Measuring Class Expansion Capacity in Frozen Representations

Anonymous authors
Paper under double-blind review

Abstract

How many new classes can a frozen representation support without updating its feature extractor? We introduce frozen-representation class expansion — an extractor is trained on B base classes, frozen permanently, and then used to recognise further classes from newly stored statistics alone — and define expansion capacity as the largest label space served at a stated accuracy and memory budget. Measured on the arriving classes alone, capacity is much lower than on the pooled label space (65.8 ± 6.8 vs. 88.2 ± 3.1 classes at 50% accuracy), and it grows with the number of base classes even as closed-set accuracy falls. Second-order estimation, not prototype multiplicity, is the dominant lever (+16.4 points against at most +1.1). This motivates a readout combining shared whitening with a low-rank class-specific residual, which holds the accuracy–storage frontier at its own budget in nine of ten representation–dataset settings and on a 300-class stress test, and matches or beats full per-class covariance at 2.6–5.3% of its memory on every pretrained extractor, including at native resolution on Food-101. Whitening instead by the base features' uncentred second moment adds 1.7 to 4.3 points on backbones trained on the base classes, partly through the directions they learned to separate them, a gain whose sign a measured specialisation index predicts in 14 of 15 settings; it does not help pretrained ViTs. The prototype and depth findings are largely specific to backbones trained from scratch on few classes, at 32 pixels and at native resolution alike.

1 Introduction

Growing label spaces are the normal case in deployed recognition: an identity gallery, a product vocabulary or a defect taxonomy acquires classes after the model ships, and retraining the extractor each time is often impractical.

Class-incremental learning adds classes without destroying what a model knows (McCloskey & Cohen, 1989; Rebuffi et al., 2017). Almost all of that work allows the extractor to change, entangling representation drift with classifier reorganisation. We remove the first: the extractor is trained once on B classes and frozen permanently, and only memory is added.

Freezing is not proposed as a better way to learn continually. It turns a design choice into a measurement, since whatever accuracy the system reaches outside its original label set was already latent in a representation that cannot move.

Measure. We define expansion capacity $\mathrm{EC}_\tau(f, M)$: the largest label space a frozen extractor f still serves at accuracy τ under memory M. It orders memories almost exactly as average incremental accuracy does (Appendix S34); what it adds is a unit, classes, and a decomposition. Pooled capacity is subsidised by the well-served base classes, so we also report it on the arriving classes alone, and it rises with the number of base classes B while closed-set accuracy falls, so closed-set accuracy is a poor proxy for the room a representation has left (§5.2).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Explain. The bottleneck is not how many prototypes a class receives (+1.1 against +16.4 points, and not established under our criterion) but how faithfully class-conditional second-order structure can be estimated from finite samples (§5.4). A covariance pooled over the base classes is a statistically efficient prior, not evidence that the classes share a geometry: at a fixed sample count its benefit comes from averaging over classes, not from more images. The best prior also includes the base classes' between-class scatter, which discounts the directions the backbone learned to separate them and moves accuracy to the arriving classes — on backbones that separate their base classes far more than other classes, and not on pretrained ones.

Exploit. That diagnosis implies a memory: shared whitening estimated before expansion, plus a per-class low-rank residual covering only the directions the samples support (§4). Its claim is storage efficiency under a frozen readout, and its ingredients predate it — Mahalanobis readouts on frozen features (Goswami et al., 2023), shared second-order terms (McDonnell et al., 2023; Hayes & Kanan, 2020), low-rank covariance models (Tipping & Bishop, 1999); what we contribute is the measurement that selects among them and the parameterisation it points to. We also report where the findings stop: an intermediate layer beats the final one and prototype count does not matter mainly for backbones trained from scratch on few classes (§5.6), and base sets cannot yet be chosen for future capacity.

