Life of Research · Published 4 November 2026 · DOI 10.5281/zenodo.23933365

A Residual Is Not a Third Factor: What the Gap Between a Perfect-Retention Ensemble and Joint Training Measures in Continual Learning, Why Chunk Size, Repetition and the Combiner Move It, and the Controls That Would Make Co-observation a Mechanism

Pranay Mahendrakar
SONYTECH · ORCID 0009-0003-7224-029X
pranaymahendrakar@sonytech.in · mahendrakarpranay@gmail.com · +91 6361723454

Abstract. Continual learning is scored against two reference points: naive sequential training below and joint training on all the data above. The usual reading is that the distance between them is made of forgetting and lost plasticity, so a learner that solved both would reach the upper line. A 2026 study disputes this. On vision benchmarks split into identically distributed chunks, a perfect-retention ensemble of sequential checkpoints still falls 2.2 to 3.9 linear-probe points short of incrementally joint training at the fourth chunk, and its authors name the residual a third facet, co-observation. In this analysis the author reviews and synthesises published results to show that the residual is real but that its name runs ahead of its evidence. Re-expressing the study's own tables, the author shows that the residual's share of the naive-to-joint gap ranges from 18 to 41 percent, smaller than the plasticity term in both self-supervised settings, that the claim that a 30 percent replay buffer closes half of it holds at the last chunk for Barlow Twins (60 percent) but not for supervised training (33 percent) or I-JEPA (14 percent), and that under the study's fixed per-chunk iteration budget each sequential member sees every example of its chunk about 100 times in one burst in supervised CIFAR-100 (about 90 in supervised ImageNet-100, and several hundred in the self-supervised settings, whose budget matches 400 joint epochs). On identically distributed chunks, seeing data together is the same thing as fitting each model on more data, so the residual is confounded with per-stage sample size and repetition, with the warm-start generalisation loss that its members also carry, and with the concatenate-and-project rule that combines them. Other studies point the same way: the sequential-to-joint gap shrinks with chunk size, falls to about one point for self-supervised pretraining on 320,000-image chunks, and closes to 0.02 in loss for language-model pretraining with 5 percent replay, while divide-and-conquer theory loses nothing in rate when local fits are regularised for the whole sample. The paper consolidates the measured values in a table, gives an attribution procedure, and specifies five controls that would turn the residual into a mechanism or dissolve it.

Keywords: continual learning; catastrophic forgetting; loss of plasticity; joint training; experience replay; warm-starting; data-incremental learning; evaluation methodology

1. Introduction

Almost every continual learning paper reports two numbers it does not try to beat. One is the accuracy of a model trained naively on the stream, one portion after another, with nothing done to protect what came earlier. The other is the accuracy of the same architecture trained on the union of all portions at once. The first is read as what forgetting costs and the second as what a perfect learner would achieve (De Lange et al., 2021; Wang et al., 2024). Between them sits the stability-plasticity dilemma: a learner must hold on to what it has learned without losing the ability to learn more (French, 1999; McCloskey and Cohen, 1989). An implicit claim follows from this framing. If a learner forgot nothing and kept its full capacity to learn, it would land on the upper line.

A 2026 study by Hess, Jha, van de Ven and Tuytelaars argues that this claim is false (Hess et al., 2026). Its design is careful. It removes distribution shift by splitting CIFAR-100 and ImageNet-100 into four identically distributed chunks, following the chunking setting of Lee and Storkey (2024). It evaluates representations rather than classifiers, training a fresh linear probe on all the data at each stage, so that a drifting output head cannot masquerade as forgetting. Probing showed that representation-level forgetting under naive training is often milder than output-level forgetting (Davari et al., 2022), and an earlier study by the anchor's first author showed that, measured relative to what was learned in a task, it is just as severe and slows the accumulation of general features (Hess et al., 2024). It controls forgetting by construction: after each chunk the sequential model's backbone is frozen and kept, and at evaluation the features of every kept checkpoint are concatenated, optionally compressed back to one backbone's width by principal component analysis, and probed. Nothing such an ensemble once encoded can be lost. Against this the study sets incremental joint training, a model warm-started at each stage and trained on all chunks seen so far. The ensemble falls short of incremental joint training at every stage after the first, in supervised training and in two self-supervised methods. Since the ensemble

© 2026 Pranay Mahendrakar. Licensed under CC BY 4.0.

cannot have forgotten, and incremental joint training shares its repeated optimisation, the authors attribute the residual to the absence of simultaneous observation, which they call the co-observation effect, and argue that it is a third facet of continual learning beside forgetting and plasticity.

The opposing position is not a single paper but a record of results that make a third, separate obstacle hard to see. Lee and Storkey (2024), who introduced the chunking setting the anchor builds on, analysed the same kind of drop and concluded that it is due to forgetting: the learner reaches 100 percent accuracy on each chunk, and its accuracy on that chunk's training data falls back to test level once it moves on. Hu et al. (2022) pretrained MoCo-v2 sequentially on four identically distributed quarters of ImageNet and found the transferred representation within about one point of joint pretraining with no continual learning method at all. Ibrahim et al. (2024) continued pretraining language models of 405 million and 10 billion parameters on a new corpus and, with learning-rate re-warming and 5 percent replay, matched retraining on the union to within 0.02 in validation loss. And in the statistical theory of divide-and-conquer estimation, fitting separate kernel ridge regressions on random partitions and averaging them loses nothing in rate, provided each local fit is regularised as if it had the whole sample (Zhang et al., 2013).

This paper's thesis is that the 2026 residual is a well-measured gap but not yet a factor. A residual becomes a factor when it is shown to respond to the variable its name refers to, and to nothing else that differs between the compared arms. In the anchor's design, at least four other variables move together with simultaneous observation: how many examples each optimisation run is fitted to, how many times each example is repeated within a stage, whether the ensemble's members carry the warm-start damage that Ash and Adams (2020) documented in stationary data, and the rule that combines several separately trained backbones into one representation. On identically distributed chunks, the first of these is not a confound in the usual sense; it is what "observing the chunks together" amounts to. A residual that cannot be separated from per-fit sample size is a statement that models trained on less data generalise worse and that concatenating them does not fully make up for it. That statement is true and useful, but it is weaker than a third facet of continual learning.

The paper makes five contributions. First, the author re-expresses the anchor's numerical appendix as a telescoping decomposition of the naive-to-joint gap, which shows the residual's share ranging from 18 to 41 percent and falling below the plasticity term in both self-supervised settings, and checks two of the anchor's claims about replay against its own tables, finding that each holds only in part (Section 3, Figure 1). Second, the paper shows that the decomposition depends on the order in which the baselines are subtracted, and that the order used counts plasticity loss in the naive trajectory as forgetting and warm-start damage in the ensemble's members as co-observation (Section 4). Third, it sets out four accounts that each predict the residual - per-stage sample size and repetition, the combiner, the object retained, and the time scale of interleaving - with the evidence for each from outside the anchor (Sections 5 to 8). Fourth, it places the claim in the wider record: where joint training is not an upper bound, where structure that spans partitions genuinely requires joint exposure, and what happens at scale (Sections 9 and 10, Table 1). Fifth, it proposes an attribution procedure (Algorithm 1) and five studies that would decide whether co-observation is a mechanism (Sections 12 and 14).

Section 2 fixes terms. Section 3 reads the anchor at the resolution of its tables. Sections 4 to 8 develop the competing accounts. Section 9 collects cases on both sides of the upper-bound assumption, and Section 10 the large-scale evidence. Section 11 consolidates the measured values, Section 12 states the procedure, Section 13 states the strongest objections, and Sections 14 to 17 list what would settle the question, what is not known, the limitations and the conclusion.

2. Scope and terms

The paper concerns supervised and self-supervised representation learning from a stream that is divided into portions, here called chunks, of which only one is available at a time unless a method stores data. Its central case is the data-incremental or chunking setting, in which the chunks are random partitions of one dataset and so are identically distributed (Lee and Storkey, 2024). Class-incremental and domain-incremental settings, in which chunks differ in distribution (van de Ven et al., 2022), enter where they bear on whether a separate cross-chunk factor exists.

Five terms are used in fixed senses.

Joint training is training one model from random initialisation on the union of all chunks, with mini-batches drawn from the whole union. **Incremental joint training** warm-starts at each stage from the previous stage's model and trains on the union of the chunks seen so far. **Naive sequential training** warm-starts at each stage and trains on the current chunk only. These follow the anchor (Hess et al., 2026).

Retention is ambiguous in a way that matters for the argument, and the paper distinguishes four objects a learner might retain about an earlier chunk: its parameters at the end of that chunk, the features those parameters compute, the function on the earlier chunk's inputs, and the earlier chunk's loss surface or a sufficient statistic of it. The anchor's ensemble retains the second. Exact sequential Bayesian updating retains the fourth.

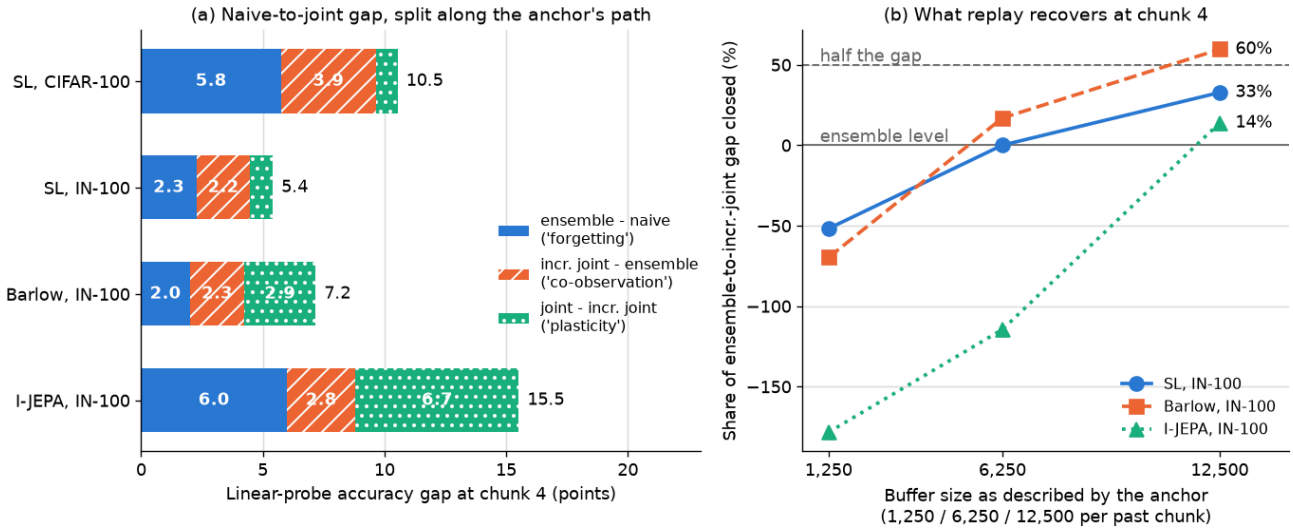


Figure 1. Panel (a): the gap between naive sequential and offline joint training at the fourth chunk, split into the three differences the anchor's design implies (ensemble minus naive, incremental joint minus ensemble, offline joint minus incremental joint), in linear-probe accuracy points. Panel (b): the share of the ensemble-to-incremental-joint gap recovered by replay with buffers of 1,250, 6,250 and 12,500 exemplars per past chunk (described by the anchor as about 3, 15 and 30 percent) at the fourth chunk on ImageNet-100; negative values lie below the ensemble. All accuracies are from Hess et al. (2026), Appendix F; the differences and shares are the author's arithmetic.

