturing, National Cheng Kung University

[*]Corresponding authors: hsuc@as.edu.tw, yaningchang@gs.ncku.edu.tw, chwang@gs.ncku.edu.tw

Abstract

Biological circuits learn continually while largely avoiding catastrophic forgetting, but the structural basis of this ability remains unclear. We test whether distance-dependent connectivity (DDC), a wiring motif of the mammalian cortex, provides part of this basis. We embed the neurons of a biologically plausible recurrent spiking neural network in a 3D Euclidean substrate, set recurrent connection probability by pairwise distance, and ablate that distance dependence *in silico*. DDC shapes a *synaptic resource geography*: it sets the spatial breadth of the candidate synaptic pool for learning, plasticity further compresses that pool, and the resulting substrate governs how subsequent inputs compete for the same synapses. On 10-way class-incremental MNIST, used as a controlled sequential-learning assay, accuracy is non-monotonic in connectivity range: an intermediate DDC range achieves the highest accuracy under a fixed, label-calibrated readout, outperforming both highly local circuits that collapse into a synaptic-resource bottleneck and random connectivity that produces diffuse synaptic contention. At the synaptic level, this pattern tracks consecutive-task contention more closely than neuronal assembly overlap. At the population level, these accuracy differences do not reflect lost information: old-class information remains decodable in every regime, and what differs is how far population responses drift away from the fixed readout. Connectivity range thus controls learning-induced representational drift and, with it, forgetting measured relative to a fixed readout. For continual learning, these results highlight how circuit wiring organizes the plastic synapses that memories share, not only which neurons form ensembles and engrams.[1]

1 Introduction

Continual learning is usually treated as a problem of protecting past knowledge through replay, regularization, or task-specific capacity allocation [Riemer et al., 2019, Kirkpatrick et al., 2017, Mallya and Lazebnik, 2018]. In neuroscience, memory stability is often framed through allocation and engrams [Josselyn and Frankland, 2018, Josselyn and Tonegawa, 2020, Rashid et al., 2016], pattern separation [Yassa and Stark, 2011, Leutgeb et al., 2007, Rolls, 2013], and complementary learning systems [McClelland et al., 1995, 2020]. These views explain how forgetting can be mitigated algorithmically and how memories may be assigned to neuronal populations. They leave open, however, whether recurrent connectivity itself determines how sequential experiences compete for plastic synapses. This motivates a substrate-level account of continual learning in recurrent plastic circuits.

[1]Code and data: https://github.com/eason891023/DDC-Continual-Learning-NeurIPS2026

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

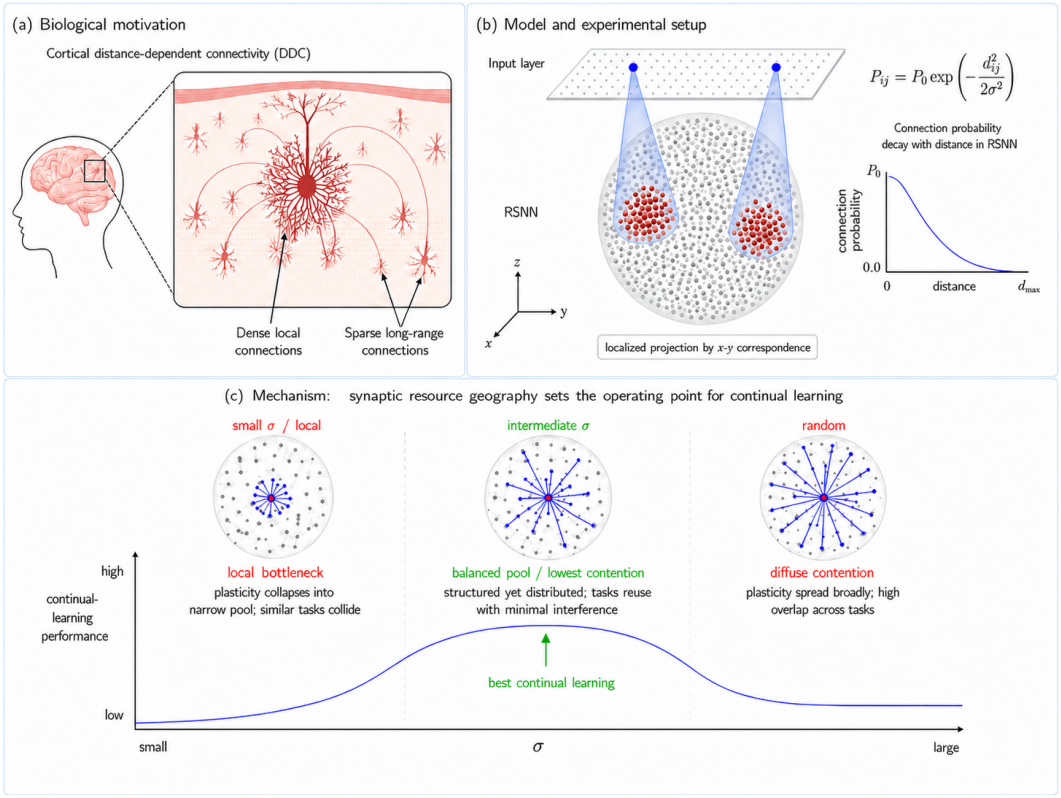


Figure 1: **Distance-dependent connectivity shapes synaptic resource geography.** **(a)** Cortical connection probability decreases with distance. **(b)** We vary recurrent connectivity range σ in a spatially embedded RSNN, with density-matched random connectivity as control. **(c)** Small σ creates a local synaptic bottleneck, random connectivity produces diffuse contention, and intermediate σ preserves spatial organization without collapsing the available synaptic pool.

We study distance-dependent connectivity (DDC) as a structural constraint on this synaptic substrate. DDC, the decay of connection probability with physical separation, is observed in cortical circuits [Holmgren et al., 2003, Campagnola et al., 2022], is embedded in non-random, functionally specific local wiring [Perin et al., 2011, Ko et al., 2011], and is often interpreted as wiring economy [Chklovskii et al., 2002, Bullmore and Sporns, 2012]. We hypothesize that DDC also shapes a *synaptic resource geography*: the spatial distribution of candidate synapses on which sequential experiences accumulate plasticity.

We test this hypothesis in a biologically grounded recurrent spiking neural network with unsupervised plasticity [Zenke et al., 2015, Tomé et al., 2024]. Neurons are embedded in a 3D Euclidean substrate, and we ablate only the recurrent connectivity rule by comparing distance-dependent Gaussian decay with density-matched random connectivity (Figure 1). The class-incremental learning (CIL) task [van de Ven and Tolias, 2019] serves as a controlled assay of memory stability; class labels enter only an experimenter-side readout, never network dynamics or plasticity.

Our analyses proceed from synapses to populations. We first show that DDC sets the spatial breadth of the recurrent synaptic pool and that plasticity compresses it into a smaller learned pool, defining where consecutive tasks contend (Section 4). On class-incremental MNIST, an intermediate range ($\sigma \approx 0.7$–1.0) performs best under a fixed readout (Section 5); this ordering follows consecutive-task contention rather than assembly overlap, and locality makes contention selective for similar inputs (Sections 6–7). Finally, old-class information survives in every regime: the regimes differ in how far learning-induced representational drift carries population responses away from the readout formed at acquisition, so forgetting under a fixed readout reflects drift rather than lost information (Section 8). Memory stability in recurrent plastic circuits thus depends not only on which neurons are recruited, but on how circuit architecture routes plasticity through shared or distinct synaptic resources.

2 Related Work

Continual learning and architectural structure. In machine learning, most continual learning methods reduce forgetting by regularizing important parameters [Kirkpatrick et al., 2017, Zenke et al., 2017], replaying past data [Riemer et al., 2019], or isolating task-specific capacity [Mallya and Lazebnik, 2018]; see De Lange et al. [2022] for a review. These approaches usually treat connectivity as fixed, although architecture itself can modulate forgetting [Mirzadeh et al., 2022], and representational structure and task semantics also affect forgetting [Ramasesh et al., 2021]. We build on this structural view and ask whether a conserved cortical wiring motif changes the synaptic substrate on which sequential tasks compete.

Memory allocation, assemblies, and interference. Neuroscience illustrates memory interference through allocation and separation. Engram studies show that memories are assigned to ensembles through excitability-dependent competition [Josselyn and Frankland, 2018, Josselyn and Tonegawa, 2020, Rashid et al., 2016], and temporally related memories can share substrates [Cai et al., 2016]. Plastic spiking network models show how such assemblies form and persist [Litwin-Kumar and Doiron, 2014, Zenke et al., 2015] or reorganize [Tomé et al., 2024], without varying the spatial structure of recurrent connectivity. Pattern separation emphasizes reduced representational overlap [Yassa and Stark, 2011, Leutgeb et al., 2007, Rolls, 2013]. These ideas motivate an assembly-separation hypothesis as a natural explanation to examine, but they do not by themselves specify how sequential plasticity competes over a shared synaptic substrate.

Representational drift. Neuronal population responses to the same stimulus or task can change over days to weeks [Ziv et al., 2013, Driscoll et al., 2017, Schoonover et al., 2021], even when task performance remains stable [Driscoll et al., 2017]. A readout fixed at one time point then loses accuracy, even though the population still discriminates stimuli at any given moment [Schoonover et al., 2021], and theory suggests that downstream circuits must track or tolerate such change [Rule et al., 2019, Driscoll et al., 2022]. In piriform cortex, drift has been proposed to reflect unstructured connectivity [Schoonover et al., 2021], raising the question of how connectivity structure sets the amount of drift.

Spatial organization and DDC. DDC is typically studied as wiring economy or representational organization (Section 1), and spatial constraints can shape brain-like representations in machine-learning models [Rathi et al., 2025, Margalit et al., 2024]. We bridge these lines of work with continual learning by varying only recurrent distance dependence while holding neuron models, plasticity, input statistics, and density unchanged.

3 Cortical-Like DDC Model

3.1 RSNN architecture

We adopt the RSNN architecture of Zenke et al. [2015] as our model platform. The network comprises 4096 excitatory and 1024 inhibitory leaky integrate-and-fire neurons with recurrent excitatory and inhibitory interactions. Each neuron is assigned a Euclidean coordinate sampled uniformly from a three-dimensional unit ball, providing a minimal spatial substrate for comparing connectivity priors under matched connection density. Full model settings are listed in Appendix A.

3.2 Distance-dependent connectivity

The key intervention is the recurrent connection probability between units i and j:

$$P_{ij} = P_0 \exp\left(-\frac{d_{ij}^2}{2\sigma^2}\right), \tag{1}$$

where d_{ij} is the Euclidean distance between their 3D coordinates, σ controls the connectivity range, and P_0 is scaled so that the mean connection probability matches a fixed target density. We use a Gaussian decay because cortical connection probability falls with intersomatic distance [Holmgren et al., 2003] and is well fit by a Gaussian profile [Campagnola et al., 2022], and because it gives a

smooth, isotropic, one-parameter control of connectivity range at fixed density. When σ is small, connectivity is highly local; as σ increases, the recurrent graph becomes increasingly diffuse and less distance-dependent (Figure 1c).

3.3 Plasticity rules

All synaptic updates are unsupervised and follow the base RSNN model of Zenke et al. [2015]. Excitatory synapses include short-term plasticity, triplet STDP [Pfister and Gerstner, 2006], transmitter-induced plasticity, and heterosynaptic plasticity [Chistiakova et al., 2014]; inhibitory synapses onto excitatory neurons follow inhibitory STDP [Vogels et al., 2011] with a global modulatory factor [Zenke et al., 2015]. These rules and their parameters are identical across conditions, so differences arise from recurrent topology alone. Appendix A summarizes the plasticity rules and full model settings.