Contributions. (i) A protocol and a unit, expansion capacity, reported pooled and on the arriving classes alone (§3, §5.2). (ii) A diagnosis: capacity is limited by estimating second-order structure, not by prototype count; a pooled prior helps by averaging over classes, and the base classes' between-class scatter helps where the backbone separates them far more than other classes, as a measured index predicts in 14 of 15 settings (§5.4). (iii) A memory that follows from it, shared whitening plus a per-class low-rank residual, undominated on the mean in nine of ten representation–dataset settings at 2.6–7.4% of full covariance's storage (§4, Figure 1). (iv) Its scope: which findings stop on pretrained, higher-resolution extractors (§5.6, Appendix S43).

2 Related Work

Replay- and distillation-based methods update the backbone while retaining old knowledge (Rebuffi et al., 2017; Hou et al., 2019; Zhu et al., 2021; Asadi et al., 2023); Zhou et al. (2024b) survey the area and insist on the matched memory budgets we follow. Nearest-class-mean over a fixed embedding is old and strong (Mensink et al., 2013; Snell et al., 2017), recent work revives it with pretrained encoders (Janson et al., 2022; Zhou et al., 2024a), and few-shot class-incremental learning likewise freezes the backbone after its base session (Tao et al., 2020; Tian et al., 2024); we differ in that arriving classes carry their full training split rather than K shots, the label space grows to 100, 200 or 300 rather than to a fixed small target, and memory is a controlled axis rather than fixed. The closest methods freeze the backbone and store no exemplars: FeTrIL (Petit et al., 2023) trains a linear classifier on translated pseudo-features, FeCAM (Goswami et al., 2023) uses a per-class Mahalanobis distance (and evaluates one common covariance, which it rejects on accuracy), RanPAC (McDonnell et al., 2023) uses random projections with a shared Gram matrix — consistent with our finding that a shared second-order term carries much of the benefit — and streaming LDA (Hayes & Kanan, 2020) maintains one shared covariance online and scores with a linear discriminant; we run it as published (ϵ=10^{-4} shrinkage, covariance updated with every class), alongside our own shared term alone (Table S14).

Our parameterisation is assembled from known components: σI plus low rank is the form of probabilistic PCA (Tipping & Bishop, 1999), with σ added to the retained eigenvalues rather than subtracted; shrinking a per-class covariance toward a common one is regularised discriminant analysis (Friedman, 1989); a shared eigenbasis with class-specific eigenvalues is Flury's common-principal-component family (Flury, 1984); and borrowing task-level second-order statistics for data-poor classes appears in few-shot classification (Bateni et al., 2020). We add whitening by the shared covariance and spending the rank on what it leaves, worth

7.93 points over a per-class low-rank store like that of Honda (2026) at comparable memory (§5.4).

Modern Hopfield networks give a high-capacity retrieval rule (Ramsauer et al., 2021), which we test as one memory among several (§5.4). Intermediate features transfer better than final ones (Yosinski et al., 2014); closest to our depth result, Masarczyk et al. (2023) report that later layers of networks trained on few classes compress the representation and transfer worse, and Harun et al. (2024) find the compression fades as the number of training classes grows — the reversal §5.6 reports.

3 Frozen-Representation Class Expansion

Let $f_\theta : \mathcal{X} \to \mathbb{R}^d$ be trained to convergence on base classes $\mathcal{C}_0$, $|\mathcal{C}_0| = B$. We then fix $\theta = \theta_0$ permanently. Sessions $t = 1, \ldots, T$ deliver disjoint class sets $\mathcal{C}_t$; at session t the system may update a memory $\mathcal{M}_t$ using $f_{\theta_0}(x)$ for x in session t only, and is evaluated over $\bigcup_{s \le t} \mathcal{C}_s$. No images or features from earlier sessions are stored, and θ never changes. Every memory we propose obeys this, as do NCM, the prototype variants, FeCAM, FeTrIL, RanPAC, streaming LDA and the analytic readout, whose accumulated $d \times d$ statistics are not retained features. One diagnostic in §5.4 uses the features of classes held out of the label space; it tests a mechanism and is never a memory we compare.