Co-observation is used in two senses that the paper keeps apart. The *operational* sense is the anchor's: the difference in linear-probe accuracy between incremental joint training and the forgetting-free ensemble. The *mechanistic* sense is the anchor's explanation of that difference: cross-chunk dependencies in the data that can only be exploited when gradients from several chunks shape the same features. The paper's question is whether evidence for the first is evidence for the second.

Interleaving time scale is the number of consecutive optimisation steps spent on data from one chunk before data from another chunk is used. Joint training interleaves at the scale of one mini-batch; naive sequential training at the scale of a whole stage; replay and periodic averaging sit in between.

All accuracy differences are reported in percentage points of the source's metric. Where the paper reports a difference, share or ratio that the source does not, it is the author's arithmetic on the source's published values and is labelled as such.

3. The anchor, read at the resolution of its tables

3.1 Design

The anchor trains ResNet-18 with cross-entropy on CIFAR-100 and ImageNet-100, ResNet-18 with Barlow Twins on ImageNet-100, and ViT-B/16 with I-JEPA on ImageNet-100, each split into four identically distributed chunks, all from scratch (Hess et al., 2026). Each stage runs a fixed number of iterations. For supervised CIFAR-100 the appendix gives 9,750 iterations per chunk at batch size 128, with a 5 percent warm-up, cosine decay from 0.1 to 0.00001, weight decay and gradient clipping; for ImageNet-100, 11,070 iterations per chunk at batch size 256. The per-chunk budget is set so that four stages together equal the budget of joint training, 100 epochs on CIFAR-100 and 90 on ImageNet-100 for supervised training and 400 epochs for Barlow Twins and I-JEPA. Linear probes are trained for 100 epochs on frozen features of the whole training set. Results are means and standard errors over three seeds, six for some I-JEPA arms. Replay stores 1,250, 6,250 or 12,500 exemplars per past chunk, described as about 3, 15 and 30 percent, merged with the current chunk and sampled uniformly. Distillation uses Learning without Forgetting for the supervised model (Li and Hoiem, 2017) and CaSSLe for Barlow Twins (Fini et al., 2022).

One consequence of the budget is worth stating because it is the basis of Section 5. On CIFAR-100, 9,750 iterations at batch 128 draw 1,248,000 samples per stage. A chunk holds 12,500 of the 50,000 training images, so a naive sequential model passes over each example of its chunk about 100 times within one stage, under a full warm-up-and-decay schedule. Incremental joint training at stage four spreads the same 1,248,000 samples over 50,000 images, about 25 passes each. Across the four stages, an image from the first chunk is seen about 208 times by incremental joint training (100, 50, 33 and 25) and about 100 times by the naive arm, the ensemble and offline joint training, but in incremental joint training the later passes are interleaved with ever more data. (This is the author's arithmetic on the anchor's stated budget.) The anchor's ensemble is therefore a set of four backbones, each the end point of a stage in which a warm-started network was driven through a full schedule on one quarter of the data.

3.2 The decomposition the tables imply

The anchor describes the expected picture qualitatively: the difference between ensemble and naive measures forgetting, the difference between offline joint and incremental joint measures lost plasticity, and the remaining difference between incremental joint and the ensemble is the co-observation effect (Hess et al., 2026, Figure 2). Its appendix reports the four arms at each stage. Taking the fourth stage, the three differences add to the total naive-to-joint gap by construction. Figure 1(a) shows them.

In supervised CIFAR-100 the naive arm reaches 64.64, the ensemble 70.39, incremental joint 74.29 and offline joint 75.18. The gap of 10.54 points splits into 5.75 for the ensemble's gain over naive (55 percent), 3.90 for the residual (37 percent) and 0.89 for the plasticity term (8 percent). In supervised ImageNet-100 the values are 78.27, 80.55, 82.75 and 83.68: 2.28, 2.20 and 0.93 of a 5.41-point gap, or 42, 41 and 17 percent. For Barlow Twins they are 69.25, 71.25, 73.50 and 76.42: 2.00, 2.25 and 2.92 of 7.17, or 28, 31 and 41 percent. For I-JEPA they are 53.83, 59.82, 62.64 and 69.35: 5.99, 2.82 and 6.71 of 15.52, or 39, 18 and 43 percent. With the capacity-controlled ensemble the residual becomes 4.04, 2.18, 1.97 and 3.81 points.

Three things follow. First, the residual is present in every setting, it is larger than its standard errors (under 0.7 points for every one of the four main arms at the fourth chunk), and it is the same order of size as the forgetting term in three of four settings. The anchor's empirical core holds on its own numbers. Second, the residual is not the dominant term. In both self-supervised settings it is smaller than the plasticity term; for I-JEPA it is under a fifth of the gap and the plasticity term is more than twice as large. The anchor says this in its own words, noting that it does not claim the penalty universally rivals forgetting (Hess et al., 2026). Third, in I-JEPA the naive arm gets worse as data accumulates, falling from 55.87 at the first chunk to 53.83 at the fourth, while the ensemble rises to 59.82. Part of what this design labels forgetting in that setting is the naive network becoming a worse learner while it trains on more data. That reading is taken up in Section 4.

3.3 Replay and distillation, checked against the tables

The anchor makes two quantitative claims about replay on ImageNet-100: that storing about 15 percent of each past chunk is enough for replay to match the ensemble, and that a 30 percent buffer pushes past the ensemble and closes half the gap to incremental joint training, with results said to hold for supervised training, Barlow Twins and I-JEPA (Hess et al., 2026). Figure 1(b) expresses the replay results as a share of the ensemble-to-incremental-joint gap at the fourth chunk.

The 15 percent claim holds for supervised training (80.55 against an ensemble of 80.55) and Barlow Twins (71.63 against 71.25). It does not hold for I-JEPA, where the 15 percent buffer reaches 56.59 against an ensemble of 59.82, below the ensemble by more than the whole ensemble-to-incremental-joint gap. Against the capacity-controlled I-JEPA ensemble of 58.83 it is still 2.24 points short.

The half-the-gap claim depends on stage and setting. With the 30 percent buffer, supervised replay recovers 27, 56 and 33 percent of the gap at chunks two, three and four; Barlow Twins recovers 62, 56 and 60 percent against the unreduced ensemble; I-JEPA recovers 31, 11 and 14 percent against the unreduced ensemble, or 65, 35 and 36 percent against the capacity-controlled one. "Half" is a fair summary of Barlow Twins and of supervised training at the third chunk; it overstates supervised training at the last chunk and I-JEPA at every chunk after the first against the unreduced ensemble (at chunks three and four against the capacity-controlled one). (All shares are the author's arithmetic on Appendix F.)

Distillation behaves as the anchor says in supervised training at the middle stages, where Learning without Forgetting tracks the ensemble, but at the fourth chunk it falls 0.62 below it (79.93 against 80.55), recovering minus 28 percent of the gap. For Barlow Twins, CaSSLe reaches 72.07, above the unreduced ensemble by 36 percent of the gap. Distillation is thus not uniformly at the ensemble level; it is below it in one setting and above it in the other.

These are corrections of degree, not of direction. Replay with larger buffers does move the representation toward incremental joint training in every setting, and the anchor's qualitative point stands. But the claim on which the third-facet reading of replay rests, that replay does something distillation cannot because it restores joint exposure, needs the two to differ at matched retention, and the tables show that they differ by setting as much as by method.

3.4 What the anchor establishes

The anchor establishes that, in its four settings at its scale and budget, a concatenation of sequential checkpoints, probed linearly, is worse than a single network trained incrementally on the accumulated data, by 2 to 4 points at the fourth chunk. It establishes that the shortfall survives compressing the ensemble to one backbone's width, and that replacing naive checkpoints by distillation-trained ones moves the ensemble little (Hess et al., 2026, Appendix C.2). It shows a proof-of-principle on MNIST in which the evaluation task requires combining a decision rule learned from one chunk with robustness to noise learned from another, and joint training transfers the noise robustness while the ensemble does not.

What it does not establish is that the shortfall in the chunking setting is caused by the absence of

simultaneous observation as such, rather than by any of the other differences between a backbone trained on one quarter of the data and a backbone trained on all of it. The MNIST construction does have a genuine cross-chunk dependency, built in by design. The chunking experiments, by construction, do not: every chunk is a random sample of the same distribution. The remaining sections take the alternatives in turn.

4. A telescoping decomposition depends on its path

The three terms in Figure 1(a) add up to the total because each is the difference between neighbours in a chosen sequence of arms: naive, then ensemble, then incremental joint, then offline joint. Economists know this kind of decomposition well. When a total difference is split by switching on factors one at a time, the share attributed to each factor depends on the order in which they are switched on unless the factors do not interact, and the usual remedy is to average over orders, which is the Shapley decomposition (Fortin et al., 2011; Shorrocks, 2013). The anchor uses one order and does not report any other.

The order matters here because the arms differ on more than one factor at each step. Moving from naive to ensemble adds retention, but it also changes the representation from one backbone to four concatenated backbones, which changes what a linear probe can read. Moving from ensemble to incremental joint adds joint exposure, but it also replaces four backbones by one, increases the number of distinct images in each stage's fit from one quarter of the data to all of it, and lowers the number of passes per image within the last stage from about 100 to about 25. Moving from incremental joint to offline joint removes warm-starting, but warm-starting is present in every arm except the last, including every member of the ensemble after the first.