3.4 Input mapping and readout

Input images are projected spatially onto the excitatory population: nearby image regions preferentially activate nearby excitatory units in the 3D substrate. This input scaffold is fixed in the main experiments, so changing σ affects only the recurrent RSNN topology.

Model evaluation is performed with a single label-calibrated, non-gradient selective-vote readout. During training, each image's excitatory spike counts are added to the column of a class-response matrix indexed by its label; labels are used only for this bookkeeping and never enter network dynamics or synaptic updates. Each class's column is fixed once that class has been learned. At test time, plasticity is disabled, no task identity is provided, and each image is assigned to the class whose mean-centered column has the largest dot product with its spike-count vector. The input mapping and readout equations are given in Appendix A.

4 How Distance-Dependent Connectivity Shapes Synaptic Resource Geography

Before turning to performance, we first address a mechanistic question: *what property of the learning substrate does the connectivity range σ actually control?* Our answer is *synaptic resource geography*: the effective spatial distribution of synaptic resources on which consecutive tasks can accumulate plasticity. Every recurrent graph, distance-dependent or random, defines a candidate synaptic substrate, and plasticity selects memory-dependent subsets from it; DDC does not create this geography, but shapes how those subsets are organized, how much they overlap, and how strongly they are rewritten. We directly examine this mechanism in a class-incremental MNIST setting. This geography emerges from three linked ingredients: the geometric width of the structural pool that DDC sets, the further compression induced by learning, and the spatial anatomy of the contention that later tasks experience within the learned pool.

Structural pool geometry: DDC first sets how broad the candidate synaptic substrate can be. Because DDC concentrates connections among nearby neurons, the set of synapses that *could* participate in learning—the structural pool—is geometrically narrower than under random connection. We quantify this with the *structural span*: the mean pre–post Euclidean distance of a random sample of connected synapses ($n = 4000$). Structural span decreases monotonically with σ, from 1.024 ± 0.002 under random connectivity to 0.788 ± 0.007 at $\sigma = 0.7$ and 0.548 ± 0.005 at $\sigma = 0.4$ (Figure 2a). This is a purely structural consequence of the connectivity rule, present before any learning.

Plasticity compresses the pool beyond what wiring defines. The structural pool, however, is only a *candidate* substrate. The quantity that matters for later tasks is the *learned* substrate after plasticity has concentrated onto a subset of those synapses. We identify task-important recurrent synapses as strong synapses whose pre- and postsynaptic neurons are both highly responsive to the same class (formal score in Appendix C). We then compare structural span to the *learned span*: the mean pre–post distance of the top-4000 task-important recurrent synapses identified after training. In every condition, learned span is smaller than structural span, with a 16–30% reduction across regimes (full numerical summary and statistics in Appendix C). Learning thus does not merely inherit

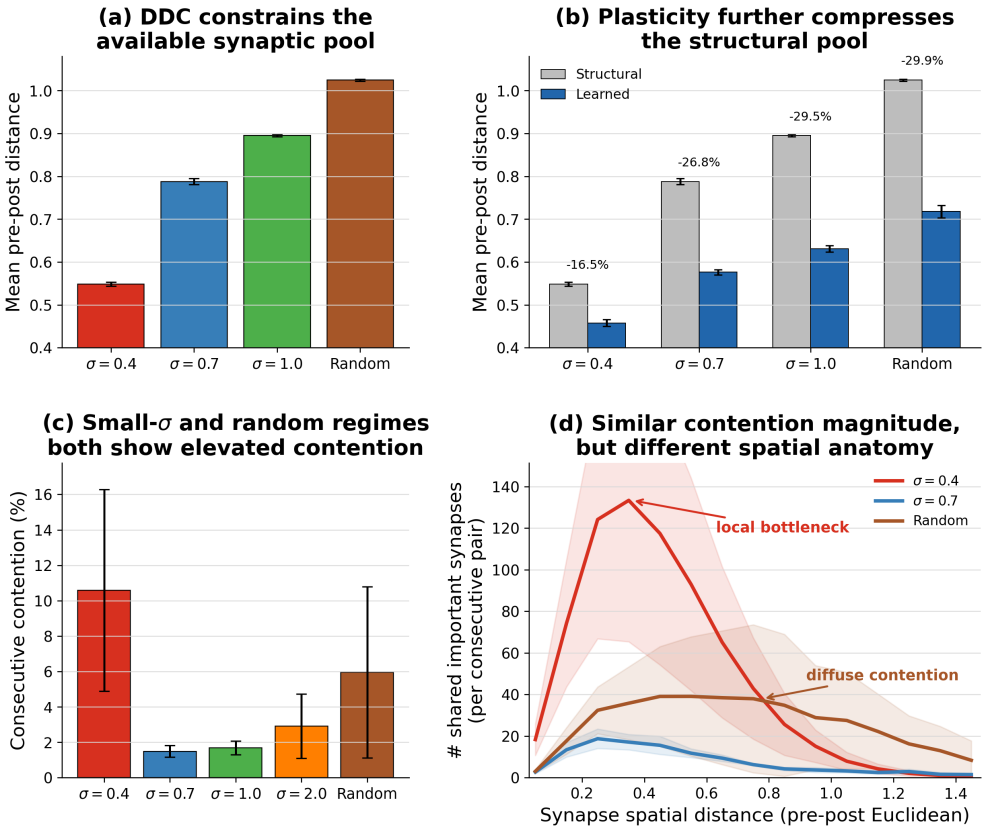


Figure 2: **DDC shapes synaptic resource geography.** (a) Pre–post Euclidean distance of connected synapses across connectivity regimes. (b) Pre–post Euclidean distance of structural versus learned task-important synapses after training. (c) Overall consecutive-task contention across regimes. (d) Distance-resolved contention distinguishes the local bottleneck at $\sigma = 0.4$ from the broader, less organized contention under random connectivity.

the structural pool but compresses it, so the usable substrate is produced jointly by topology and plasticity.

The learned pool determines the spatial anatomy of future contention. We define *consecutive-task contention* as the degree to which a new task's top important synapses overlap with those already claimed by the previous task (operationally, the Jaccard overlap of the two important-synapse sets; Appendix C). Regimes differ not only in pool width but in *which* synapses consecutive tasks share and *where* that sharing sits (Figure 2c–d). At $\sigma = 0.4$, the pool becomes so locally concentrated that consecutive tasks are forced to contend over the same narrow, short-range pocket of synapses—a regime we call a *local bottleneck*. Under random connectivity, contention is also elevated, but it is broader and less spatially organized: the substrate is wide, yet poorly guided. The intermediate regime occupies a different operating point—its pool remains spatially organized, but not so narrow that later tasks are forced into the same short-range synapses.

Mechanistic prediction for the benchmark. These three measurements motivate the term *synaptic resource geography*. *Resource* refers to the synaptic pool available for learning: first the structural pool that DDC sets, then the smaller learned pool of task-important synapses selected by plasticity. *Geography* refers to where this pool lies in the Euclidean substrate and where later tasks are forced to contend. Accuracy in Section 5 should therefore not improve monotonically as connectivity narrows; instead, performance should peak where the learned pool remains spatially organized without collapsing into a narrow contention bottleneck. Very small σ should fail by over-concentrating plasticity onto a tiny local substrate, while broad or random connectivity should fail by weakening the spatial bias that lets plasticity converge onto a coherent, task-specific subset of synapses.

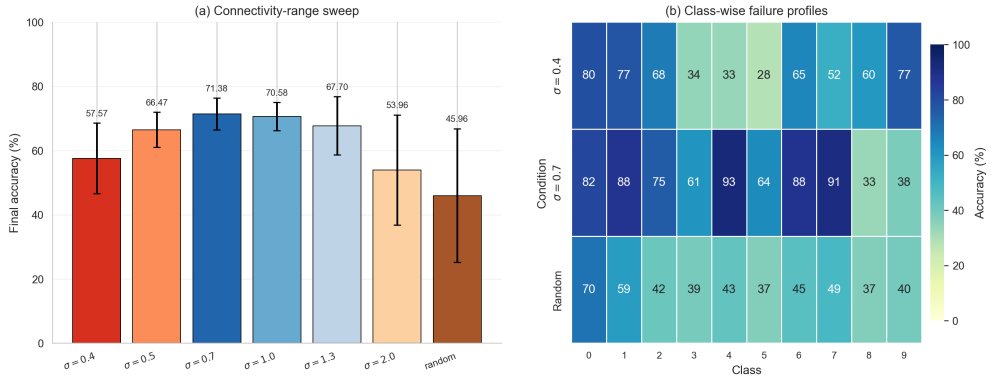


Figure 3: **Functional validation of the mechanistic prediction. (a)** Connectivity-range sweep on 10-way class-incremental MNIST (ten seeds per condition). **(b)** Per-class accuracy heatmap for $\sigma = 0.4$, $\sigma = 0.7$, and random connectivity. The low-σ and random regimes both underperform, but with different class-wise failure profiles.

5 Functional Validation of the Mechanistic Prediction

To validate the mechanistic prediction from Section 4, we evaluate continual learning performance on 10-way class-incremental MNIST [LeCun et al., 1998]. The network learns all ten digit classes ($0 \to 1 \to \cdots \to 9$) sequentially with a single pass through 100 training samples per class, and no task identity is provided at test time. We compare connectivity ranges $\sigma \in \{0.4, 0.5, 0.7, 1.0, 1.3, 2.0\}$ against the matched random-connectivity control, aggregating ten seeds per condition. We do not compare against machine-learning continual-learning algorithms such as regularization [Kirkpatrick et al., 2017, Zenke et al., 2017] or replay [Riemer et al., 2019], which add supervised objectives, importance estimates, or external memory; DDC is a structural variable rather than a learning algorithm, and whether it complements such algorithms is a separate question.

The benchmark follows the prediction closely (Figure 3a). DDC networks outperform the random-connectivity control on average ($45.96 \pm 20.82\%$), and the gain is non-monotonic across σ: accuracy peaks in an intermediate regime ($\sigma = 0.7$: $71.38 \pm 4.94\%$; $\sigma = 1.0$: $70.58 \pm 4.37\%$; Welch's t-test between them, $p = 0.706$), while both the most local regime ($\sigma = 0.4$; $57.57 \pm 11.01\%$) and the broadest ($\sigma = 2.0$; $53.96 \pm 17.12\%$) fall off substantially (full numerical summary in Table 3, Appendix B). We therefore refer to $\sigma \approx 0.7$–1.0 as the intermediate regime rather than identifying a single optimal σ; this peak is consistent with the synaptic-resource trade-off of Section 4. The same direction, though not statistically significant, is observed on Fashion-MNIST [Xiao et al., 2017] (Appendix G).

The per-class accuracy heatmaps (Figure 3b) show different failure profiles. Random connectivity depresses accuracy broadly, consistent with diffuse contention. The low-σ regime instead fails selectively: some classes are overwritten while others remain preserved, consistent with a local bottleneck in which only classes routed into the same narrow pool collide. The intermediate regime avoids both extremes.

A natural neuroscience interpretation is that DDC improves continual learning by separating neural assemblies. However, the benchmark peak alone does not determine whether performance is driven by stronger assembly separation or by reduced competition over the synaptic substrate, nor whether lower accuracy under the fixed readout reflects lost information. We therefore adjudicate between the two accounts directly (Sections 6–7) and then examine what the fixed readout measures (Section 8).