Two baselines are exceptions, marked wherever they appear: the k-NN and joint linear probe revisit every feature each session, so we report them as upper bounds. An incremental probe, seeing only the current session, is the like-for-like comparison; trained identically (60 epochs, same 5 seeds) it reaches 0.3030 ± 0.0085 against the joint probe's 0.4294 ± 0.0091 and ours at 0.4760 ± 0.0058; Appendix S44 extends this to trained heads and fine-tuning. Appendix S20 records, for every method, what data it may revisit, what it stores and how it was tuned.

Definition 1 (Expansion capacity). For a frozen extractor f and a memory M, write $A_{f,M}(n)$ for accuracy when the label space has n classes. Then $\mathrm{EC}_\tau(f, M)$ is the label-space size at which accuracy first falls below τ, interpolated between measured sessions. Accuracy is non-increasing in n on every curve we measure, which is what makes the first crossing well defined and makes it coincide with $\max\{n : A_{f,M}(n) \ge \tau\}$ under the interpolation; we define it by the first crossing to match the released code. $A_{f,M}(n)$ is an expectation over arrival orders, estimated here by the seed average; a single ordering can differ, which Appendices S24 and S37 quantify. We abbreviate it EC_τ where f and M are fixed by context.

EC_τ is protocol-conditioned: a property of the representation, the memory, the arrival order, the threshold, the evaluation distribution and the benchmark ceiling, never of the backbone alone. Permuting arrivals at a fixed backbone moves EC_{50} by an sd of 2.8 against the 3.1 a seed shows, so much of a quoted spread may be the sequence rather than the representation; it cancels in paired comparisons (Appendix S37).

A union bound gives a sufficient condition on EC_τ for an idealised nearest-prototype readout; Appendix S35 tests it and its through-the-origin prediction fails.

4 A Shared-Prior Low-Rank Memory

We state the memory first for notation; the experiments that selected it (§5.4) show that what a frozen embedding needs from its memory is second-order structure, and that a covariance pooled over the base classes estimates the bulk of it far more reliably than any single class can. This suggests spending memory asymmetrically: one shared second-order term, estimated from B times as many samples, plus a small per-class correction covering only the directions each class can support.

Every memory L2-normalises the frozen features; the covariance family (ours and FeCAM's variants) also applies FeCAM's power transform $z \mapsto \mathrm{sign}(z)|z|^{1/2}$ (Goswami et al., 2023), which does not help the other baselines (Appendix S20). Let Σ_{sh} be the covariance pooled over the base classes, shrunk with FeCAM's fixed-intensity rule (Goswami et al., 2023),

whose first term is a fixed-λ form of Ledoit & Wolf (2004) and whose second shrinks toward a constant off-diagonal, then correlation-normalised, as FeCAM does (standardising the features first changes accuracy by $+0.06 \pm 0.10$ points, Appendix S20). Let W be the Cholesky factor with $WW^\top = \Sigma_{\text{sh}}^{-1}$; features are row vectors, so $\|(z - p_c)W\|^2$ is the shared Mahalanobis distance. It is estimated once, before expansion, and never updated — a conservative headline: Table 1 shows three stronger variants, a prior refreshed each session (Appendix S38), rank 64, and a better prior (§5.4). For each arriving class we whiten its centred features, $\tilde{X}_c = (Z_c - p_c)W$, and fit a rank-$r$ factorisation of the residual, $\tilde{\Sigma}_c \approx \sigma_c I + V_c^\top \Lambda_c V_c$ with $V_c \in \mathbb{R}^{r \times d}$ from the top-r SVD. Scoring uses the Woodbury form,

$$s_c(z) = -\frac{1}{\sigma_c}\left(\|\tilde{x}\|^2 - \sum_{k=1}^{r} \frac{\lambda_k}{\lambda_k + \sigma_c}\,(V_c\tilde{x})_k^2\right), \qquad \tilde{x} = (z - p_c)W. \tag{1}$$