Two consequences follow. The first is that plasticity, in this design, is measured on only one path. The plasticity term is the difference between incremental and offline joint training. The naive trajectory also loses plasticity, and in I-JEPA visibly so, since the naive model gets worse with more data. That loss is counted inside the ensemble-minus-naive term and labelled forgetting. The second is that the ensemble's members are warm-started networks fitted to quarter-size chunks. If warm-starting on stationary data damages generalisation, as Ash and Adams (2020) and Berariu et al. (2021) report and as Section 5 details, the members carry that damage, and the design counts it in the residual. Incremental joint training also carries warm-start damage, so the two partly cancel, but not exactly: the damage depends on how much of the data the earlier fit saw (Ash and Adams, 2020), and the members and the incremental joint model saw different amounts.

A path-independent version of the analysis would need at least two further arms: a cold-start ensemble, each member trained from scratch on its own chunk, and a cold-start incremental model trained from scratch at each stage on the accumulated data. With them, warm-starting and joint exposure could be varied independently, and their interaction estimated. The anchor's appendix ablation, which rebuilds the ensemble from distillation-trained members, changes the members' retention but not their warm start or their sample size, so it does not address this.

5. Account one: per-stage sample size and repetition

5.1 On identically distributed chunks, co-observation is more data per fit

The anchor's conceptual argument supposes two kinds of features: shared features that predict across the whole distribution, and local shortcuts that suffice within one chunk. A network trained on one chunk adopts the shortcuts, because simplicity bias favours them (Shah et al., 2020), and the shared features become unidentifiable from its parameters (Hess et al., 2026, Section 2.1). For chunks drawn from different distributions, this picture is coherent and supported, as Section 9 shows. For identically distributed chunks it needs restating. Every feature that is predictive across the distribution is, in expectation, equally predictive in every chunk. A chunk differs from the whole only by sampling. The "local" features available to a network on one chunk are therefore features that fit the particular sample: in the language of the warm-start literature, memorised noise (Shin et al., 2024). Read this way, co-observation in the chunking setting is the statement that a network fitted to n/4 examples learns less transferable structure, and more sample-specific structure, than one fitted to n examples, and that four such networks concatenated do not recover the difference.

That statement is close to a restatement of finite-sample generalisation, and its size should scale with how far a chunk is from enough data. The evidence supports this.

5.2 The gap shrinks with chunk size

Lee and Storkey (2024) varied the chunk size directly. On CIFAR-100 with a ResNet-18, plain SGD and every continual learning method they tested reached about 73 percent with all data in one chunk and about 45 percent with chunks of 1,000 examples; the drop grew smoothly as chunks shrank, on CIFAR-10, CIFAR-100 and Tiny ImageNet. With DER++ and a ten-way split, chunking alone cost 10.37 points on CIFAR-100 (73.72 offline against 63.35) and 10.09 on Tiny ImageNet (60.63 against 50.54), about half of the full class-incremental drop in each case. Their analysis of where the loss comes from points at the same mechanism as Section 5.1: the learner reaches 100 percent accuracy on the current

chunk, its accuracy on a chunk's training data falls to test level soon after it moves on, and they raise overfitting to the current chunk as a candidate cause. Their remedy, averaging the weights reached at the end of each chunk, improves accuracy and reduces forgetting, which is consistent with, though it does not show, each end point carrying chunk-specific noise that averaging cancels.

Hu et al. (2022) supply the large-chunk end of the same curve. They split ImageNet-1K into four identically distributed chunks of about 320,000 images, trained MoCo-v2 with a ResNet-50 for 200 epochs per chunk, and evaluated by linear probing on twelve downstream datasets. Naive sequential self-supervised training ended 1.04 points below joint training on average at the fourth chunk; supervised training ended 4.83 below. The anchor's self-supervised settings split a 100-class subset of ImageNet, so each chunk holds roughly a tenth as many images and report naive-to-joint gaps of 7.17 points for Barlow Twins and 15.52 for I-JEPA, measured in-distribution. The comparison crosses objectives, evaluations and architectures and cannot be made exact, but its direction is the one the sample-size account predicts and the anchor's genericity claim does not.

5.3 Warm-starting damages generalisation in stationary data, and the damage depends on the first fit

Ash and Adams (2020) trained an 18-layer ResNet on half of CIFAR-10 and then on all of it, and compared it with the same network trained on all of it from random initialisation. Both reached full training accuracy. On held-out data the warm-started network was worse: 51.7 against 56.2 percent with SGD and 74.4 against 78.0 with Adam, in a regime without data augmentation or learning-rate schedules. Logistic regression, whose loss is convex, lost 0.9 points with SGD (39.6 against 40.5), a difference they did not count as significant. The gap grew as the fraction of data in the first round shrank, and appeared after only a few epochs of first-round training. Berariu et al. (2021) reproduced the effect across optimisers, found that a few passes over the pretraining data are enough to produce a gap, before any point at which early stopping would apply, and found that it persists when the transition from the subset to the full set is made gradually. Shin et al. (2024) gave an account in a feature-learning framework, in which warm-starting on stationary data leads the network to memorise noise in the first data it fits, and proposed shrinking the weights selectively so as to forget memorised noise while keeping learned features, which improved test accuracy and training efficiency on vision tasks. Lee et al. (2024) revisited the Ash and Adams experiments and found that methods designed to maintain trainability provide limited benefit to generalisation, distinguishing the two senses of plasticity.

This stationary-data effect belongs to a wider literature on how early training constrains later learning. Deficits imposed during the first epochs of training can permanently impair a network even after they are removed, and information in the weights rises early and then becomes hard to redistribute (Achille et al., 2019). In reinforcement learning, transient non-stationarity during training can permanently affect the latent representation and harm generalisation, and repeatedly transferring the current policy into a freshly initialised network improves generalisation (Igl et al., 2021), and agents trained on growing datasets over-rely on early experience, which periodic resets of part of the network correct (Nikishin et al., 2022). In continual supervised and reinforcement learning, standard networks gradually lose the ability to learn new tasks over long sequences (Dohare et al., 2024); the loss has been tied to changes in the curvature of the loss landscape and often occurs without saturated units (Lyle et al., 2023) and decomposed into several independent mechanisms, no one of which explains every case (Lyle et al., 2024). On the other side, deliberate partial forgetting can improve generalisation (Zhou et al., 2022), and re-initialisation helps in the absence of other regularisation but adds little once data augmentation, weight decay and learning-rate schedules are tuned (Zaidi et al., 2023). The last result is a reminder that the size of any warm-start effect depends on the regularisation regime, which in the anchor includes augmentation, weight decay and a cosine schedule.

Two things about this literature matter for the anchor. The first is that the warm-start loss is a loss of generalisation, not of trainability, and it occurs without any distribution shift: the same phenomenon the anchor calls co-observation, observed in a design with no ensemble and no chunks beyond two. The second is that the anchor's own design contains it in two places. Incremental joint training is a warm-started model, and its gap to offline joint training is the anchor's plasticity term. Every member of the ensemble after the first is also a warm-started model, fitted on one quarter of the data. Whether the residual measures something beyond the warm-start damage of the members is exactly what a cold-start ensemble would show.

5.4 Repetition, and why a fixed iteration budget does not hold it fixed

The anchor argues that because the training budget is identical in every arm, larger replay buffers cannot be helping merely by providing more optimisation (Hess et al., 2026). The total number of iterations is indeed fixed. The number of passes over each example is not. With a fixed number of iterations per stage, a larger buffer enlarges the stage's dataset and lowers the number of times each example is revisited. A naive model at stage four on CIFAR-100 visits each of its 12,500 images about 100 times; incremental joint training visits each of 50,000 images about 25 times; a replay arm sits in

between, according to its buffer.

The effect of repetition on generalisation is well documented outside continual learning. For language models trained under a fixed compute budget, Muennighoff et al. (2023) found that repeating data for up to four epochs changes loss negligibly compared with unique data, while further repetition yields rapidly diminishing returns. Hernandez et al. (2022) found that repeating a small fraction of data many times can degrade a model severely: repeating 0.1 percent of the data 100 times brought an 800-million-parameter model down to the level of one half its size. Neither study concerns vision or chunked training, and the paper does not transfer their numbers. They establish that the count of passes per example is a variable with its own effect on generalisation, and that it moves with buffer size in the anchor's replay arms.

Within continual learning, the evidence on repetition of stored data points both ways, which is itself a reason to measure it rather than assume it away. Chaudhry et al. (2019) found, in a single-pass setting, that training repeatedly on a tiny episodic memory did not harm generalisation and improved it, by 7 to 17 percent with one stored example per class. Verwimp et al. (2021) argued, with conceptual and empirical support on three benchmarks, that rehearsal keeps a model in the low-loss region of earlier tasks but risks overfitting to the stored sample. The anchor's own finding that 3 percent buffers fall below the ensemble in every setting, while 30 percent buffers rise above it, is compatible with both. One possible reading, which no study has tested, is that small buffers are revisited often enough to be overfitted, while large ones dilute the current chunk enough to reduce its repetition.

5.5 What this account predicts

If the residual is mainly per-stage sample size and repetition, then (i) it should shrink as chunks grow and vanish when each chunk is large relative to what the model needs; (ii) at a fixed chunk size, it should shrink when passes per example are reduced, for instance by early stopping each stage or by fewer iterations per chunk; (iii) a replay buffer should help in proportion to how much it enlarges the stage's dataset, whether or not the replayed data come from earlier chunks, so that adding the same number of fresh, never-seen examples from the same distribution should help as much as replaying old ones; and (iv) a cold-start ensemble should show a residual at least as large as the warm-start ensemble's, since its members are fitted to the same quarter-size samples. Prediction (iii) is the sharpest, because the co-observation account predicts the opposite: fresh same-distribution data adds sample size but no cross-chunk exposure. No published study has run it.

6. Account two: the combiner

6.1 The baseline's shortfall is partly a property of how it is combined

The ensemble is described as a perfect-retention baseline, and in one sense it is: on the training data of every past chunk, its probe accuracy never falls (Hess et al., 2026, Appendix C.1). But perfect retention of four representations is not a representation as good as the best single network could have built from the same four chunks. The residual compares a particular combination rule, concatenation of frozen features followed by an optional linear projection and a linear probe, with a single network trained on the union. A shortfall can come from the members, from the combination, or from both.

The statistical theory of divide-and-conquer estimation is direct on this point. Zhang et al. (2013) partition N examples randomly into m subsets, fit kernel ridge regression on each, and average the fits. They prove that the averaged estimator attains the minimax rate over all estimators that use the N examples, as long as m does not grow too fast. The condition that makes this work is specific: each local problem has to be regularised as though it had all N samples, so that from its own point of view it is under-regularised. Its bias stays small and its variance blows up, and averaging across partitions removes the variance. With each local fit regularised for its own sample size, the averaged estimate is worse. Separate fitting, in other words, loses nothing in this model class when the local fits are tuned for the combination and the combiner is an average; it loses when the local fits are tuned for themselves.