6 Synaptic Contention Predicts Continual Learning Beyond Assembly Separation

We investigate the assembly-separation account against the synaptic-resource account. If assembly separation drives the benchmark pattern, lower overlap between class-responsive assemblies should

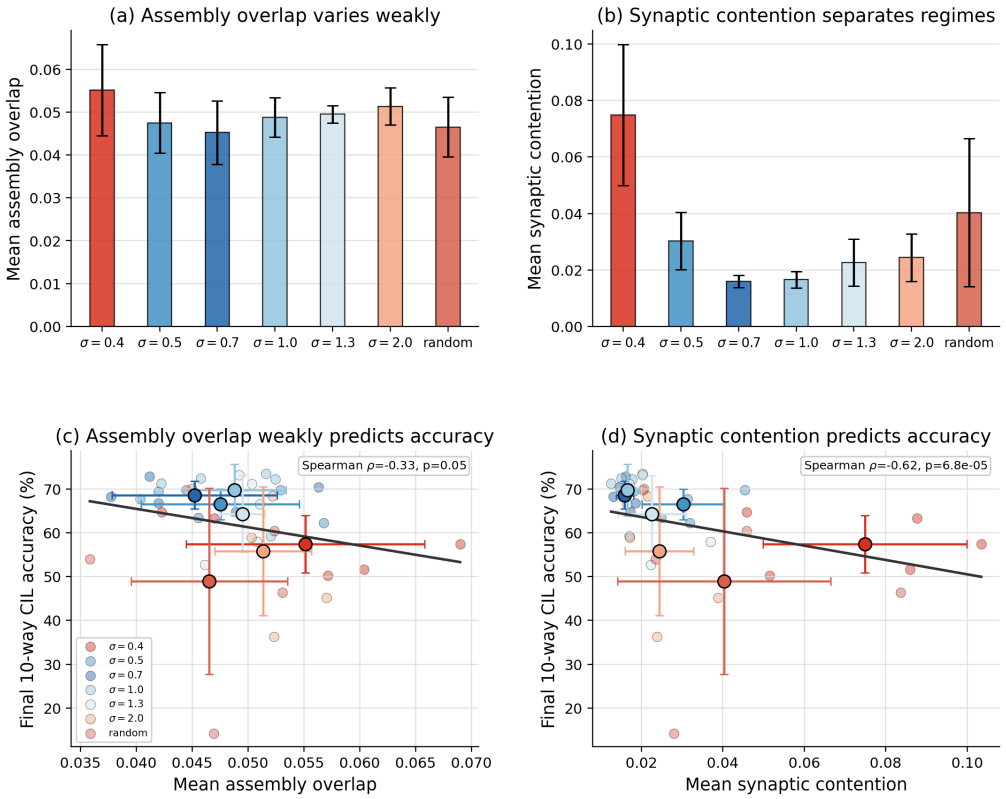


Figure 4: **Synaptic contention predicts continual learning beyond assembly separation. (a,b)** Condition-level: assembly overlap is nearly flat across σ, whereas consecutive-task synaptic contention separates the best intermediate regime from the local-bottleneck and random-connectivity ends. **(c,d)** Run-level: five runs per condition, assembly overlap is at most a weak predictor of final CIL accuracy (Spearman $\rho = -0.33$), while synaptic contention predicts more strongly (Spearman $\rho = -0.62$).

track higher final CIL accuracy. If synaptic-resource competition drives the pattern, performance should instead follow how strongly consecutive tasks claim and rewrite the same important synapses.

Operationally, we measure the inverse of separation: *assembly overlap*. For each class, we take the top-100 most responsive excitatory neurons as its class assembly, then compute the mean pairwise Jaccard overlap between class assemblies (Appendix C).

Assembly overlap fails to track the benchmark. Assembly overlap does not reproduce the non-monotonic accuracy curve (Figure 4a). The most local regime $\sigma = 0.4$ has higher overlap than the rest (0.055 ± 0.011), but the remaining DDC regimes are clustered in a narrow band: $\sigma = 0.7$ is lowest (0.045 ± 0.007), $\sigma = 0.5$ is only marginally higher (0.048 ± 0.007), and broader DDC regimes remain similar. This small ordering is incompatible with the much larger accuracy differences across σ. This weakness persists across individual runs ($n = 35$; five seeds per connectivity condition; Figure 4c; Pearson $r = -0.23$, $p = 0.18$; Spearman $\rho = -0.33$, $p = 0.05$). This does not make assemblies irrelevant, but shows that cell-level overlap is insufficient.

Synaptic contention tracks the benchmark. Consecutive-task contention—the Jaccard overlap of consecutive tasks' important-synapse sets, introduced in Section 4—separates the best-performing intermediate regimes from the local-bottleneck and random-connectivity ends. At the condition level (Figure 4b), $\sigma = 0.7$ shows the lowest contention (0.016 ± 0.002) and $\sigma = 1.0$ is comparable (0.017 ± 0.003), while contention is higher at $\sigma = 0.5$ (0.030 ± 0.010), $\sigma = 0.4$ (0.075 ± 0.025), and the random control (0.040 ± 0.026). Across the same runs ($n = 35$; Figure 4d), contention predicts accuracy more strongly than assembly overlap (Pearson $r = -0.31$, $p = 0.066$; Spearman

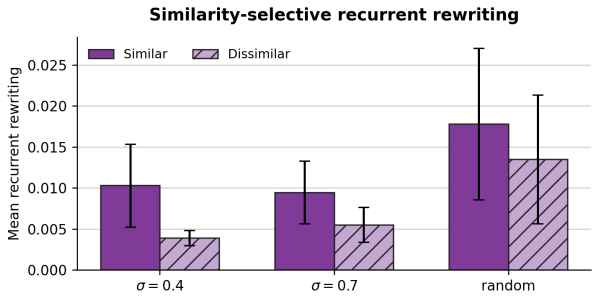


Figure 5: **Locality is a similarity selector.** Recurrent rewriting of the first class's important synapses, split by similar vs. dissimilar second classes. The local regime ($\sigma = 0.4$) shows the largest similar/dissimilar gap and the lowest dissimilar-pair rewriting of any regime; random connectivity is nearly class-blind.

$\rho = -0.62$, $p = 6.8 \times 10^{-5}$). Substrate-level contention, whether later tasks compete for the synapses supporting earlier representations, therefore tracks the benchmark more closely than cell-level overlap.

Directional interference tracks contention. From the machine-learning perspective, continual learning analyses often interpret forgetting through interference, in which later learning rewrites weights supporting earlier tasks [McCloskey and Cohen, 1989, French, 1999]. On the same runs we measured its directional component, the rewriting of an earlier task's important synapses by the next task (Appendix E.1). Across runs, contention closely tracks interference (Pearson $r = 0.91$, Spearman $\rho = 0.90$). Interference also reproduces the same condition-level ordering as Figure 4b and predicts accuracy with essentially the same strength (Spearman $\rho = -0.64$). Because both measures are defined on the same important-synapse sets, we treat this agreement as convergent weight-space evidence rather than as independent confirmation.

7 Locality Makes Synaptic Contention Similarity-Selective

The condition-level contention measure in Section 6 is class-blind: it captures how much consecutive tasks compete on average, but not which class pairs are routed into the same substrate. This distinction matters because $\sigma = 0.4$ and random connectivity fail differently in the per-class heatmap (Figure 3b). We therefore tested whether locality makes synaptic contention selective to input similarity.

To test this, we used a pairwise rewriting assay with predefined high-similarity MNIST pairs $\{(3,5),(5,8),(3,8)\}$ and low-similarity pairs $\{(0,1),(1,4),(0,7)\}$, defined by cosine similarity between class-mean images (Appendix C). The network first learned one class, then learned either a similar or dissimilar second class, and we measured rewriting of the first class's task-important recurrent synapses (Appendix D.1).

Locality strongly increased similarity-selective rewriting. The similar-to-dissimilar rewriting ratio was $2.65\times$ at $\sigma = 0.4$ ($p = 4.1 \times 10^{-4}$), $1.72\times$ at $\sigma = 0.7$, and $1.32\times$ under random connectivity. The absolute values reveal the mechanism: in the most local regime, dissimilar-pair rewriting was the lowest of all regimes, whereas similar-pair rewriting remained comparable to $\sigma = 0.7$. Thus, locality acts as a similarity selector: it protects dissimilar classes by keeping their synaptic pools apart, but routes similar classes into the same narrow short-range substrate. DDC range thus sets how strongly synaptic contention scales with stimulus similarity.

8 Connectivity Range Shapes Representational Drift, Not Information Loss

The benchmark in Section 5 reads out old classes through a selective-vote readout that is calibrated while each class is learned and then held fixed. Its accuracy can therefore fall for two reasons: later plasticity may erase class information from the recurrent population, or it may preserve that information while moving population responses away from the coordinates the readout expects, a form of representational drift [Rule et al., 2019, Driscoll et al., 2022], here induced by later learning. To separate these possibilities, we probed $\sigma = 0.4$, $\sigma = 0.7$, and random connectivity with plasticity disabled and network state reset before every probe image, so that probing itself left the network unchanged, tracking each class's mean response of all 4096 excitatory neurons to fixed probe images from acquisition to the end of the stream (Appendix F).

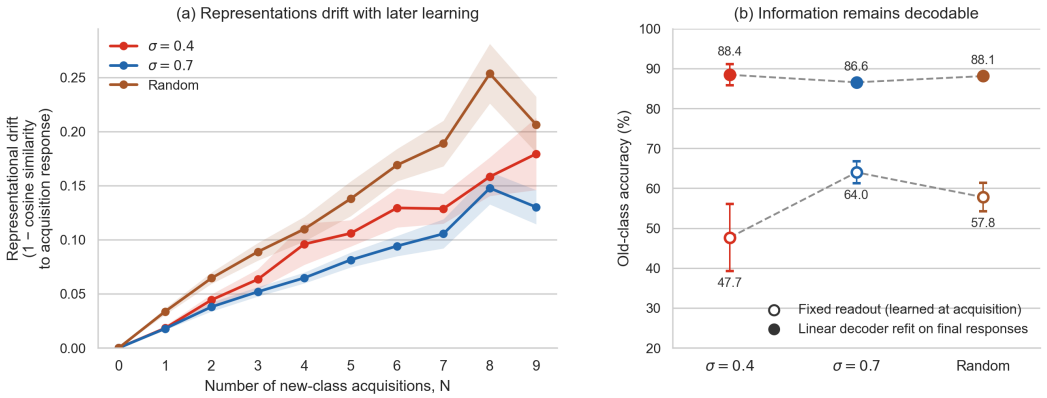


Figure 6: **Representations drift, but old-class information persists. (a)** Drift of old classes (1 minus the cosine similarity between a class's mean population response at acquisition and after N new-class acquisitions; mean $\pm$ SD, three paired seeds; points at $N \geq 8$ average only one or two classes). **(b)** Old-class accuracy after the full stream under the fixed selective-vote readout (open circles) and a linear decoder refit on final responses (filled circles); dashed lines join the three connectivity regimes for each readout (mean $\pm$ SD, five paired seeds).

Acquisition is matched; retention is not. Immediately after learning, old classes were recognized equally well in all three regimes (77.2%, 77.4%, and 76.3% for $\sigma = 0.4$, $\sigma = 0.7$, and random connectivity). By the end of the stream, old-class accuracy was highest at $\sigma = 0.7$ (63.1%, versus 52.4% and 59.3%), with the least forgetting (16.7 versus 25.6 and 22.0 percentage points; Table 4). The topology ordering therefore reflects how well old classes were retained, not how well they were initially encoded.