Here p_c is the class mean, $\tilde{\Sigma}_c$ the covariance of the whitened features and $\Lambda_c = \operatorname{diag}(\lambda_1, \ldots, \lambda_r)$. The isotropic floor is $\sigma_c = \frac{1}{d-r}\sum_{k>r} \lambda_k + \gamma_1 \bar{\lambda}$: the variance the discarded directions leave, plus FeCAM's shrinkage constant γ_1=1 times the mean retained eigenvalue $\bar{\lambda}$, which damps each residual direction to a weight of at most $\lambda_k/(\lambda_k + \bar{\lambda})$ (ablated in Appendix S40). A test point is assigned $\arg\max_c s_c(z)$. We write the memory Shared+LRr throughout, and Appendix S11 states it as executable steps with every constant named. Shared+LRr uses FeCAM's prior; the uncentred variant (§5.4) is a stronger configuration of it. Memory is d^2 once plus $d + rd + r + 1$ per class — prototype, basis, eigenvalues, floor, or $(r{+}1)d$ to leading order — against d^2 per class for a full covariance; reported megabytes count those values exactly. At d=512 and r=32, 100 MB holds about 100 full covariances but the memory for 1,500 classes.

Why pooling helps. The following makes the estimation argument precise. Write $\Sigma_c = \Sigma_0 + \Delta_c$, where Σ_0 is common to the classes and Δ_c is class-specific with $\mathbb{E}_c[\Delta_c] = 0$.

Proposition 1 (estimation, not classification). Let the features be sub-Gaussian with $\|\Sigma_c\|_2 \le \sigma^2$. The per-class sample covariance from n samples obeys $\mathbb{E}\|\hat{\Sigma}_c - \Sigma_c\|_F^2 \le C\sigma^4 d^2/n$ for an absolute constant C, while the estimator pooled over B base classes has variance $\le C\sigma^4 d^2/(Bn)$ at the cost of a bias $\|\Delta_c\|_F^2$. The bound on the pooled error is smaller whenever $\|\Delta_c\|_F^2 < (1 - \frac{1}{B})\, C\sigma^4 d^2/n$; both sides being upper bounds, this orders the guarantees, not the errors, which §5.5 measures directly. The rank-r correction recovers the leading eigenspace of Δ_c at an accuracy governed by the eigengap: by Davis–Kahan (Davis & Kahan, 1970) the subspace error scales as $\sqrt{d/n}/\delta_r$ with δ_r the gap at the r-th eigenvalue, so the requirement is $n \gtrsim d/\delta_r^2$, not $n \gtrsim r$.

Only its first sentence is a theorem, and it bounds estimation error, not classification error; it fixes the regime rather than predicting a winner. With d=512 and n=500 the variance term is of the order of the covariance's own scale, where pooling wins, and smaller n should favour it further (§5.5). The assumption $\mathbb{E}_c[\Delta_c]$=0 fails on the arriving classes, but almost entirely in scale (2.31× more dispersed), which the memory divides out; the remaining 36% shape mismatch is what the rank-r term is for (Appendix S32). The proposition covers the pooled within-class estimator only; the between-class term of §5.4 adds bias on purpose and is identified empirically, not by it.

A necessary detail. Each $\tilde{\Sigma}_c$ must be normalised to unit mean variance ($\operatorname{tr}/d = 1$) before scoring, or the per-class scale differences make Mahalanobis distances incomparable, since we do not model the log-determinant: a per-class rank-8 covariance scores 0.3012 unnormalised against 0.3739 normalised, and restoring the log-determinant — scoring the full Gaussian discriminant — costs a further 29 to 33 points, because that term is dominated by the smallest and least reliable eigenvalues (Appendix S21). The failure is silent: the method simply looks bad.