The anchor's members are tuned for themselves: each stage runs a full schedule to a learning rate of 0.00001 on its own chunk. Its combiner is concatenation and projection rather than averaging in function or parameter space. The theory is for kernel ridge regression, not for deep networks, and it does not show that a better combiner would close the anchor's residual. It shows that the existence of a residual between separate fits and a joint fit depends on how the fits are made and combined, which is the variable the anchor holds fixed.

6.2 Separately trained models, combined differently, sometimes beat joint training

At language-model scale there is direct evidence that training on separate portions of the data and combining at inference need not cost anything. Branch-Train-Merge trains expert language models independently on different domains, branching each from existing experts, and combines them by ensembling or by weight averaging; compared with dense transformer language models at matched training cost, it reported better in-domain and out-of-domain perplexity (Li et al., 2022). The same authors report that these results "require expert domain

specialization; LM ensembles with random data splits do not perform well" (Li et al., 2022). Random splits are the case closest to the anchor's random chunks, so on random partitions this result favours the anchor's reading. Its successor clusters an arbitrary corpus, trains one expert per cluster without communication, and combines them sparsely at inference; it outperformed dense baselines on several corpora and few-shot tasks, and its authors found that specialising experts to meaningful clusters was key to the gains (Gururangan et al., 2023). The second finding cuts in an interesting direction for the anchor. Separate training paid off when the partition followed the data's structure, which the anchor's random chunks do not. These systems differ from the anchor in many ways, including initialisation from shared seed models and routing at inference, and the paper uses them only to show that the sign of the separate-versus-joint comparison is not fixed.

Weight-space combination gives a more mixed picture. Averaging the weights of models fine-tuned from one pretrained network with different hyperparameters often improves accuracy over the best single model (Wortsman et al., 2022). Merging models fine-tuned on different tasks into one model, by contrast, can lose accuracy through interference between their parameter updates, from redundant values and from disagreement in sign, which is what methods such as TIES-Merging were designed to reduce (Yadav et al., 2023). Per-chunk weight averaging improves the chunking setting (Lee and Storkey, 2024). Across these results, how much is lost by separate training depends heavily on the combination rule and on whether the parts share a starting point, which is the point of this section.

6.3 Ensembles combine features differently from single networks

There is also a reason to expect a concatenated ensemble of separately trained networks to read differently from one network, even with no data partition at all. In the theory of Allen-Zhu and Li (2023), when data have several independent views that each predict the label, each independently trained network learns a subset of them, and an ensemble's gain comes from covering more views; that gain can be distilled into a single network. Last-layer retraining results point the same way from the other side: networks that rely on simple spurious features often still learn the core features, which a re-trained linear head can then use (Kirichenko et al., 2023). Both results show that what a linear probe on frozen features can read depends on which features each backbone happened to learn and how they are pooled, not only on what data each saw. The anchor's PCA control addresses the width of the concatenation; it does not address whether four networks' features, pooled and projected, are the right comparison for one network's features.

6.4 What this account predicts

If the combiner carries part of the residual, then changing it should move the residual at fixed members: averaging member predictions, training one student by distillation from the ensemble on only the current chunk's inputs, or averaging member weights where they share an initial model. It also predicts that members trained deliberately for combination, for instance with weaker regularisation or earlier stopping, should form a better ensemble than members trained to the end of a full schedule, even though each member alone is worse. The second prediction is the divide-and-conquer theory's, carried over without proof to deep networks.

7. Account three: what is retained

7.1 Perfect retention of parameters is not perfect retention of information

The third-facet reading rests on the idea that a learner with perfect stability and plasticity can still miss what joint training finds. That is true for a learner that retains the wrong object. It is not true for every learner. In exact Bayesian inference, updating on one chunk and then on the next yields the same posterior as updating on their union. A learner that carried the full posterior from chunk to chunk would lose nothing by seeing the chunks apart. Practical continual learning methods approximate this: elastic weight consolidation keeps a quadratic approximation to the earlier loss around the earlier solution (Kirkpatrick et al., 2017), which Huszar (2018) read as an approximation to Laplace propagation, showing that its quadratic penalties, as originally stated for more than two tasks, are inconsistent with that derivation and may double-count earlier data; variational continual learning propagates an approximate posterior (Nguyen et al., 2018). Each fails to the extent that its approximation discards information about the earlier loss surface.

Theory states the requirement sharply. Knoblauch et al. (2020) show that a continual learner that is optimal in their sense, never forgetting, must in general solve an NP-hard problem and must have perfect memory, which they define as storing enough of the past to distinguish the relevant sets of hypotheses, and they note that this explains why replay and core-set methods outperform regularisation. Peng et al. (2023) define an ideal continual learner that never forgets by construction, under the assumption that all tasks share a common global minimiser, and connect it to multitask learning and to rehearsal. Neither result is about co-observation, but together they say that what separates a sequential learner from a joint one, in principle, is how much information about earlier data it keeps, not the bare fact that the data arrived apart.

Seen this way, the anchor's ensemble is a learner that retains the second of the four objects in Section 2, the features at the end of each stage. It does not retain the

earlier chunks' loss surfaces, and it cannot use later data to revise what earlier features encode. Replay retains a sample of the earlier data, which is a noisy estimate of the fourth object, and so it should, and does, recover more as the sample grows. Calling what is recovered "co-observation" and calling what is retained "perfect retention" puts a name on the difference between retaining features and retaining data that is accurate, but the name suggests a separate process where the evidence shows a difference in what is kept.

7.2 Forgetting and distance from the joint solution are different quantities even in linear models

A second theoretical result shows that the anchor's dissociation, low forgetting with a remaining gap to joint training, is not specific to deep networks or to a third mechanism. Evron et al. (2022) analyse an overparameterised linear model fitted to a sequence of tasks. They bound forgetting when tasks recur cyclically, by a quantity that falls with the number of iterations, and contrast it with convergence to the offline solution, which, by results on alternating projections, can be arbitrarily slow. A linear learner can therefore forget little while remaining far from what joint training would reach. Lin et al. (2023) give explicit expressions for forgetting and generalisation error in overparameterised linear models and analyse how overparameterisation, task similarity and task ordering affect each. In both analyses the gap to the joint solution follows from the geometry of sequential projection; no additional factor is needed.

7.3 Features learned but not used

The conceptual argument also supposes that shared features, once not needed on the first chunk, are never learned. Results on spurious correlation suggest that networks often learn more than they use. Kirichenko et al. (2023) found that classifiers relying on backgrounds still encoded core features that a re-trained last layer could exploit. Shah et al. (2020) found the opposite extreme in their synthetic datasets, where networks relied exclusively on the simplest feature and remained invariant to more complex predictive ones. Which regime applies to the anchor's chunks is not known, and the linear probe the anchor uses is precisely the tool that would reveal learned-but-unused features. Since its probes are trained on all the data, the residual shows that the ensemble's features support less accuracy, not that the shared features were never learned.

8. Account four: the time scale of interleaving

8.1 Joint training does not observe data simultaneously either

Joint training with stochastic gradient descent never sees all the data at once. It sees one mini-batch at a time, and the "joint" in its name refers to the interleaving of data from the whole distribution on the time scale of a single step. Replay with a small buffer interleaves earlier data at a lower rate. Naive sequential training interleaves on the time scale of a stage. Co-observation, in the operational sense, is therefore not a binary condition but a position on a continuum of interleaving time scales, and the anchor's design samples only the two ends.

The distributed optimisation literature has studied this continuum in detail. Local SGD runs SGD independently on several workers and averages their parameters every so often. Stich (2019) proved that for convex problems it converges at the same rate as mini-batch SGD in the number of gradients evaluated, while communicating far less. Lin et al. (2020) found that a variant that switches to local updates late in training generalises better than large-batch training. Woodworth et al. (2020) proved that for quadratic objectives local SGD strictly dominates mini-batch SGD, and gave a lower bound showing that for general convex objectives it does not dominate. Federated averaging, which runs several local epochs on each client between averages, was robust to non-identically distributed client data in its original evaluation (McMahan et al., 2017); later work found accuracy losses of up to 55 percent when every client holds a single class, explained them by divergence of the clients' weights, and raised CIFAR-10 accuracy by 30 percent by sharing a global subset of 5 percent of the data with all clients (Zhao et al., 2018). The pattern across these results is that separating optimisation into long local runs costs little when the parts are identically distributed and the combination is frequent, and costs a great deal when the parts differ.

8.2 Re-occurrence and resampling recover what one pass cannot

If the cost of separate training is a matter of time scale, revisiting chunks should recover it. Lesort et al. (2022) trained deep networks with plain SGD on long sequences of tasks in which data re-occur sparsely, and found that overall accuracy improves over the sequence despite catastrophic forgetting at each step; forgetting had a limited effect on knowledge accumulation. Lee and Storkey (2024) make the same point for chunking: if a learner could resample chunks it would approach offline accuracy, which is standard for online learning. Neither result separates sample size from interleaving, since revisiting both adds exposures and shortens the effective time scale. Both show that the deficit of a single sequential pass is not a fixed property of having seen the data apart.

8.3 Interleaving and blocking in human learning

Human category learning is often taken as the natural analogue, and it does not give a uniform verdict. Kornell and Bjork (2008) found that interleaving paintings by different artists improved later induction of each artist's style compared with presenting each artist's paintings consecutively, even though learners judged the consecutive schedule more effective. Carvalho and

Goldstone (2014) found that whether interleaving beats blocking depends on the structure of the categories and on the temporal arrangement. Flesch et al. (2018) compared humans and neural networks learning two tasks, and found that humans, but not the networks, benefited from blocked training, especially when they started with a bias to keep the tasks apart. The complementary learning systems account explains the brain's reliance on slow, interleaved learning in neocortex as a way to avoid interference while extracting shared structure (McClelland et al., 1995). The human record supports the claim that the schedule of exposure changes what is learned. It does not support the claim that simultaneous exposure is always better, and in the one direct comparison located here, the schedule that suited humans was not the one that suited networks.

8.4 What this account predicts

If interleaving time scale is the operative variable, then at a fixed set of chunks and a fixed total budget the residual should shrink smoothly as the number of consecutive steps per chunk falls from a full stage toward one mini-batch, with no further benefit once the time scale is short relative to how fast the network's features drift. This prediction can be separated from the sample-size account by holding the stage's data fixed and varying only the schedule.