Representations drift with later learning. We quantified drift as one minus the cosine similarity between a class's mean response at acquisition and after N subsequent new-class acquisitions. Drift accumulated in every regime, but at different rates (Figure 6a): random connectivity drifted most from $N = 1$ onward, and $\sigma = 0.7$ drifted least from $N = 2$ onward. Averaged over old classes at the end of the stream, drift was 0.065 at $\sigma = 0.7$, versus 0.089 at $\sigma = 0.4$ and 0.116 under random connectivity. Following the first-learned class through the whole stream gives the same ordering in every paired seed (Appendix F).

Old-class information remains decodable. If later learning had erased old classes, no readout should recover them. Instead, in a separate five-seed readout audit (Appendix F), a linear decoder refit on final responses recovered old classes almost equally well in every regime (88.4%, 86.6%, and 88.1%; Figure 6b), whereas the fixed readout on the same images reached only 47.7%, 64.0%, and 57.8%. Class information thus survives sequential learning under all three topologies; what differs is whether it remains accessible to the readout formed when the class was learned. The refit gain, which measures how far a downstream readout must be realigned, was smallest at $\sigma = 0.7$ (22.6 percentage points, versus 40.8 at $\sigma = 0.4$ and 30.3 under random connectivity) in each of the five paired seeds, and recalibrating the original selective-vote readout on final responses gave the same ordering, so the contrast does not depend on the decoder family.

What the fixed readout measures. Under the fixed readout, accuracy loss is therefore protocol-relative functional forgetting: class information that remains in the population but has drifted away from the readout that originally decoded it. Fixing the readout exposes forgetting that recalibration would partly conceal, but it does not determine which topology drifts least or produce the non-monotonic ordering; that difference is the empirical result. In the regimes we tested, connectivity range changed not how much class information survived, but how far responses drifted, and therefore how much a downstream readout would need to adapt to keep reading earlier memories. Because drift and fixed-readout accuracy both compare acquisition-time and final responses, we treat them as convergent measurements of the same phenomenon, not as independent steps of a causal chain from synaptic contention to forgetting.

9 Discussion

DDC as a synaptic substrate for memory interference. Beyond wiring economy [Chklovskii et al., 2002, Bullmore and Sporns, 2012], our results suggest a functional role for DDC. By setting the geometric breadth of the recurrent synaptic pool, DDC controls where plasticity can accumulate and which synapses later inputs are forced to reuse. The key result is not that stronger locality is always better, but that recurrent connectivity range sets an operating point between local bottlenecks and diffuse contention. This agrees with evidence that architecture modulates forgetting in artificial networks [Mirzadeh et al., 2022] and extends it to a conserved cortical wiring motif.

From assemblies to synapses and drift. Pattern-separation accounts hold that lower representational overlap reduces interference [Yassa and Stark, 2011]. In our model, however, assembly overlap varied only weakly across regimes and predicted accuracy at most weakly, whereas consecutive-task synaptic contention tracked it more closely. This suggests that memory stability depends not only on which neurons are active, but also on whether later learning reuses the plastic synapses that support a memory, consistent with experience-dependent molecular specialization at input-specific engram synapses [Moharana et al., 2026]. At the population level, old-class information remained decodable in every regime while drifting least in the intermediate one. That random connectivity drifted most agrees with the proposal that drift in piriform cortex reflects unstructured connectivity [Schoonover et al., 2021] and extends theories in which downstream circuits must track or tolerate drift [Rule et al., 2019, Driscoll et al., 2022]: connectivity range may set how much tracking is required.

Locality sorts interference by similarity. Locality also changed which memories interfere. In the pairwise rewriting assay, the gap between similar and dissimilar classes grew with locality: the most local regime rewrote the synapses of dissimilar classes least of any regime, whereas similar classes were rewritten about as much as in the intermediate regime. Locality thus made interference depend on input similarity by sparing dissimilar memories rather than by separating similar ones. This is consistent with complementary-learning-systems accounts, in which cortical learning relies on overlapping representations while hippocampal pattern separation keeps similar memories apart [McClelland et al., 1995, Leutgeb et al., 2007]: cortex-like connectivity may protect dissimilar memories but need a separate mechanism for similar ones.

9.1 Limitations

Three questions remain open. First, does DDC organize memory competition for richer stimuli? The network stayed near chance on grayscale CIFAR-10 in a single-seed screen (18–21%; Appendix G), so beyond MNIST and Fashion-MNIST we claim neither broad visual generality nor invariance to translations or rotations. Second, how general is the operating point? Our conclusions are conditional on the plasticity rules of Zenke et al. [2015] (without the heterosynaptic and transmitter-induced terms, neither network learned; Appendix G), on a Gaussian kernel, and on a 3D unit ball without laminae, cell-type diversity, neuromodulation, or developmental wiring; density- and range-matched comparisons with other kernels and rules are needed. Third, does contention cause drift? We manipulated connectivity range but not pool geometry or plasticity directly, measured forgetting relative to a fixed, experimenter-side readout that a plastic downstream population could partly compensate for, and used three to five paired seeds for the readout diagnostics; contention, drift, and accuracy are therefore convergent observations rather than a demonstrated causal chain.

10 Conclusion

This study suggests that cortical distance-dependent connectivity may support continual learning by organizing the synaptic substrate on which memories compete. An intermediate range ($\sigma \approx 0.7$–1.0 in our sweep) kept connectivity spatially structured without collapsing plasticity into a local bottleneck; in this regime, old-class information remained decodable while drifting least from the readout formed at acquisition, so DDC may limit learning-induced representational drift through a structured-yet-not-collapsed synaptic resource geography. Measuring contention and drift in circuits that differ in connectivity range, and combining connectivity priors with replay or regularization, are natural next steps.

Acknowledgments and Disclosures

Funding. The research was supported by the National Science and Technology Council of Taiwan (Grant No. NSTC 113-2221-E-006-138, NSTC 113-2410-H-006-118, NSTC 114-2221-E-006-205, NSTC 114-2410-H-006-122-MY3, NSTC 115-2221-E-006-237), in part by Higher Education Sprout Project, Ministry of Education to the Headquarters of University Advancement at National Cheng Kung University (NCKU), AI Sustainable Development Goals Future Star Project from Miin Wu School of Computing, NCKU, and Taiwan Semiconductor Research Institute-Joint Development Project (JDP113-Y1-007 and JDP114-Y1-009). The research also received funding from the Institute of Biomedical Sciences (IBMS), Academia Sinica and the Ministry of Science and Technology (now National Science and Technology Council, NSTC) 2030 Cross-Generation Young Investigator Award (MOST 111-2628-B-001-017-MY3).

Competing interests. The authors declare no competing interests.

References

Ed Bullmore and Olaf Sporns. The economy of brain network organization. *Nature Reviews Neuroscience*, 13(5):336–349, 2012. doi: 10.1038/nrn3214.

Denise J. Cai, Daniel Aharoni, Tristan Shuman, Justin Shobe, Jeremy Biane, Weilin Song, Brandon Wei, Michael Veshkini, Mimi La-Vu, Jerry Lou, Sergio E. Flores, Isaac Kim, Yoshitake Sano, Miou Zhou, Karsten Baumgaertel, Ayal Lavi, Masakazu Kamata, Mark Tuszynski, Mark Mayford, Peyman Golshani, and Alcino J. Silva. A shared neural ensemble links distinct contextual memories encoded close in time. *Nature*, 534(7605):115–118, 2016. doi: 10.1038/nature17955.

Luke Campagnola, Stephanie C. Seeman, Thomas Chartrand, Lisa Kim, Alex Hoggarth, Clare Gamlin, Shinya Ito, Jessica Trinh, Pasha Davoudian, Cristina Radaelli, et al. Local connectivity and synaptic dynamics in mouse and human neocortex. *Science*, 375(6585):eabj5861, 2022. doi: 10.1126/science.abj5861.

Marina Chistiakova, Nicholas M. Bannon, Maxim Bazhenov, and Maxim Volgushev. Heterosynaptic plasticity: Multiple mechanisms and multiple roles. *The Neuroscientist*, 20(5):483–498, 2014. doi: 10.1177/1073858414529829.

Dmitri B. Chklovskii, Thomas Schikorski, and Charles F. Stevens. Wiring optimization in cortical circuits. *Neuron*, 34(3):341–347, 2002. doi: 10.1016/S0896-6273(02)00679-7.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3366–3385, 2022. doi: 10.1109/TPAMI.2021.3057446.

Laura N. Driscoll, Noah L. Pettit, Matthias Minderer, Selmaan N. Chettih, and Christopher D. Harvey. Dynamic reorganization of neuronal activity patterns in parietal cortex. *Cell*, 170(5):986–999.e16, 2017. doi: 10.1016/j.cell.2017.07.021.

Laura N. Driscoll, Lea Duncker, and Christopher D. Harvey. Representational drift: Emerging theories for continual learning and experimental future directions. *Current Opinion in Neurobiology*, 76: 102609, 2022. doi: 10.1016/j.conb.2022.102609.

Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3 (4):128–135, 1999. doi: 10.1016/S1364-6613(99)01294-2.

Carl Holmgren, Tibor Harkany, Björn Svennenfors, and Yuri Zilberter. Pyramidal cell communication within local networks in layer 2/3 of rat neocortex. *The Journal of Physiology*, 551(1):139–153, 2003. doi: 10.1113/jphysiol.2003.044784.

Sheena A. Josselyn and Paul W. Frankland. Memory allocation: Mechanisms and function. *Annual Review of Neuroscience*, 41:389–413, 2018. doi: 10.1146/annurev-neuro-080317-061956.

Sheena A. Josselyn and Susumu Tonegawa. Memory engrams: Recalling the past and imagining the future. *Science*, 367(6473):eaaw4325, 2020. doi: 10.1126/science.aaw4325.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Ho Ko, Sonja B. Hofer, Bruno Pichler, Katherine A. Buchanan, P. Jesper Sjöström, and Thomas D. Mrsic-Flogel. Functional specificity of local synaptic connections in neocortical networks. *Nature*, 473(7345):87–91, 2011. doi: 10.1038/nature09880.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

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. doi: 10.1109/5.726791.

Jill K. Leutgeb, Stefan Leutgeb, May-Britt Moser, and Edvard I. Moser. Pattern separation in the dentate gyrus and CA3 of the hippocampus. *Science*, 315(5814):961–966, 2007. doi: 10.1126/science.1135801.

Ashok Litwin-Kumar and Brent Doiron. Formation and maintenance of neuronal assemblies through synaptic plasticity. *Nature Communications*, 5:5319, 2014. doi: 10.1038/ncomms6319.

Arun Mallya and Svetlana Lazebnik. PackNet: Adding multiple tasks to a single network by iterative pruning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7765–7773, 2018. doi: 10.1109/CVPR.2018.00810.

Eshed Margalit, Hyodong Lee, Dawn Finzi, James J. DiCarlo, Kalanit Grill-Spector, and Daniel L. K. Yamins. A unifying framework for functional organization in early and higher ventral visual cortex. *Neuron*, 112(14):2435–2451.e7, 2024. doi: 10.1016/j.neuron.2024.04.018.

James L. McClelland, Bruce L. McNaughton, and Randall C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3):419–457, 1995. doi: 10.1037/0033-295X.102.3.419.

James L. McClelland, Bruce L. McNaughton, and Andrew K. Lampinen. Integration of new information in memory: New insights from a complementary learning systems perspective. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 375(1799):20190637, 2020. doi: 10.1098/rstb.2019.0637.

Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, pages 109–165. Academic Press, 1989. doi: 10.1016/S0079-7421(08)60536-8.

Seyed Iman Mirzadeh, Arslan Chaudhry, Dong Yin, Timothy Nguyen, Razvan Pascanu, Dilan Görür, and Mehrdad Farajtabar. Architecture matters in continual learning. *arXiv preprint arXiv:2202.00275*, 2022.

Biswajit Moharana, Panthea Nemat, Renee M. Pullen, Anna Gradl, Marijn Schipper, Jeanne E. Savage, Remco V. Klaassen, Rolinka J. van der Loo, Cora H. Chadick, Frank Koopmans, Yvonne Gouwenberg, Juan J. Garcia Vallejo, Michel C. van den Oever, August B. Smit, and Priyanka Rao-Ruiz. GRASPing experience-dependent protein expression signatures enriched for hippocampal engram cell synapses. *Science Advances*, 12(20):eadv3557, 2026. doi: 10.1126/sciadv.adv3557.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011.

Rodrigo Perin, Thomas K. Berger, and Henry Markram. A synaptic organizing principle for cortical neuronal groups. *Proceedings of the National Academy of Sciences*, 108(13):5419–5424, 2011. doi: 10.1073/pnas.1016051108.

Jean-Pascal Pfister and Wulfram Gerstner. Triplets of spikes in a model of spike timing-dependent plasticity. *The Journal of Neuroscience*, 26(38):9673–9682, 2006. doi: 10.1523/JNEUROSCI.1425-06.2006.

Vinay V. Ramasesh, Ethan Dyer, and Maithra Raghu. Anatomy of catastrophic forgetting: Hidden representations and task semantics. In *International Conference on Learning Representations (ICLR)*, 2021.

Asim J. Rashid, Chen Yan, Valentina Mercaldo, Hwa-Lin (Liz) Hsiang, Sungmo Park, Christina J. Cole, Antonietta De Cristofaro, Julia Yu, Charu Ramakrishnan, Soo Yeun Lee, Karl Deisseroth, Paul W. Frankland, and Sheena A. Josselyn. Competition between engrams influences fear memory formation and recall. *Science*, 353(6297):383–387, 2016. doi: 10.1126/science.aaf0594.

Neil Rathi, Johannes Mehrer, Badr AlKhamissi, Taha Binhuraib, Nicholas M. Blauch, and Martin Schrimpf. TopoLM: brain-like spatio-functional organization in a topographic language model. In *International Conference on Learning Representations (ICLR)*, 2025.

Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesauro. Learning to learn without forgetting by maximizing transfer and minimizing interference. In *International Conference on Learning Representations (ICLR)*, 2019.

Edmund T. Rolls. The mechanisms for pattern completion and pattern separation in the hippocampus. *Frontiers in Systems Neuroscience*, 7:74, 2013. doi: 10.3389/fnsys.2013.00074.

Michael E. Rule, Timothy O'Leary, and Christopher D. Harvey. Causes and consequences of representational drift. *Current Opinion in Neurobiology*, 58:141–147, 2019. doi: 10.1016/j.conb.2019.08.005.

Carl E. Schoonover, Sarah N. Ohashi, Richard Axel, and Andrew J. P. Fink. Representational drift in primary olfactory cortex. *Nature*, 594(7864):541–546, 2021. doi: 10.1038/s41586-021-03628-7.

Douglas Feitosa Tomé, Ying Zhang, Tomomi Aida, Olivia Mosto, Yifeng Lu, Mandy Chen, Sadra Sadeh, Dheeraj S. Roy, and Claudia Clopath. Dynamic and selective engrams emerge with memory consolidation. *Nature Neuroscience*, 27(3):561–572, 2024. doi: 10.1038/s41593-023-01551-w.

Gido M. van de Ven and Andreas S. Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019.

Tim P. Vogels, Henning Sprekeler, Friedemann Zenke, Claudia Clopath, and Wulfram Gerstner. Inhibitory plasticity balances excitation and inhibition in sensory pathways and memory networks. *Science*, 334(6062):1569–1573, 2011. doi: 10.1126/science.1211095.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Michael A. Yassa and Craig E. L. Stark. Pattern separation in the hippocampus. *Trends in Neurosciences*, 34(10):515–525, 2011. doi: 10.1016/j.tins.2011.06.006.

Friedemann Zenke, Everton J. Agnes, and Wulfram Gerstner. Diverse synaptic plasticity mechanisms orchestrated to form and retrieve memories in spiking neural networks. *Nature Communications*, 6:6922, 2015. doi: 10.1038/ncomms7922.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pages 3987–3995, 2017.

Yaniv Ziv, Laurie D. Burns, Eric D. Cocker, Elizabeth O. Hamel, Kunal K. Ghosh, Lacey J. Kitch, Abbas El Gamal, and Mark J. Schnitzer. Long-term dynamics of CA1 hippocampal place codes. *Nature Neuroscience*, 16(3):264–266, 2013. doi: 10.1038/nn.3329.

A Model Details

The RSNN follows the model family of Zenke et al. [2015] and the simulation setting reused by Tomé et al. [2024]. The paper-specific intervention is the recurrent connectivity mask: we replace the fixed-probability recurrent graph with a density-matched distance-dependent graph, while keeping the neuron dynamics, conductance dynamics, plasticity rules, and evaluation protocol fixed across conditions.

A.1 Neuron Model

All RSNN units are conductance-based leaky integrate-and-fire neurons. For neuron i,

$$\tau_m \frac{dU_i}{dt} = U_{\text{rest}} - U_i + g_i^{\text{exc}}(t)(U_{\text{exc}} - U_i) + g_i^{\text{inh}}(t)(U_{\text{inh}} - U_i), \tag{2}$$

where U_i is the membrane potential, g_i^{exc} and g_i^{inh} are total excitatory and inhibitory conductances, and $(U_{\text{rest}}, U_{\text{exc}}, U_{\text{inh}}) = (-70, 0, -80)$ mV. A spike is emitted when

$$S_i(t) = \mathbb{I}[U_i(t) \geq \vartheta_i(t)]. \tag{3}$$

After a spike, U_i is reset to U_{rest}, the threshold is increased by $\Delta\vartheta = 2$ mV, and the unit enters an absolute refractory period. For non-spiking units, the threshold relaxes as

$$\tau_\vartheta \frac{d\vartheta_i}{dt} = \vartheta_{\text{rest}} - \vartheta_i, \qquad \vartheta_{\text{rest}} = -50 \text{ mV}, \quad \tau_\vartheta = 5 \text{ ms}. \tag{4}$$

The simulation time step is $\Delta t = 0.1$ ms.

A.2 Synapse Model

Excitatory synapses use short-term plasticity and AMPA/NMDA-like conductances. For a synapse from presynaptic neuron j to postsynaptic neuron i, the short-term variables are

$$\frac{du_j}{dt} = \frac{U - u_j}{\tau_f} + U(1 - u_j)S_j(t), \tag{5}$$

$$\frac{dx_j}{dt} = \frac{1 - x_j}{\tau_d} - u_j x_j S_j(t), \tag{6}$$

and the effective presynaptic event is $u_j x_j S_j(t)$. For excitatory projections,

$$\frac{dg_i^{\text{AMPA}}}{dt} = -\frac{g_i^{\text{AMPA}}}{\tau_{\text{AMPA}}} + \sum_j m_{ji} w_{ji} u_j x_j S_j(t), \tag{7}$$

$$\frac{dg_i^{\text{NMDA}}}{dt} = -\frac{g_i^{\text{NMDA}}}{\tau_{\text{NMDA}}} + g_i^{\text{AMPA}}, \tag{8}$$

$$g_i^{\text{exc}} = \alpha g_i^{\text{AMPA}} + (1 - \alpha) g_i^{\text{NMDA}}, \tag{9}$$

where $m_{ji} \in \{0, 1\}$ is the connectivity mask. Inhibitory projections use GABA-like conductance dynamics,

$$\frac{dg_i^{\text{GABA}}}{dt} = -\frac{g_i^{\text{GABA}}}{\tau_{\text{GABA}}} + \sum_j m_{ji} w_{ji} S_j(t), \tag{10}$$

and spike-triggered adaptation is integrated as

$$\frac{dg_i^a}{dt} = -\frac{g_i^a}{\tau_a} + \Delta_a S_i(t). \tag{11}$$

We use $\tau_{\text{AMPA}} = 5$ ms, $\tau_{\text{NMDA}} = 100$ ms, $\tau_{\text{GABA}} = 10$ ms, $\tau_a = 100$ ms, $\Delta_a = 0.1$, $U = 0.2$, $\tau_f = 600$ ms, $\tau_d = 150$ ms for E→E and input→E synapses, and $\tau_d = 200$ ms for E→I synapses.

A.3 Synaptic Plasticity Model

Plasticity is unsupervised. Excitatory E→E and input→E synapses follow the plasticity rules of Zenke et al. [2015], combining triplet STDP, transmitter-induced plasticity, heterosynaptic plasticity, and slow synaptic consolidation. The traces are

$$\frac{dz_j^+}{dt} = -\frac{z_j^+}{\tau_+} + S_j(t), \qquad \frac{dz_i^-}{dt} = -\frac{z_i^-}{\tau_-} + S_i(t), \tag{12}$$

$$\frac{dz_i^{\mathrm{slow}}}{dt} = -\frac{z_i^{\mathrm{slow}}}{\tau_{\mathrm{slow}}} + S_i(t), \qquad \frac{dz_i^{\mathrm{ht}}}{dt} = -\frac{z_i^{\mathrm{ht}}}{\tau_{\mathrm{ht}}} + S_i(t). \tag{13}$$

For an excitatory synapse $j \to i$, the online weight update is

$$\frac{dw_{ji}}{dt} = \eta\big[A z_j^+ z_i^{\mathrm{slow}} S_i(t) - B_i z_i^- S_j(t)\big] + \delta S_j(t) - \beta(w_{ji} - \tilde{w}_{ji})(z_i^-)^3 S_i(t), \tag{14}$$

with weights clipped to $[w_{\min}, w_{\max}] = [0, 5]$. The homeostatic depression factor is

$$B_i = \begin{cases} AC_i, & C_i \le 1, \\ A, & C_i > 1, \end{cases} \qquad \frac{dC_i}{dt} = -\frac{C_i}{\tau_{\mathrm{hom}}} + (z_i^{\mathrm{ht}})^2. \tag{15}$$

The slow reference weight follows

$$\tau_{\mathrm{cons}} \frac{d\tilde{w}_{ji}}{dt} = -\tilde{w}_{ji} + w_{ji} - P\tilde{w}_{ji}\left(\frac{w_p}{2} - \tilde{w}_{ji}\right)(w_p - \tilde{w}_{ji}). \tag{16}$$

We use $\eta = 10^{-3}$, $A = 1$, $\beta = 50\eta$, $\delta = 0.02\eta$, $\tau_+ = \tau_- = 20$ ms, $\tau_{\mathrm{slow}} = 100$ ms, $\tau_{\mathrm{ht}} = 100$ ms, $\tau_{\mathrm{hom}} = 10$ min, $\tau_{\mathrm{cons}} = 20$ min, $P = 20$, $w_p = 0.5$, and $\tilde{w}_{ji}(0) = 0$.