9. Where joint training is and is not the upper line

The anchor's reading is strongest where chunks differ in distribution and weakest where they are identical. The published record shows both cases.

Cross-partition structure that only joint exposure reveals. In class-incremental learning, a classifier must separate classes that were never seen together. Soutif-Cormerais et al. (2021) ablated the learning of cross-task features with basic replay and concluded that forgetting is not the main cause of low class-incremental accuracy, and that methods must also improve the quality of cross-task features, especially when tasks have little data. Busch et al. (2025) built a continually confounded dataset in which a spurious correlation that differs across tasks is easy to ignore under joint training but hard to avoid sequentially, and standard continual learning methods failed to ignore it. The anchor's MNIST construction is of the same kind. In these cases, joint exposure supplies information that no single chunk contains, and the mechanistic sense of co-observation applies.

Joint training is not always the upper line. Wu et al. (2022) designed a continual self-supervised setting in which each task learns invariance to one family of augmentations, and found that continual learning often beat multitask learning, which they attribute to tasks that act adversarially when learned jointly. Multitask learning itself has a long record of tasks that compete: grouping tasks into separate networks can beat one joint network (Standley et al., 2020), and conflicting gradients across tasks are one identified cause (Yu et al., 2020). The classical case for joint training, that related tasks supply a mutually useful inductive bias (Caruana, 1997; Baxter, 2000), is a claim about related tasks. Curriculum results suggest that for standard benchmarks the order of examples matters little: randomly ordered schedules did as well as or better than curricula and anti-curricula (Wu et al., 2021).

Even full replay is not joint training. The incremental joint arm is itself a continual learner with a perfect approximation to the joint loss. De Lange et al. (2023) showed that replay, distillation and regularisation methods suffer a temporary but substantial drop in accuracy on earlier tasks at each task boundary, the stability gap, and Hess et al. (2023) showed that the drop persists even when the objective is a perfect approximation to the joint loss. Mirzadeh et al. (2020) found that training choices such as dropout, learning-rate decay and batch size change how much is forgotten. The incremental joint arm's position relative to offline joint training therefore depends on its optimisation schedule, which the anchor holds fixed across arms by design and acknowledges as a limitation.

10. What happens at scale

The anchor hypothesises that co-observation matters most in continual pretraining, where the backbone is substantially updated, and points to language-model results as suggestive (Hess et al., 2026, Section 6.2). The large-scale record is informative, though not in a way that isolates co-observation.

In continual pretraining of language models, Ibrahim et al. (2024) started from models pretrained on 300 billion tokens of the Pile and continued on 300 billion tokens of SlimPajama or 200 billion of German Common Crawl, at 405 million and 10 billion parameters. With re-warming and re-decaying of the learning rate alone, the continued models reached average final validation losses of 1.99 (10 billion, English shift), 2.47 (405 million, English) and 2.34 (405 million, German), against 1.87, 2.35 and 1.75 for models trained from scratch on the union. Adding 5 percent replay for the English shift and 25 percent for German gave 1.89, 2.37 and 1.75, and average evaluation scores of 47.7, 35.1 and 32.5 against 48.0, 34.3 and 32.4 for the union baseline. (The differences, 0.12, 0.12 and 0.59 without replay and 0.02, 0.02 and 0.00 with it, are the author's arithmetic.) Gupta et al. (2023) had earlier studied warm-up strategies for continued pretraining of a 410-million-parameter model, starting from the hypothesis that the learning rate must be raised again to train efficiently on new data. The anchor reads the need for replay here as consistent with co-observation, and it is. It is equally consistent with forgetting, which is how Ibrahim et al. frame it, and the two readings make the same prediction in this design.

What the language-model results add is a calibration of size. A first chunk of 300 billion tokens, trained in roughly a single pass, plus a 5 percent replay stream, leaves almost nothing to recover. The anchor's first chunk is a quarter of CIFAR-100 or of ImageNet-100, trained for about 100 passes in the supervised CIFAR-100 case, and a 30 percent buffer leaves 40 to 86 percent of the residual at the fourth chunk. Both observations fit the sample-size and repetition account of Section 5. Neither discriminates it from co-observation.

In time-continual training of CLIP models, Garg et al. (2024) found that warm-starting from the last checkpoint and replaying all past data came within 1 percent of retraining from scratch on dynamic evaluations, at a quarter to two-fifths of the compute. On static tasks the gap to retraining stayed small at small scales but reached 4.7 percent at their large scale and 1.8 and 4 percent in two extra-large configurations. That replay-everything arm is the large-scale analogue of incremental joint training, and the gap to from-scratch training on static tasks is the analogue of the anchor's plasticity term, not its residual; Garg et al. add that training these models "with slightly extra compute bridges the gap" while remaining at least 2.7 times more computationally efficient.

In self-supervised pretraining with identically distributed chunks, the result of Hu et al. (2022) discussed in Section 5.2 is the most direct large-scale test of the chunking claim: naive sequential MoCo-v2 on quarter-ImageNet chunks ended about one point from joint training in transfer. Hu et al. also found that self-supervised models forget less than supervised ones and related this to their representations and to the sharpness of the minima they reach. The anchor reports the opposite pattern on ImageNet-100, with naive-to-joint gaps of 7.17 and 15.52 points for its self-supervised settings against 5.41 for supervised training; the difference in chunk size, evaluation and objective is large enough that the paper does not treat this as a contradiction, but it is a reason to doubt that the anchor's self-supervised gaps generalise upward.

Two further lines bear on scale without settling it. Generic pretraining makes sequential fine-tuning forget less than random initialisation does (Mehta et al., 2023). And when memory is abundant but retraining is too expensive, the main difficulty shifts from stability to plasticity, with simple replay outperforming more elaborate methods at a fraction of the cost (Cho et al., 2026). An earlier large-scale study argued that compute rather than storage is the binding constraint in practice and evaluated continual learning methods under compute budgets accordingly (Prabhu et al., 2023), and a survey of the field found memory-constrained settings dominant in published work while several open problems that need continual learning, faster retraining among them, are not memory-bound (Verwimp et al., 2024). None of these studies measures a residual of the anchor's kind. Whether one exists at scale is not known.

11. Consolidated evidence

Table 1 collects the measured values the argument rests on, one row per published result, with the account each bears on. Values are as reported by the cited source except where a row says "derived", which marks the author's arithmetic on the source's published numbers. Metrics differ across rows and should not be compared across sources.

Three patterns stand out in Table 1. First, the residual is clearly present in the anchor's design. Two studies vary chunk size within one design. In Lee and Storkey the gap between chunked and offline training shrinks as chunks grow. The anchor's extension from 4 to 8 and 12 chunks of ImageNet-100 reports that the gap between the ensemble and incremental joint training persists at the smaller chunk sizes (Hess et al., 2026, Appendix E), which the sample-size account also predicts; the anchor's tables cover only the four-chunk runs, and the extension does not go to larger chunks, so neither study shows whether the residual vanishes as chunks grow large. Hu et al. and Ibrahim et al. each supply one large-chunk point with a small gap. That is consistent with the sample-size account but does not by itself show the trend, and Ibrahim et al.'s gap is the same at 405M and 10B parameters. Second, the rows that show separate training failing badly (Zhao et al., Soutif-Cormerais et al., Busch et al.) all involve partitions that differ in distribution; the rows with identically distributed partitions show small or no losses when combination is frequent or regularisation is set for the whole sample. Third, no row varies the passes per example, the members' initialisation or the combiner while holding the others fixed, which is why the question remains open.

12. An attribution procedure

Before a gap between a perfect-retention baseline and joint training in some setting S is put down to the absence of joint exposure, it should survive the controls in Algorithm 1. They are listed from least to most expensive, and the inexpensive ones already address most of the alternatives raised in Sections 4 to 8.

Three points about the procedure. First, steps 2 to 4 cost little relative to the anchor's experiments and address the path dependence of Section 4 and the repetition confound of Section 5.4. A cold-start ensemble is the cleanest single control: if its residual equals the warm-start ensemble's, warm-starting is not the explanation; if it is smaller, part of the residual was warm-start damage.

Second, step 6 is the sharpest discriminating test the paper can identify. The co-observation account says that replay helps because replayed data come from other chunks and are seen alongside the current one. The sample-size account says that replay helps because it

Table 1. Published results on the gap between sequential and joint training, with the account each bears on. LP is linear-probe accuracy; IJ is incremental joint training; ens. is the forgetting-free ensemble.

Source	Setting	Sequential arm	Reference arm	Reported value	Bears on
Hess et al. 2026	SL, CIFAR-100, 4 chunks	ensemble	IJ	LP 70.39 vs 74.29	residual exists
Hess et al. 2026	SL, IN-100, 4 chunks	ensemble	IJ	LP 80.55 vs 82.75	residual exists
Hess et al. 2026	Barlow, IN-100	ensemble	IJ; offline joint	LP 71.25 vs 73.50; 76.42	plasticity term larger
Hess et al. 2026	I-JEPA, IN-100	naive; ensemble	IJ; offline joint	LP 53.83; 59.82 vs 62.64; 69.35	plasticity term larger
Hess et al. 2026	IN-100, 30% replay, chunk 4	replay	ens.-to-IJ gap	33% / 60% / 14% closed (derived)	"half the gap" in part
Hess et al. 2026	I-JEPA, 15% replay	replay	ensemble	LP 56.59 vs 59.82	replay below ensemble
Hess et al. 2026	CIFAR-100 budget	naive stage	IJ stage 4	∼100 vs ∼25 passes per example (derived)	repetition differs
Lee and Storkey 2024	DER++, CIFAR-100	chunked, no shift	offline	63.35 vs 73.72	chunking costs ∼half of CL drop
Lee and Storkey 2024	CIFAR-100, chunk size	1,000 per chunk	one chunk	∼45 vs ∼73	gap grows as chunks shrink
Hu et al. 2022	MoCo-v2, ImageNet, 4 IID chunks	naive sequential	joint	1.04 pt gap, 12 datasets	small gap, large chunks
Hu et al. 2022	supervised, same chunks	naive sequential	joint	4.83 pt gap	objective matters
Ash and Adams 2020	ResNet-18, CIFAR-10, SGD	warm start from 50%	random init	51.7 vs 56.2	warm-start loss, no shift
Ash and Adams 2020	logistic regression	warm start	random init	39.6 vs 40.5	convex case barely affected
Ibrahim et al. 2024	10B LM, Pile to SlimPajama	LR re-warm, no replay	union	loss 1.99 vs 1.87	replay needed
Ibrahim et al. 2024	same, 5% replay	LR re-warm + replay	union	loss 1.89 vs 1.87	small replay suffices at scale
Ibrahim et al. 2024	405M, to German, 25% replay	LR re-warm + replay	union	loss 1.75 vs 1.75	gap closes
Garg et al. 2024	CLIP, replay all past data	warm start	from scratch	within 1% dynamic; 4.7% static	IJ-type gap at scale
Zhao et al. 2018	federated, 1 class per client	local + averaging	centralised	up to 55% lower	separate training costs when parts differ
Zhang et al. 2013	kernel ridge, random split	averaged local fits	full fit	minimax rate kept	combiner and regularisation
Muennighoff et al. 2023	LM, fixed compute	up to 4 epochs	unique data	negligible loss change	repetition threshold
Hernandez et al. 2022	800M LM	0.1% repeated 100x	unique data	∼400M-model level	repetition harms
Soutif-Cormerais et al. 2021	class-incremental	replay variants	cross-task ablation	forgetting not main cause	cross-partition features
Busch et al. 2025	confounded CLEVR tasks	CL methods	joint	sequential fails to ignore confounders	cross-partition structure
Wu et al. 2022	augmentation-invariance tasks	continual	multitask	continual often better	joint not always upper line

enlarges the stage's dataset and lowers repetition. Giving the learner the same number of fresh, never-seen examples from the same distribution separates the two: under the first account fresh data should help less than replay, since it contains no earlier chunk; under the second it should help as much or more, since it adds new information as well as size. In an identically distributed stream, fresh data are available by construction.

Third, steps 10 to 13 are ordered so that a residual is labelled co-observation only after the more ordinary explanations have been ruled out, and the label carries the setting and chunk size it was measured at. Applied to the anchor, the procedure stops at step 1 for seeds and intervals (three seeds, six for the main I-JEPA arms, and standard errors only), and steps 2 to 8 have not been run. The anchor's PCA control is a version of step 7 for width only; its distillation ablation varies the members' retention but none of the variables in steps 2 to 8.

13. The strongest case against this reading

"On identically distributed chunks, 'more data per fit' is what co-observation means. The paper has renamed the effect, not explained it away." This is the most serious objection, and it is partly right. In the operational sense, the anchor's residual is a real cost of seeing the data in pieces, and for a practitioner it does not matter what it is called. The difference is in what follows. If the residual is per-fit sample size, it should vanish as chunks grow, it should not need data from other chunks to be repaired, and continual learning methods should target it by enlarging or re-using each stage's data. If it is a cross-chunk dependency, it should

Algorithm 1. Attributing a sequential-to-joint gap to co-observation

```
Input: chunks D_1..D_T of one dataset; architecture f; total budget B;
       arms: naive N, warm ensemble E_w, incremental joint IJ, joint J
1  report all arms at every stage with >= 5 seeds and intervals over seeds
2  add cold-start ensemble E_c (member t trained from scratch on D_t)
       and cold-start incremental model IJ_c (from scratch on D_1..D_t)
3  record passes per example per stage for every arm
4  rerun E_w and IJ with passes per example matched (fixed epochs, not
       fixed iterations); report the residual R = acc(IJ) - acc(E_w) both ways
5  sweep chunk size |D_t| over >= 3 values at fixed total data; fit R(|D_t|)
6  fresh-data control: give the naive learner, at stage t, m fresh examples
       from the same distribution instead of m replayed examples
7  vary the combiner at fixed members: concatenate+project, prediction average,
       weight average (if shared init), distillation into one student
8  vary interleaving: split each stage into blocks of k steps per chunk with
       stored data, k from one batch to one stage, at fixed data and budget
9  compute Shapley shares over the orders of {retention, warm start,
       joint exposure} using arms {N, E_w, E_c, IJ, IJ_c, J}
10 if R(|D_t|) -> 0 as |D_t| grows, or fresh data matches replay (step 6):
       label "per-fit sample size", scoped to (f, S)
11 elif R shrinks to its interval under some combiner (step 7):
       label "combination deficit"
12 elif R(k) falls smoothly with k and vanishes at short k (step 8):
       label "interleaving time scale"
13 else: label "co-observation residual not explained by sample size,
       repetition, warm start or combiner", scoped to (f, S, |D_t|)
14 separately: if chunks differ in distribution, test a cross-chunk
       evaluation task as in the anchor's MNIST construction
```

persist at large chunk sizes and should need joint exposure. The two readings make different predictions (Section 5.5), and the anchor's design cannot tell them apart. Naming the effect a third facet, distinct from forgetting and plasticity, commits to the second reading before the test.

"**The MNIST experiment shows a genuine cross-chunk effect, so the mechanism exists.**" It does, and Section 9 places it with the class-incremental and confounding results where the mechanism is well supported. The anchor's claim is broader: that the effect is generic and does not require any particular distribution shift (Hess et al., 2026). The MNIST case has a shift built in by design. The chunking experiments are the evidence for genericity, and they are the experiments in which the alternatives of Sections 5 to 8 are not controlled.

"**The training budget was held fixed, so replay's advantage cannot be more optimisation.**" The total number of iterations was held fixed. The number of passes per example was not, and it falls as the buffer grows (Section 5.4). Holding iterations fixed controls compute; it does not control repetition.

"**The PCA control shows that the ensemble's shortfall is not about capacity, and the distillation ablation shows it is not about transfer.**" Both controls are real and the paper credits them. The PCA control shows that a wider concatenation does not explain the residual; it does not show that concatenation is the right way to combine separately trained networks (Section 6). The distillation ablation shows that members with better retention do not form a much better ensemble; it does not change the members' initialisation, sample size or repetition (Section 4).

"**Lee and Storkey attribute the chunking drop to forgetting, and the ensemble removes forgetting. The residual is exactly what their account cannot explain.**" This is a fair reading of the anchor's contribution relative to Lee and Storkey: the ensemble shows that something survives perfect retention of features. But Lee and Storkey's forgetting is measured as a fall in accuracy on a chunk's training data to test level, and their candidate cause is overfitting to the current chunk. An ensemble keeps each member's fit to its own chunk, overfitted or not. Perfect retention of an overfitted representation is not a remedy for overfitting, so a residual that survives it is consistent with their account as well as with co-observation.

"**Self-supervised objectives are thought to be robust to forgetting, yet the anchor finds large gaps for them, which strengthens the case for a non-forgetting factor.**" The anchor's self-supervised gaps are large, but most of each is not the residual: in both self-supervised settings the plasticity term is the largest of the three (Figure 1). And in the closest large-chunk comparison, naive sequential self-supervised pretraining came within about a point of joint training (Hu et al., 2022). If anything, the self-supervised results point to warm-start and sample-size effects more than to a third factor.

14. Studies that would settle the open part

Five studies, each feasible at the anchor's scale with its released code, would close most of what remains.

The fresh-data control. In the anchor's supervised ImageNet-100 setting, replace the replay buffer at each stage by the same number of fresh, held-back examples from the same distribution, at matched iterations. If

fresh data recover as much of the residual as replay, the residual is a sample-size effect; if replay recovers more, cross-chunk exposure is doing work that sample size does not. This is step 6 of Algorithm 1 and requires holding out part of the training set before chunking.

Cold-start arms and Shapley shares. Add a cold-start ensemble and a cold-start incremental model to the anchor's four arms, and report the residual and the Shapley shares of retention, warm start and joint exposure (step 9). This would show how much of the residual is the members' warm-start damage, which the warm-start literature predicts is not small (Ash and Adams, 2020; Shin et al., 2024).

A chunk-size and repetition grid. Cross chunk size (for instance 4, 8 and 16 chunks of a fixed dataset, and the same number of chunks of a dataset four times larger) with passes per example (fixed epochs versus fixed iterations), and fit the residual as a function of both. The anchor's extension to 8 and 12 chunks scaled iterations to keep the total budget constant (Hess et al., 2026, Appendix E). Since chunk size and per-chunk iterations both fall in proportion, each naive or ensemble member still passes over its chunk about 90 times while the chunk shrinks, but the incremental joint arm's passes per example fall, so the comparison between the arms still mixes chunk size with repetition (author's arithmetic, as in Section 3.1).

Combiner variants at fixed members. Combine the same four members by concatenation and projection, by prediction averaging, and by distillation into one student trained only on the last chunk's inputs, and train a second set of members with early stopping or weaker regularisation for combination. If any combiner or member regime brings the ensemble to incremental joint training, the residual is a combination deficit (Section 6).

An interleaving sweep. With all data stored, train on the four chunks in blocks of k consecutive steps per chunk, from one mini-batch to one full stage, at fixed total budget and fixed data. The residual as a function of k would show whether co-observation has a time scale, and where it sits relative to the learning-rate schedule (Section 8).

15. What is not known

It is not known whether the residual between a perfect-retention ensemble and incremental joint training persists when each chunk is large relative to what the model needs. Every measurement of it uses chunks of one quarter (or, in an extension, one eighth or one twelfth) of CIFAR-100 or ImageNet-100.

It is not known how much of the residual is warm-start damage in the ensemble's members. No study has run a cold-start ensemble in this design.

It is not known whether fresh same-distribution data repair the residual as well as replayed data do. This is the test that would separate sample size from cross-chunk exposure, and it has not been run.

It is not known whether a different combination of the same members would close the residual. Only concatenation, with and without projection, has been tested.

It is not known whether co-observation, in the mechanistic sense, operates on identically distributed data at all. The cases where joint exposure demonstrably supplies information that no chunk contains all involve partitions that differ.

16. Limitations

The design limitations of the cited experiments carry over to this analysis. The anchor is a recent preprint, read in its first version; a later version may add arms or intervals that address some of the points above. Several sources are preprints, including the anchor, the abundant-memory study and several of the theoretical papers, and the paper uses them for the existence and direction of effects rather than for exact sizes. The decomposition in Figure 1 and the replay shares are arithmetic on the anchor's means and do not carry its standard errors forward; with three seeds and standard errors of up to about one point in some I-JEPA cells, the smaller shares are imprecise. Cross-study comparisons in Sections 5 and 10 cross architectures, objectives, datasets and evaluation protocols, and are used only for direction. The theoretical results invoked in Sections 6 and 7 concern kernel ridge regression, linear models and idealised learners, and the paper uses them to show that a residual is not forced by sequential exposure in general, not to predict its size for deep networks. Finally, the paper does not evaluate the practical value of the anchor's framing, which may be considerable: a name for the gap that perfect retention leaves can direct method design even before its cause is settled.