Inhibitory I→E synapses follow globally modulated inhibitory STDP. Let z_j and z_i be pre- and postsynaptic traces with $\tau_{\mathrm{iSTDP}} = 20$ ms, and let

$$\frac{dH}{dt} = -\frac{H}{\tau_H} + \sum_{i \in E} S_i(t), \qquad G(t) = H(t) - \gamma. \tag{17}$$

The inhibitory update is

$$\frac{dw_{ji}}{dt} = \eta_{\mathrm{inh}} G(t)\left[(z_j + 1)S_i(t) + z_i S_j(t)\right], \tag{18}$$

with $\eta_{\mathrm{inh}} = \eta/50$, $\gamma = 4$ Hz, $\tau_H = 10$ s, and weights clipped to $[0, 5]$. E→I and I→I weights are not modified by long-term plasticity; they retain short-term or conductance dynamics as described above.

A.4 Network Model

The RSNN contains $N_E = 4096$ excitatory and $N_I = 1024$ inhibitory neurons. Each RSNN neuron is assigned a coordinate $\mathbf{x}_i \in \mathbb{R}^3$ sampled uniformly from the volume of a unit ball. The visual input layer contains $28 \times 28 = 784$ image-driven input neurons arranged on a two-dimensional grid at fixed $z = 2$ coordinates.

The manipulated recurrent pathways are E→E, E→I, I→E, and I→I. For distance-dependent connectivity,

$$\tilde{P}_{ji} = \exp\left(-\frac{\|\mathbf{x}_j - \mathbf{x}_i\|_2^2}{2\sigma^2}\right), \qquad P_{ji} = \min\left(1, s\tilde{P}_{ji}\right), \tag{19}$$

where s is chosen so that the mean recurrent connection probability matches the target density 0.05. The random control uses $P_{ji} = 0.05$ for all recurrent pairs. A binary mask is then sampled as $m_{ji} \sim \mathrm{Bernoulli}(P_{ji})$ and kept fixed during training.

The main input projection is spatially localized. For each input pixel p, we choose a nearby excitatory center in the 3D substrate and connect the pixel to the 250 nearest excitatory neurons around that center, with a directional filter that preserves the image-to-substrate spatial scaffold.

Table 1: **Core parameters used in all main experiments.**

Component	Value	Notes
Excitatory / inhibitory neurons	$4096/1024$	RSNN populations
Input neurons	784	MNIST or Fashion-MNIST pixels
Simulation step	0.1 ms	fixed across conditions
Recurrent target density	0.05	matched DDC and random masks
Connectivity ranges	$\sigma \in \{0.4, 0.5, 0.7, 1.0, 1.3, 2.0\}$	main sweep
Input receptive field	250 E neurons / pixel	spatial input mode
Readout	label-calibrated selective vote	fixed evaluation rule

A.5 Training and Evaluation Protocol

The main benchmark is 10-way class-incremental MNIST. Classes are presented sequentially in the order $0 \to 1 \to \cdots \to 9$, with 100 training samples per class and no replay. Before class-incremental training, each run undergoes a burn-in period using one randomly selected training image: excitatory long-term plasticity is disabled, while inhibitory plasticity remains active to stabilize activity. During training on each sample, the image is presented for $T_{\mathrm{on}} = 1000$ simulation steps, followed by a zero-input interval of $T_{\mathrm{off}} = 2000$ steps. Excitatory spike counts from step $T_{\mathrm{skip}} = 100$ of the presentation through the end of the zero-input interval are added to the column of the class-response matrix $R \in \mathbb{R}^{N_E \times C}$ indexed by the image's label. Labels are used only for this bookkeeping and never enter network dynamics or synaptic updates; each class's column is fixed once that class has been learned.

At evaluation time, all plasticity is disabled and no task identity is provided. No supervised classifier is trained on top of the RSNN for the benchmark; the refit linear decoder of Section 8 is a post hoc diagnostic (Appendix F). For a test sample n, we collect an excitatory spike-count vector $\mathbf{s}_n$ over the same window used during training. In the main benchmark, 100 held-out test images per class are presented in class order, and network state is carried over from one test image to the next, each image being followed by the zero-input interval. The selective-vote readout subtracts each neuron's across-class mean response,

$$\tilde{R}_{ic} = R_{ic} - \frac{1}{C}\sum_{c'=1}^{C} R_{ic'}, \qquad \hat{y}_n = \arg\max_c \sum_i s_{n,i}\tilde{R}_{ic}. \tag{20}$$

This fixed readout is used for all main benchmark curves, per-class heatmaps, and cross-dataset validation. All reported accuracies are final accuracies after all ten classes have been learned unless stated otherwise. Table 2 summarizes the protocol used by each analysis.

Table 2: **Protocols by analysis.** All analyses use the same network, plasticity rules, timing ($T_{\mathrm{on}} = 1000$, $T_{\mathrm{off}} = 2000$, $T_{\mathrm{skip}} = 100$), class order $0 \to \cdots \to 9$, and recurrent density.

Analysis	Train/class	Seeds	Test/class	Test state	Readout
MNIST benchmark (Fig. 3, Table 3)	100	10 per condition	100	carried over	selective vote
Contention, interference (Fig. 4, 8)	100	5 per condition	100	carried over	selective vote
Fashion-MNIST (Fig. 10)	as MNIST	42–46	as MNIST	carried over	selective vote
Pairwise rewriting (Fig. 5)	10 per class	3 per pair	–	–	–
Fixed probe (Fig. 6a, Table 4)	50	43, 44, 46	20	reset	selective vote, frozen at acquisition
Readout ladder (Fig. 6b)	50	42–46	30 calib. + 20 eval.	reset	fixed, recalibrated, refit
CIFAR-10 and plasticity screens	10	80	10	reset	selective vote

B Benchmark Numerical Details

C Analysis Metric Definitions

This appendix defines the analysis quantities used throughout the paper. Let C be the number of classes, E the excitatory population, $W^{(t)}$ a weight matrix after task t, and M the corresponding binary connectivity mask.

Table 3: **Full numerical results for 10-way class-incremental MNIST.** Mean ± sample SD over ten seeds per condition. $\sigma = 0.7$ and $\sigma = 1.0$ do not differ significantly (Welch's t-test, $p = 0.706$).

Condition	Final 10-way CIL accuracy (%)
Distance, $\sigma = 0.4$	57.57 ± 11.01
Distance, $\sigma = 0.5$	66.47 ± 5.47
Distance, $\sigma = 0.7$	71.38 ± 4.94
Distance, $\sigma = 1.0$	70.58 ± 4.37
Distance, $\sigma = 1.3$	67.70 ± 9.06
Distance, $\sigma = 2.0$	53.96 ± 17.12
Random connectivity	45.96 ± 20.82

Class-response matrix. During training and probing, we accumulate excitatory spike counts into a class-response matrix $R \in \mathbb{R}_{\geq 0}^{|E| \times C}$:

$$R_{ic} = \sum_{n: y_n = c} \sum_{\tau \in \mathcal{T}_{\text{count}}} S_i(\tau; n), \tag{21}$$

where $S_i(\tau; n)$ is the spike of excitatory neuron i at time τ for sample n, and $\mathcal{T}_{\text{count}}$ excludes the initial transient period. For importance scoring, we use the nonnegative normalized affinity

$$a_i(c) = \frac{R_{ic}}{\max_{i'} R_{i'c} + \epsilon}. \tag{22}$$

Class assemblies and assembly overlap. For each class c, the class assembly is the top-K_A set

$$A_c = \text{TopK}_{K_A}\{a_i(c) : i \in E\}, \qquad K_A = 100. \tag{23}$$

The assembly-overlap proxy in Section 6 is the mean pairwise Jaccard overlap:

$$O_{\text{asm}} = \frac{2}{C(C-1)} \sum_{c<c'} J(A_c, A_{c'}), \qquad J(A, B) = \frac{|A \cap B|}{|A \cup B|}. \tag{24}$$

Lower O_{asm} corresponds to greater assembly separation.

Task-specific recurrent importance. For a task with class label c, each recurrent E→E synapse $j \to i$ receives the score

$$q_{ji}^{\text{EE}}(c; t) = M_{ji}^{\text{EE}} |W_{ji}^{\text{EE},(t)}| a_j(c) a_i(c). \tag{25}$$

The recurrent important-synapse set is

$$\mathcal{I}_c^{\text{EE}}(t) = \text{TopK}_{K_{\text{EE}}}\{q_{ji}^{\text{EE}}(c; t) : j, i \in E\}, \qquad K_{\text{EE}} = 4000. \tag{26}$$

Structural and learned span. For a synapse set $\mathcal{S}$ with pre- and postsynaptic coordinates $\mathbf{x}_{\text{pre}(e)}$ and $\mathbf{x}_{\text{post}(e)}$, the spatial span is

$$\text{Span}(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{e \in \mathcal{S}} \left\| \mathbf{x}_{\text{pre}(e)} - \mathbf{x}_{\text{post}(e)} \right\|_2. \tag{27}$$

The *structural span* samples 4000 E→E synapses uniformly from the recurrent mask M^{EE}. The *learned span* is $\text{Span}(\mathcal{I}_c^{\text{EE}}(t))$, averaged over tasks and runs after learning. The percentage reduction reported in Figure 2b is

$$\text{Reduction} = 100 \frac{\text{Span}_{\text{struct}} - \text{Span}_{\text{learned}}}{\text{Span}_{\text{struct}}}. \tag{28}$$

Structural-versus-learned span summary. For the main-text comparison in Section 4, the numerical summary is (mean ± SD across runs; p from a paired t-test of structural versus learned span across runs):

	Structural span	Learned span	Reduction	p
$\sigma = 0.4$	0.548 ± 0.005	0.458 ± 0.008	16.5%	$<10^{-4}$
$\sigma = 0.7$	0.788 ± 0.007	0.576 ± 0.006	26.8%	$<10^{-4}$
$\sigma = 1.0$	0.895 ± 0.003	0.631 ± 0.007	29.5%	$<10^{-4}$
Random	1.024 ± 0.002	0.718 ± 0.015	29.9%	$<10^{-4}$

Consecutive-task contention. For two consecutive classes c_t and c_{t+1}, recurrent synaptic contention is the Jaccard overlap between their important-synapse sets:

$$C^{\mathrm{EE}}_{t,t+1} = J\Big(\mathcal{I}^{\mathrm{EE}}_{c_t}(t), \mathcal{I}^{\mathrm{EE}}_{c_{t+1}}(t+1)\Big)\,. \tag{29}$$

Condition-level contention is the mean of $C^{\mathrm{EE}}_{t,t+1}$ over consecutive task pairs and random seeds. The input-to-E variant scores each input synapse $p \to i$ by $|W^{\mathrm{in},(t)}_{pi}|\, a_i(c)$ and keeps the top $K_{\mathrm{in}} = 2000$ as $\mathcal{I}^{\mathrm{in}}_c(t)$. Section 6, Figure 4, and Figure 8 report the average of the recurrent and input-to-E values, $\frac{1}{2}(C^{\mathrm{EE}} + C^{\mathrm{in}})$, and likewise for directional interference; Figure 2c–d uses the recurrent measure alone. With the recurrent measure alone, contention predicts accuracy as strongly (Spearman $\rho = -0.61$, $n = 35$).

Directional interference. Directional interference measures how much learning task $t+1$ rewrites the synapses that were important for task t:

$$I_{\mathrm{dir}}(t \to t+1) = \frac{1}{|\mathcal{I}^{\mathrm{EE}}_{c_t}(t)|} \sum_{(j,i)\in\mathcal{I}^{\mathrm{EE}}_{c_t}(t)} \Big| W^{\mathrm{EE},(t+1)}_{ji} - W^{\mathrm{EE},(t)}_{ji} \Big|\,. \tag{30}$$

The reported interference value averages Eq. 30 over consecutive task pairs and seeds. The input-synapse variant replaces $\mathcal{I}^{\mathrm{EE}}$ and W^{EE} with $\mathcal{I}^{\mathrm{in}}$ and W^{in}.

Distance-resolved contention. To obtain Figure 2d, shared important synapses are binned by pre–post Euclidean distance. For distance bin $B_\ell = [b_\ell, b_{\ell+1})$,

$$H_\ell(t, t+1) = \sum_{(j,i)\in\mathcal{I}^{\mathrm{EE}}_{c_t}(t)\cap\mathcal{I}^{\mathrm{EE}}_{c_{t+1}}(t+1)} \mathbb{I}[\|\mathbf{x}_j - \mathbf{x}_i\|_2 \in B_\ell]\,. \tag{31}$$

The plotted profile is the mean of H_ℓ over consecutive task pairs and seeds.

Pairwise rewriting selectivity. The pairwise assay uses a code-level binary split into similar and dissimilar digit pairs. To document the input-space basis of this split, we compute a class-prototype similarity from the MNIST training set. Let $\mathbf{x}_n \in [0,1]^{784}$ be a flattened MNIST image and let

$$\boldsymbol{\mu}_c = \frac{1}{N_c} \sum_{n:y_n=c} \mathbf{x}_n \tag{32}$$

be the mean image of digit class c. The pixel-prototype similarity of two classes is

$$S_{\mathrm{pix}}(a,b) = \frac{\boldsymbol{\mu}_a^\top \boldsymbol{\mu}_b}{\|\boldsymbol{\mu}_a\|_2 \|\boldsymbol{\mu}_b\|_2}. \tag{33}$$

The experiment script does not threshold all possible digit pairs at runtime. Instead, it uses the fixed default sets

$$\mathcal{P}_{\mathrm{sim}} = \{(3,5),(5,8),(3,8)\}, \qquad \mathcal{P}_{\mathrm{dis}} = \{(0,1),(1,4),(0,7)\}, \tag{34}$$

and assigns a trial to a group by sorting the unordered class pair and checking set membership. Pairs outside these sets are not used in the assay. The prototype-cosine audit confirms the intended split: the similar pairs have $S_{\mathrm{pix}} \in [0.876, 0.912]$, whereas the dissimilar pairs have $S_{\mathrm{pix}} \in [0.451, 0.650]$.

Pair	Group	S_{pix}
$(3,5)$	similar	0.898
$(5,8)$	similar	0.912
$(3,8)$	similar	0.876
$(0,1)$	dissimilar	0.451
$(1,4)$	dissimilar	0.542
$(0,7)$	dissimilar	0.650

In the pairwise assay, a network learns class A, the weights are snapshotted as W_A, and the class-A important set $\mathcal{I}_A$ is identified. The same network then learns class B, producing W_B. Rewriting of the class-A substrate is

$$R(A \to B) = \frac{1}{|\mathcal{I}_A|} \sum_{e\in\mathcal{I}_A} |W_B(e) - W_A(e)|. \tag{35}$$

Similarity selectivity is summarized by the ratio

$$\Gamma_{\mathrm{sim/dis}} = \frac{\mathbb{E}_{(A,B)\in\mathcal{P}_{\mathrm{sim}}} R(A \to B)}{\mathbb{E}_{(A,B)\in\mathcal{P}_{\mathrm{dis}}} R(A \to B)}, \tag{36}$$

where $\mathcal{P}_{\mathrm{sim}}$ and $\mathcal{P}_{\mathrm{dis}}$ are the fixed pair sets in Eq. 34.

D Supporting Mechanistic Assays

D.1 Similarity-Selective Rewriting Assay

Protocol. Each trial uses a single connectivity condition ($\sigma = 0.4$, $\sigma = 0.7$, or random control) and a single ordered class pair (A, B). The network is initialised, burned in, and trained on 10 MNIST samples of class A under the standard plasticity rules of Section 3.3. We then snapshot the post-class-A weights W_A and identify the *class-A functional important synapses* using the same functional criterion as Section 6: a recurrent E→E synapse $j \to i$ is class-A-important if both presynaptic neuron j and postsynaptic neuron i are class-A selective under spike-count selectivity ($k_{\mathrm{EE}} = 4000$, top-k over 4096^2 candidate edges). Without resetting weights, the same network is then trained on 10 samples of class B, producing W_B. Directional rewriting is the mean absolute weight change on class-A-important synapses, $\langle |W_B - W_A| \rangle_{\mathrm{important}}$. The code assigns each unordered class pair to the fixed similar/dissimilar sets in Eq. 34; the prototype-cosine scores in Appendix C document the input-space separation of those sets. Each (condition, pair) cell is repeated over three seeds, yielding nine measurements per group per condition. Statistical comparisons use a two-sided Mann–Whitney U test. The pairwise contention metric reported alongside is the Jaccard overlap between the class-A-important and class-B-important index sets, computed on the post-B snapshot.

Per-pair breakdown. Figure 7 shows the recurrent E→E assay broken down by individual class pair. Two regularities are visible. First, the similar-vs-dissimilar contrast is not driven by a single outlier pair: similar pairs consistently produce stronger rewriting in the local regime, while dissimilar pairs remain comparatively protected. Second, the similar-vs-dissimilar gap shrinks monotonically with locality: it is largest at $\sigma = 0.4$, intermediate at $\sigma = 0.7$, and nearly absent under random connectivity.

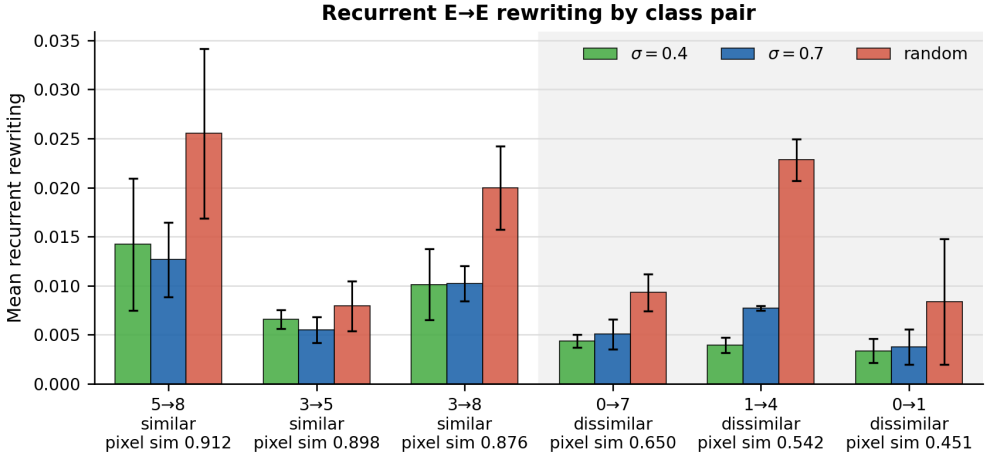


Figure 7: **Per-pair breakdown of similarity-selective rewriting.** Mean absolute weight change on recurrent E→E class-A-important synapses for each ordered class pair $A \to B$ (mean ± SD over three seeds; gray background: dissimilar pairs). At $\sigma = 0.4$, every similar pair is rewritten more than every dissimilar pair; at $\sigma = 0.7$ and under random connectivity, individual dissimilar pairs (e.g., $1 \to 4$) exceed some similar pairs, consistent with the smaller group-level contrast in Figure 5.

E Additional Benchmark-Level Adjudication

E.1 Directional Interference Tracks Contention

The main text frames adjudication around *consecutive-task synaptic contention*—the substrate-level Jaccard overlap of important-synapse sets across tasks. Continual learning analyses are more commonly framed around *directional interference*—the magnitude of weight change a later task imposes on synapses important for earlier tasks ($\langle |W_B - W_A| \rangle_{\text{important}}$). To make our account transparent to that literature, this appendix collects the interference-side evidence and shows that it carries largely the same information as contention.

Run-level bridge. Across the same five-run benchmark set used in Figure 4, contention is strongly associated with interference (Figure 8): runs with greater consecutive-task synaptic contention also exhibit substantially greater directional interference (Pearson $r = 0.91$, Spearman $\rho = 0.90$; $n = 35$). Because both measures are computed on the same important-synapse sets, the interference-side adjudication largely mirrors contention rather than providing independent evidence.

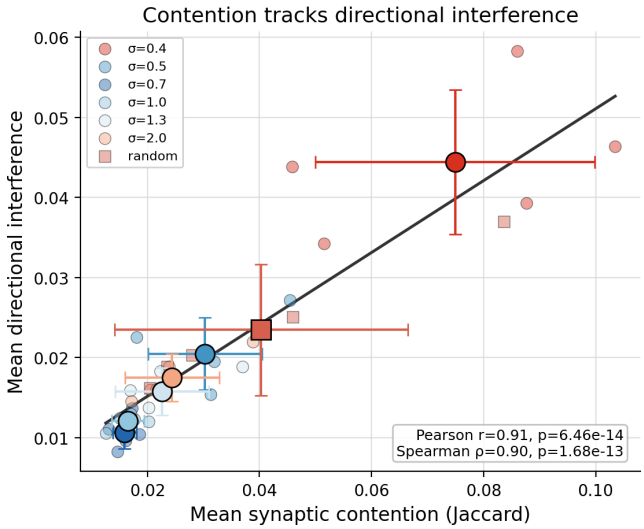


Figure 8: **Contention tracks directional interference.** Across the same five-run benchmark set used in Figure 4, consecutive-task synaptic contention is strongly associated with directional interference (Pearson $r = 0.91$, Spearman $\rho = 0.90$; $n = 35$). Large symbols show condition means $\pm$ SD.

F Representational Drift and Readout Diagnostics

Networks and probing. The readout diagnostics use networks with $\sigma = 0.4$, $\sigma = 0.7$, and random connectivity trained on the same class-incremental MNIST stream with 50 images per class (Table 2). All probes are run with plasticity disabled and network state reset before every image, using the training spike-count window.

Fixed-probe audit. For seeds 43, 44, and 46, the network is restored at the checkpoint after each class is learned and probed with 20 fixed held-out test images per class. Each class's readout column is frozen at the checkpoint where that class is learned. Acquisition accuracy is a class's accuracy at that checkpoint, BWT is final minus acquisition accuracy, and forgetting is peak minus final accuracy, each averaged over classes 0–8. The drift of class c after N new-class acquisitions is $1 - \cos(\mathbf{r}_c^{(a_c)}, \mathbf{r}_c^{(a_c+N)})$, where $\mathbf{r}_c^{(t)}$ is the class-mean excitatory spike-count vector over the probe images at checkpoint t and a_c is the checkpoint at which class c is learned.

Readout ladder. For seeds 42–46, final-state responses are collected for 30 calibration and 20 disjoint evaluation test images per class. The fixed readout is the selective vote accumulated during

sequential training. The recalibrated readout rebuilds the class-response matrix from the calibration responses with the same selective-vote rule. The refit decoder is a ten-way L2-regularized logistic regression (inverse regularization strength 1) on standardized spike counts, fitted to the calibration responses. All three readouts are scored on the same evaluation images of classes 0–8, and gains are computed within each paired seed before averaging.