17. Conclusion

The 2026 study measured something real. In four settings, a forgetting-free ensemble of sequential checkpoints falls short of incrementally joint training by 2 to 4 linear-probe points at the fourth chunk, and that shortfall is not explained by forgetting or by the plasticity loss that the study measures. The study names it co-observation and calls it a third facet of continual learning.

The name runs ahead of the evidence in three ways. On identically distributed chunks, observing the chunks together is the same thing as fitting each model on more data, so the residual cannot be separated from per-fit sample size by the design used, and the passes per example differ fourfold between the arms being compared. The ensemble's members carry the warm-start damage that stationary-data experiments have documented for years, and the order in which the arms

are subtracted counts it in the residual. And the residual compares a particular way of combining separate fits with a single joint fit, where statistical theory shows that separate fitting loses nothing in rate when local fits are regularised for the combination. The sequential-to-joint gap shrinks as chunks grow in the one study that varies chunk size upward toward offline training (Lee and Storkey, 2024), and the anchor's own finer partitions leave the residual in place. Outside the anchor, the gap is about a point for self-supervised pretraining on large identically distributed chunks and closes almost completely in language-model pretraining with a small replay stream.

Joint exposure does matter where partitions carry different information, as class-incremental and confounded settings show. Whether it matters as a separate factor when they do not is an open question, and the experiments that would answer it are inexpensive: a fresh-data control, a cold-start ensemble, a chunk-size grid at matched repetition, alternative combiners and an interleaving sweep. Until they are run, the gap between perfect retention and joint training is best reported as what it is, a residual with several candidate causes, and Algorithm 1 lists the order in which to test them.

References

Achille, A., Rovere, M., & Soatto, S. (2019). Critical Learning Periods in Deep Neural Networks. International Conference on Learning Representations (ICLR 2019). arXiv:1711.08856. doi:10.48550/arXiv.1711.08856.

Allen-Zhu, Z., & Li, Y. (2023). Towards Understanding Ensemble, Knowledge Distillation and Self-Distillation in Deep Learning. International Conference on Learning Representations (ICLR 2023). arXiv:2012.09816. doi:10.48550/arXiv.2012.09816.

Ash, J. T., & Adams, R. P. (2020). On Warm-Starting Neural Network Training. Advances in Neural Information Processing Systems 33 (NeurIPS 2020). arXiv:1910.08475. doi:10.48550/arXiv.1910.08475.

Baxter, J. (2000). A Model of Inductive Bias Learning. Journal of Artificial Intelligence Research, 12, 149-198. doi:10.1613/jair.731.

Berariu, T., Czarnecki, W., De, S., Bornschein, J., Smith, S., Pascanu, R., & Clopath, C. (2021). A study on the plasticity of neural networks. arXiv:2106.00042. doi:10.48550/arXiv.2106.00042.

Busch, F. P., Kamath, R., Mitchell, R., Stammer, W., Kersting, K., & Mundt, M. (2025). Where is the Truth? The Risk of Getting Confounded in a Continual World. Proceedings of the 42nd International Conference on Machine Learning (ICML 2025). arXiv:2402.06434. doi:10.48550/arXiv.2402.06434.

Caruana, R. (1997). "Multitask Learning." Machine Learning, 28(1), 41-75. doi:10.1023/A:1007379606734.

Carvalho, P. F., & Goldstone, R. L. (2014). Putting category learning in order: Category structure and temporal arrangement affect the benefit of interleaved over blocked study. Memory & Cognition, 42(3), 481-495. doi:10.3758/s13421-013-0371-0.

Chaudhry, A., Rohrbach, M., Elhoseiny, M., Ajanthan, T., Dokania, P. K., Torr, P. H. S., & Ranzato, M. (2019). On Tiny Episodic Memories in Continual Learning. arXiv:1902.10486. doi:10.48550/arXiv.1902.10486.

Cho, D., Moon, T., Chunara, R., Cho, K., & Cha, S. (2026). Forget Forgetting: Continual Learning in a World of Abundant Memory. International Conference on Learning Representations (ICLR 2026). arXiv:2502.07274. doi:10.48550/arXiv.2502.07274.

Davari, M., Asadi, N., Mudur, S., Aljundi, R., & Belilovsky, E. (2022). Probing Representation Forgetting in Supervised and Unsupervised Continual Learning. Proceedings of CVPR 2022. arXiv:2203.13381. doi:10.48550/arXiv.2203.13381.

De Lange, M., Aljundi, R., Masana, M., Parisot, S., Jia, X., Leonardis, A., Slabaugh, G., & Tuytelaars, T. (2021). A continual learning survey: Defying forgetting in classification tasks. IEEE Transactions on Pattern Analysis and Machine Intelligence. arXiv:1909.08383. doi:10.48550/arXiv.1909.08383.

De Lange, M., van de Ven, G., & Tuytelaars, T. (2023). Continual evaluation for lifelong learning: Identifying the stability gap. International Conference on Learning Representations (ICLR 2023). arXiv:2205.13452. doi:10.48550/arXiv.2205.13452.

Dohare, S., Hernandez-Garcia, J. F., Lan, Q., Rahman, P., Mahmood, A. R., & Sutton, R. S. (2024). Loss of plasticity in deep continual learning. Nature, 632(8026), 768-774. doi:10.1038/s41586-024-07711-7.

Evron, I., Moroshko, E., Ward, R., Srebro, N., & Soudry, D. (2022). How catastrophic can catastrophic forgetting be in linear regression? Proceedings of the 35th Conference on Learning Theory (COLT 2022). arXiv:2205.09588. doi:10.48550/arXiv.2205.09588.

Fini, E., Turrisi da Costa, V. G., Alameda-Pineda, X., Ricci, E., Alahari, K., & Mairal, J. (2022). Self-Supervised Models are Continual Learners. Proceedings of CVPR 2022. arXiv:2112.04215. doi:10.48550/arXiv.2112.04215.

Flesch, T., Balaguer, J., Dekker, R., Nili, H., & Summerfield, C. (2018). Comparing continual task learning in minds and machines. Proceedings of the National Academy of Sciences, 115(44). doi:10.1073/pnas.1800755115.

Fortin, N., Lemieux, T., & Firpo, S. (2011). Decomposition Methods in Economics. Handbook of Labor Economics, Volume 4A, 1-102. doi:10.1016/S0169-7218(11)00407-2.

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

Garg, S., Farajtabar, M., Pouransari, H., Vemulapalli, R., Mehta, S., Tuzel, O., Shankar, V., & Faghri, F. (2024). TiC-CLIP: Continual Training of CLIP Models. International Conference on Learning Representations (ICLR 2024). arXiv:2310.16226. doi:10.48550/arXiv.2310.16226.

Gupta, K., Therien, B., Ibrahim, A., Richter, M. L., Anthony, Q., Belilovsky, E., Rish, I., & Lesort, T. (2023). Continual Pre-Training of Large Language Models: How to (re)warm your model? arXiv:2308.04014. doi:10.48550/arXiv.2308.04014.

Gururangan, S., Li, M., Lewis, M., Shi, W., Althoff, T., Smith, N. A., & Zettlemoyer, L. (2023). Scaling Expert Language Models with Unsupervised Domain Discovery. arXiv:2303.14177. doi:10.48550/arXiv.2303.14177.

Hernandez, D., Brown, T., Conerly, T., DasSarma, N., Drain, D., El-Showk, S., Elhage, N., Hatfield-Dodds, Z., et al. (2022). Scaling Laws and Interpretability of Learning from Repeated Data. arXiv:2205.10487. doi:10.48550/arXiv.2205.10487.

Hess, T., Jha, A., van de Ven, G. M., & Tuytelaars, T. (2026). Forgetting, plasticity, and co-observation: a third facet of continual learning. arXiv:2608.18803. doi:10.48550/arXiv.2608.18803. [PREPRINT]

Hess, T., Tuytelaars, T., & van de Ven, G. M. (2023). Two Complementary Perspectives to Continual Learning: Ask Not Only What to Optimize, But Also How. Proceedings of the 1st ContinualAI Unconference, PMLR 249, 37-61. arXiv:2311.04898. doi:10.48550/arXiv.2311.04898.

Hess, T., Verwimp, E., van de Ven, G. M., & Tuytelaars, T. (2024). Knowledge Accumulation in Continually Learned Representations and the Issue of Feature Forgetting. Transactions on Machine Learning Research. arXiv:2304.00933. doi:10.48550/arXiv.2304.00933.

Hu, D., Yan, S., Lu, Q., Hong, L., Hu, H., Zhang, Y., Li, Z., Wang, X., & Feng, J. (2022). How Well Does Self-Supervised Pre-Training Perform with Streaming Data? International Conference on Learning Representations (ICLR 2022). arXiv:2104.12081. doi:10.48550/arXiv.2104.12081.

Huszar, F. (2018). Note on the quadratic penalties in elastic weight consolidation. Proceedings of the National Academy of Sciences, 115(11). doi:10.1073/pnas.1717042115.

Ibrahim, A., Therien, B., Gupta, K., Richter, M. L., Anthony, Q., Lesort, T., Belilovsky, E., & Rish, I. (2024). Simple and Scalable Strategies to Continually Pre-train Large Language Models. Transactions on Machine Learning Research. arXiv:2403.08763. doi:10.48550/arXiv.2403.08763.

Igl, M., Farquhar, G., Luketina, J., Boehmer, W., & Whiteson, S. (2021). Transient Non-Stationarity and Generalisation in Deep Reinforcement Learning. International Conference on Learning Representations (ICLR 2021). arXiv:2006.05826. doi:10.48550/arXiv.2006.05826.

Kirichenko, P., Izmailov, P., & Wilson, A. G. (2023). Last Layer Re-Training is Sufficient for Robustness to Spurious Correlations. International Conference on Learning Representations (ICLR 2023). arXiv:2204.02937. doi:10.48550/arXiv.2204.02937.

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., et al. (2017). Overcoming catastrophic forgetting in neural networks. Proceedings of the National Academy of Sciences, 114(13), 3521-3526. doi:10.1073/pnas.1611835114.

Knoblauch, J., Husain, H., & Diethe, T. (2020). Optimal Continual Learning has Perfect Memory and is NP-hard. Proceedings of the 37th International Conference on Machine Learning (ICML 2020). arXiv:2006.05188. doi:10.48550/arXiv.2006.05188.

Kornell, N., & Bjork, R. A. (2008). Learning Concepts and Categories. Psychological Science, 19(6), 585-592. doi:10.1111/j.1467-9280.2008.02127.x.

Lee, H., Cho, H., Kim, H., Kim, D., Min, D., Choo, J., & Lyle, C. (2024). Slow and Steady Wins the Race: Maintaining Plasticity with Hare and Tortoise Networks. Proceedings of the 41st International Conference on Machine Learning (ICML 2024). arXiv:2406.02596. doi:10.48550/arXiv.2406.02596.

Lee, T. L., & Storkey, A. (2024). Chunking: Continual Learning is not just about Distribution Shift. Conference on Lifelong Learning Agents (CoLLAs 2024). arXiv:2310.02206. doi:10.48550/arXiv.2310.02206.

Lesort, T., Ostapenko, O., Misra, D., Arefin, M. R., Rodriguez, P., Charlin, L., & Rish, I. (2022). Challenging Common Assumptions about Catastrophic Forgetting. arXiv:2207.04543. doi:10.48550/arXiv.2207.04543.

Li, M., Gururangan, S., Dettmers, T., Lewis, M., Althoff, T., Smith, N. A., & Zettlemoyer, L. (2022). Branch-Train-Merge: Embarrassingly Parallel Training of Expert Language Models. arXiv:2208.03306. doi:10.48550/arXiv.2208.03306.

Li, Z., & Hoiem, D. (2017). Learning without Forgetting. IEEE Transactions on Pattern Analysis and Machine Intelligence (conference version ECCV 2016). arXiv:1606.09282. doi:10.48550/arXiv.1606.09282.

Lin, S., Ju, P., Liang, Y., & Shroff, N. (2023). Theory on Forgetting and Generalization of Continual Learning. arXiv:2302.05836. doi:10.48550/arXiv.2302.05836.

Lin, T., Stich, S. U., Patel, K. K., & Jaggi, M. (2020). Don't Use Large Mini-Batches, Use Local SGD. International Conference on Learning Representations (ICLR 2020). arXiv:1808.07217. doi:10.48550/arXiv.1808.07217.

Lyle, C., Zheng, Z., Nikishin, E., Avila Pires, B., Pascanu, R., & Dabney, W. (2023). Understanding plasticity in neural networks. Proceedings of the 40th International Conference on Machine Learning (ICML 2023). arXiv:2303.01486. doi:10.48550/arXiv.2303.01486.

Lyle, C., Zheng, Z., Khetarpal, K., van Hasselt, H., Pascanu, R., Martens, J., & Dabney, W. (2024). Disentangling the Causes of Plasticity Loss in Neural Networks. arXiv:2402.18762. doi:10.48550/arXiv.2402.18762.

McClelland, J. L., McNaughton, B. L., & O'Reilly, R. C. (1995). 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. doi:10.1037/0033-295X.102.3.419.

McCloskey, M., & Cohen, N. J. (1989). Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. Psychology of Learning and Motivation, 24, 109-165. doi:10.1016/S0079-7421(08)60536-8.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., & Aguera y Arcas, B. (2017). Communication-Efficient Learning of Deep Networks from Decentralized Data. Proceedings of AISTATS 2017. arXiv:1602.05629. doi:10.48550/arXiv.1602.05629.

Mehta, S. V., Patil, D., Chandar, S., & Strubell, E. (2023). An Empirical Investigation of the Role of Pre-training in Lifelong Learning. Journal of Machine Learning Research, 24, 1-50. arXiv:2112.09153. doi:10.48550/arXiv.2112.09153.

Mirzadeh, S. I., Farajtabar, M., Pascanu, R., & Ghasemzadeh, H. (2020). Understanding the Role of Training Regimes in Continual Learning. arXiv:2006.06958. doi:10.48550/arXiv.2006.06958.

Muennighoff, N., Rush, A. M., Barak, B., Le Scao, T., Piktus, A., Tazi, N., Pyysalo, S., Wolf, T., & Raffel, C. (2023). Scaling Data-Constrained Language Models. arXiv:2305.16264. doi:10.48550/arXiv.2305.16264.

Nguyen, C. V., Li, Y., Bui, T. D., & Turner, R. E. (2018). Variational Continual Learning. International Conference on Learning Representations (ICLR 2018). arXiv:1710.10628. doi:10.48550/arXiv.1710.10628.

Nikishin, E., Schwarzer, M., D'Oro, P., Bacon, P.-L., & Courville, A. (2022). The Primacy Bias in Deep Reinforcement Learning. Proceedings of the 39th International Conference on Machine Learning (ICML 2022). arXiv:2205.07802. doi:10.48550/arXiv.2205.07802.

Peng, L., Giampouras, P. V., & Vidal, R. (2023). The Ideal Continual Learner: An Agent That Never Forgets. Proceedings of the 40th International Conference on Machine Learning (ICML 2023). arXiv:2305.00316. doi:10.48550/arXiv.2305.00316.

Prabhu, A., Hammoud, H. A. A. K., Dokania, P., Torr, P. H. S., Lim, S.-N., Ghanem, B., & Bibi, A. (2023). Computationally Budgeted Continual Learning: What Does Matter? Proceedings of CVPR 2023. arXiv:2303.11165. doi:10.48550/arXiv.2303.11165.

Shah, H., Tamuly, K., Raghunathan, A., Jain, P., & Netrapalli, P. (2020). The Pitfalls of Simplicity Bias in Neural Networks. Advances in Neural Information Processing Systems 33 (NeurIPS 2020). arXiv:2006.07710. doi:10.48550/arXiv.2006.07710.

Shin, B., Oh, J., Cho, H., & Yun, C. (2024). DASH: Warm-Starting Neural Network Training in Stationary Settings without Loss of Plasticity. Advances in Neural Information Processing Systems 37 (NeurIPS 2024). arXiv:2410.23495. doi:10.48550/arXiv.2410.23495.

Shorrocks, A. F. (2013). Decomposition procedures for distributional analysis: a unified framework based on the Shapley value. The Journal of Economic Inequality, 11(1), 99-126. doi:10.1007/s10888-011-9214-z.

Soutif-Cormerais, A., Masana, M., van de Weijer, J., & Twardowski, B. (2021). On the importance of cross-task features for class-incremental learning. Theory and Foundation of Continual Learning Workshop, ICML 2021. arXiv:2106.11930. doi:10.48550/arXiv.2106.11930.

Standley, T., Zamir, A. R., Chen, D., Guibas, L., Malik, J., & Savarese, S. (2020). Which Tasks Should Be Learned Together in Multi-task Learning? Proceedings of the 37th International Conference on Machine Learning (ICML 2020). arXiv:1905.07553. doi:10.48550/arXiv.1905.07553.

Stich, S. U. (2019). Local SGD Converges Fast and Communicates Little. International Conference on Learning Representations (ICLR 2019). arXiv:1805.09767. doi:10.48550/arXiv.1805.09767.

van de Ven, G. M., Tuytelaars, T., & Tolias, A. S. (2022). Three types of incremental learning. Nature Machine Intelligence, 4(12), 1185-1197. doi:10.1038/s42256-022-00568-3.

Verwimp, E., De Lange, M., & Tuytelaars, T. (2021). Rehearsal revealed: The limits and merits of revisiting samples in continual learning. Proceedings of ICCV 2021, 9385-9394. arXiv:2104.07446. doi:10.48550/arXiv.2104.07446.

Verwimp, E., Aljundi, R., Ben-David, S., Bethge, M., Cossu, A., Gepperth, A., Hayes, T. L., Hullermeier, E., et al. (2024). Continual Learning: Applications and the Road Forward. Transactions on Machine Learning Research. arXiv:2311.11908. doi:10.48550/arXiv.2311.11908.

Wang, L., Zhang, X., Su, H., & Zhu, J. (2024). A Comprehensive Survey of Continual Learning: Theory, Method and Application. IEEE Transactions on Pattern Analysis and Machine Intelligence. arXiv:2302.00487. doi:10.48550/arXiv.2302.00487.

Woodworth, B., Patel, K. K., Stich, S. U., Dai, Z., Bullins, B., McMahan, H. B., Shamir, O., & Srebro, N. (2020). Is Local SGD Better than Minibatch SGD? Proceedings of the 37th International Conference on Machine Learning (ICML 2020). arXiv:2002.07839. doi:10.48550/arXiv.2002.07839.

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., & Schmidt, L. (2022). Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. Proceedings of the 39th International Conference on Machine Learning (ICML 2022). arXiv:2203.05482. doi:10.48550/arXiv.2203.05482.

Wu, X., Dyer, E., & Neyshabur, B. (2021). When Do Curricula Work? International Conference on Learning Representations (ICLR 2021). arXiv:2012.03107. doi:10.48550/arXiv.2012.03107.

Wu, Z., Tran, H., Pirsiavash, H., & Kolouri, S. (2022). Is Multi-Task Learning an Upper Bound for Continual Learning? arXiv:2210.14797. doi:10.48550/arXiv.2210.14797.

Yadav, P., Tam, D., Choshen, L., Raffel, C., & Bansal, M. (2023). TIES-Merging: Resolving Interference When Merging Models. Advances in Neural Information Processing Systems 36 (NeurIPS 2023). arXiv:2306.01708. doi:10.48550/arXiv.2306.01708.

Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K., & Finn, C. (2020). Gradient Surgery for Multi-Task Learning. Advances in Neural Information Processing Systems 33 (NeurIPS 2020). arXiv:2001.06782. doi:10.48550/arXiv.2001.06782.

Zaidi, S., Berariu, T., Kim, H., Bornschein, J., Clopath, C., Teh, Y. W., & Pascanu, R. (2023). When Does Re-initialization Work? Proceedings of the I Can't Believe It's Not Better Workshop at NeurIPS 2022, PMLR 187. arXiv:2206.10011. doi:10.48550/arXiv.2206.10011.

Zhang, Y., Duchi, J. C., & Wainwright, M. J. (2013). Divide and Conquer Kernel Ridge Regression: A Distributed Algorithm with Minimax Optimal Rates. arXiv:1305.5029. doi:10.48550/arXiv.1305.5029.

Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., & Chandra, V. (2018). Federated Learning with Non-IID Data. arXiv:1806.00582. doi:10.48550/arXiv.1806.00582.

Zhou, H., Vani, A., Larochelle, H., & Courville, A. (2022). Fortuitous Forgetting in Connectionist Networks. International Conference on Learning Representations (ICLR 2022). arXiv:2202.00155. doi:10.48550/arXiv.2202.00155.

© 2026 Pranay Mahendrakar. Licensed under CC BY 4.0.

DOI: 10.5281/zenodo.23933365