Table 4 reports the full acquisition-versus-retention audit summarized in Section 8. Recalibrating the original selective-vote readout on final responses raised old-class accuracy by 22.6, 4.6, and 8.0 percentage points for $\sigma = 0.4$, $\sigma = 0.7$, and random connectivity (five paired seeds), the same ordering as the refit linear decoder (Figure 9c).

Table 4: **Acquisition is matched; retention is not.** Old-class (classes 0–8) accuracy immediately after each class was learned and at the end of the stream under the fixed selective-vote readout; backward transfer (BWT; final minus acquisition accuracy); forgetting (peak minus final accuracy); and representational drift at the end of the stream (1 minus the cosine similarity between a class's mean population response at acquisition and at the end). Mean ± SD over three paired seeds.

Connectivity	Acquisition (%)	Final old-class (%)	BWT (pp)	Forgetting (pp)	Drift
$\sigma = 0.4$	77.2 ± 6.8	52.4 ± 7.5	-24.8 ± 1.2	25.6 ± 0.6	0.089 ± 0.018
$\sigma = 0.7$	77.4 ± 2.7	63.1 ± 1.7	-14.3 ± 3.6	16.7 ± 1.5	0.065 ± 0.005
Random	76.3 ± 8.9	59.3 ± 1.4	-17.0 ± 10.0	22.0 ± 7.6	0.116 ± 0.010

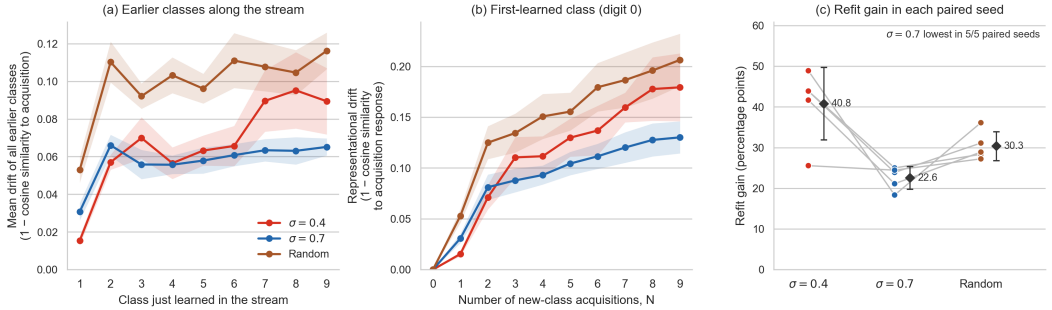


Figure 9: **Supplementary drift and readout analyses. (a)** Drift measured along the stream: after each class is learned, the mean drift of all earlier classes relative to their own acquisition (mean ± SD, three paired seeds). The final point equals the drift reported in Table 4. **(b)** Drift of the first-learned class (digit 0) after N new-class acquisitions (mean ± SD, three paired seeds); $\sigma = 0.7$ has the lowest final drift in each paired seed. **(c)** Refit gain (refit linear decoder minus fixed selective-vote readout) in each of five paired seeds (gray lines); diamonds show mean ± SD. $\sigma = 0.7$ has the lowest gain in every seed.

G Fashion-MNIST Cross-Dataset Validation

Figure 10 shows the Fashion-MNIST connectivity-range sweep (five seeds per condition). The intermediate regime again has the highest mean accuracy ($\sigma = 0.7$: $52.96 \pm 9.58\%$; $\sigma = 0.5$: $38.68 \pm 11.31\%$; $\sigma = 1.0$: $46.60 \pm 5.46\%$; random: $41.02 \pm 13.63\%$), but the paired difference between $\sigma = 0.7$ and random connectivity (+11.9 percentage points) is not significant (paired t-test, $p = 0.20$), so we treat Fashion-MNIST as directional support only.

Scope screens. Two single-seed screens bound the scope of our claims and are not evidence for or against DDC. First, CIFAR-10 [Krizhevsky, 2009], converted to grayscale and resized to 28×28 to match the input layer, reached 20.0%, 18.0%, and 21.0% selective-vote accuracy for $\sigma = 0.4$, $\sigma = 0.7$, and random connectivity (chance: 10%). Second, on MNIST, setting the heterosynaptic and transmitter-induced terms to zero ($\beta = \delta = 0$) while retaining triplet STDP and inhibitory plasticity reduced both the $\sigma = 0.7$ and random networks to chance (10.0%), compared with 53.0% for both under the full rule.

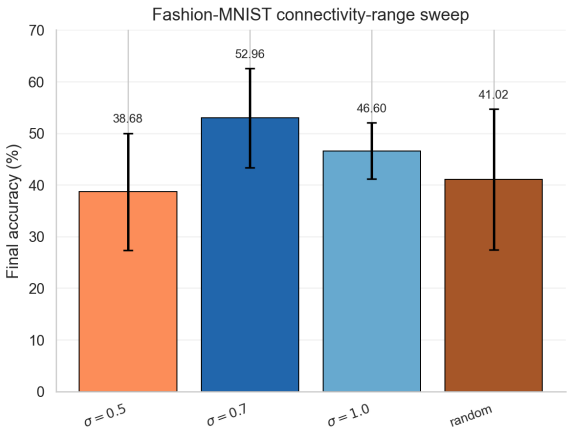


Figure 10: **Fashion-MNIST cross-dataset validation.** Connectivity-range sweep under the same protocol as MNIST (five seeds per condition).

H Reproducibility Notes

The main results are produced with `experiments/cl_sigma_sweep.py`, using the selective-vote readout. The MNIST numbers aggregate the original sweep together with later snapshot-enabled runs, yielding ten seeds per condition. The balanced snapshot-enabled subset used for the main interference and contention analysis contains five seeds per condition. The full snapshot-enabled appendix analyses include all available matched runs. The mechanistic figures are built from dedicated scripts for the synaptic-pool assay and the pairwise rewriting assay.

Compute resources. All experiments were run as single-GPU jobs on consumer NVIDIA hardware, primarily NVIDIA GeForce RTX 4090 devices with 24 GB VRAM. The benchmark sweeps, snapshot-enabled reruns, and figure-building scripts are all executable from the repository without distributed training infrastructure. A ten-class run with 100 training images per class takes about three hours on a single RTX 4090, so the 70-run MNIST sweep required roughly 200 GPU-hours. The full project, including preliminary and screening runs, used more compute than reported here; that total was not logged.

Datasets, software, and licenses. The datasets are MNIST [LeCun et al., 1998] (CC BY-SA 3.0), Fashion-MNIST [Xiao et al., 2017] (MIT License), and, for the scope screen, CIFAR-10 [Krizhevsky, 2009], which is distributed without an explicit license and is cited as its authors request; all were downloaded through `torchvision`. The implementation used Python 3.10.12 with PyTorch 2.2.2, torchvision 0.17.2, NumPy 1.26.4, SciPy 1.15.3, scikit-learn 1.7.1 [Pedregosa et al., 2011], seaborn 0.13.2, and Matplotlib 3.8.3, all under BSD-style licenses except Matplotlib, which uses its own PSF-based license. Code, the compact data behind every figure and table, seed manifests, and reproduction commands are released under the MIT License at `https://github.com/eason891023/DDC-Continual-Learning-NeurIPS2026` (see `REPRODUCE.md`).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and Introduction state the main claims, which Sections 4–8 support: DDC shapes the synaptic resource geography, an intermediate connectivity range performs best under a fixed readout, accuracy tracks consecutive-task synaptic contention more closely than assembly overlap, and old-class information remains decodable while its drift differs across regimes. They also state the scope: class-incremental MNIST serves as a controlled assay, and forgetting is measured relative to a fixed readout. Section 9.1 states the remaining limits.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 9.1 discusses three open questions: generality beyond MNIST and Fashion-MNIST (a CIFAR-10 screen stayed near chance, and we make no invariance claims); dependence on the plasticity rules of Zenke et al., the Gaussian kernel, and the simplified 3D substrate; and causality (we manipulate connectivity range but not pool geometry or plasticity, forgetting is measured relative to a fixed experimenter-side readout, and the readout diagnostics use three to five seeds).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.

- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: The paper is empirical and mechanistic; it does not claim formal theorems or proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Section 3 and Appendices A and C specify the network, connectivity prior, plasticity rules, input mapping, readout, the protocol of every analysis (seeds, images per class, test-state policy), and all metric definitions; Appendix H lists the hardware and software versions. The released repository additionally provides the code, seed and data-split manifests, and one command per figure and table.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.

(b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code (MIT License), the compact data behind every figure and table, seed and data-split manifests, and exact commands are available at `https://github.com/eason891023/DDC-Continual-Learning-NeurIPS2026` (`README.md`, `REPRODUCE.md`). The datasets are downloaded through `torchvision`. Raw simulation outputs, which include dense weight snapshots, are too large to host and are regenerated by the listed commands.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 3 describes the model, spatial prior, plasticity rules, input mapping, readout, and benchmark protocol. Appendix A gives the neuron and synapse equations, core hyperparameters, timing, and a protocol table for every analysis; Appendix C defines all metrics; Appendix H lists the datasets and software. The network is not trained with a gradient-based optimizer; learning is driven by the local plasticity rules, and Appendix F gives the settings of the diagnostic linear decoder.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All $\pm$ values and error bars are sample standard deviations (ddof $= 1$) across independent runs with different random seeds, which set the connectivity mask, the initial weights, and, in the benchmark, the training-image draw; captions and the protocol table in Appendix A give the number of runs. Tests are named where used: Welch's t-test ($\sigma = 0.7$ versus 1.0), a paired t-test (Fashion-MNIST), Pearson and Spearman correlations with p-values ($n = 35$ runs), paired t-tests for structural versus learned span, and two-sided Mann–Whitney U tests for the pairwise assay.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix H reports the hardware (single-GPU jobs, primarily NVIDIA GeForce RTX 4090 with 24 GB), the time of a ten-class run with 100 training images per class (about three hours), and the total for the 70-run MNIST sweep (about 200 GPU-hours). It also states that the full project, including preliminary and screening runs, used more compute than reported and that this total was not logged.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work is a simulation-based study on public benchmark datasets and does not involve human subjects, sensitive personal data, or undisclosed experimental manipulation.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This is foundational computational-neuroscience research on how connectivity shapes memory interference in a simulated spiking network trained on public image benchmarks. It releases no deployable system, no model with misuse potential, and no personal data, and we see no direct path to negative applications; its expected impact is scientific, linking continual-learning mechanisms to neuroscience hypotheses. To limit overinterpretation of a simplified simulation, Section 9.1 states the scope of the claims.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release a high-risk generative model, scraped dataset, or comparable asset with an obvious misuse profile.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Appendix H cites each dataset and states its license (MNIST, CC BY-SA 3.0; Fashion-MNIST, MIT License; CIFAR-10, distributed without an explicit license and cited as its authors request) and lists the software with versions and licenses (BSD-style, and the PSF-based Matplotlib license). The network follows the published rules of Zenke et al. [2015], which we cite and reimplemented.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We release code and compact data under the MIT License at `https://github.com/eason891023/DDC-Continual-Learning-NeurIPS2026`, documented by `README.md` (overview and installation), `REPRODUCE.md` (one command per figure and table), `results/README.md` (data dictionary), and `manifests/` (seeds, data splits, and analysis settings). No human-subject or personal data are involved.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.

- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The work does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The work does not involve human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are not part of the model, training procedure, analysis pipeline, or scientific contribution of the paper.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.

- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.