5 Experiments

Setup. CIFAR-100 (Krizhevsky, 2009), ResNet-18 (He et al., 2016) adapted to 32×32, trained 100 epochs on the base classes and then frozen. Expansion $20 \to 30 \to 40 \to 50 \to$

$60 \to 80 \to 100$. The CIFAR base-20 headline comparison (Table 1, §5.2, Appendix S23) uses 5 seeds; most other experiments use 3, and each table says which. A seed changes the initialisation and which classes are the base set; the base set dominates (Appendix S22). An accuracy is over the whole label space unless the column names the arriving classes, which are macro-averaged. The six representations are ResNet-18, VGG-16 and ResNet-50 trained here, plus ResNet-50/ImageNet-1k, CLIP ViT-B/16 and DINOv2 ViT-B/14 (§5.6); with Food-101 they make ten settings. Features come from layer4 unless stated; memory is the float32 values stored, computed in MiB (2^{20} bytes) and written "MB" in tables. A $\pm$ on a table cell is the standard deviation over seeds; on a difference quoted in the text it is the half-width of the 95% paired t-interval over seeds. A difference with the same sign on all three seeds of a 3-seed experiment has a one-in-four chance under the null; we treat such a result as consistent with a claim, not as a test of it, and pool seeds across settings where a claim needs more (§5.4). The CIFAR-100 test split was read throughout development, so CIFAR effect sizes are post hoc; the prior and ridge are confirmed on held-out training data, and Food-101 was run once, with every memory and constant fixed beforehand. Every memory is fitted on the 500 training images of a class, in the session it arrives, and scored on the untouched test split with uniform priors; the benchmarks are class-balanced. Joint training on all 100 classes reaches 0.7782, and the 20-way closed-set head reaches 0.8753 against the frozen memory's 0.8782 (5-seed means), so freezing costs nothing on the base classes — the entire loss is expansion.

5.1 At a fixed backbone, the readout sets EC_τ

Table 1 reports the comparison. A 20-class representation supports the 100-class label space far above chance, and capacity depends strongly on the readout: EC_{50} runs from 48.2 ± 1.8 classes for a class mean to 88.2 ± 3.1 for our headline memory, and our strongest variant never falls below 0.50 at all. EC_τ inverts the accuracy curve, so it is close to an order-preserving reparametrisation: AUC-EC ranks the twenty memories at Spearman 0.977 against accuracy at K=100 and 0.992 against average incremental accuracy, the usual curve summary (Rebuffi et al., 2017), with four rows changing rank (Appendix S34). What it adds is a unit — classes, not points — and the base/arriving decomposition of §5.2. It also has a window: the best–worst spread is 50.1 classes at τ=0.50 but 5.5 at 0.70, so the readout binds only for $\tau \in [0.50, 0.65]$, above which the representation does. We therefore compare memories on AUC-EC, EC_τ averaged over that window (Appendix S30).

5.2 What arriving classes get, and how it scales with B

With $B = 20$, 80% of the label space is new but the base classes stay much better served, so EC_τ on pooled accuracy overstates what the representation does for classes it never saw. For the rank-32 memory $\mathrm{EC}_{50} = 88.2 \pm 3.1$ but $\mathrm{EC}_{50}^{\mathrm{new}} = 65.8 \pm 6.8$ on the arriving classes alone (Appendix S30). Base-class accuracy itself falls from 0.875 to 0.650 purely from added competitors, with representation and memory bit-identical — the inter-task confusion of Masana et al. (2023), isolated from representation drift by construction.

Varying only B, with the label space fixed at 100 (Appendix S6), each base class buys roughly 2.7 further hostable classes ($\mathrm{EC}_{50}^{\mathrm{new}}$: $24.3 \to 38.4 \to 65.5$ for $B = 5, 10, 20$; at B=50 both capacities saturate the benchmark), while closed-set accuracy moves the other way ($0.905 \to 0.873$). The closed-set task differs at each B, so this does not mean the representation worsens; it means capacity must be measured rather than inferred from closed-set skill. B is not a proxy for base-set geometry: no statistic of the frozen base classes improves on B alone, and base sets cannot yet be chosen for future capacity (Appendices S5 and S36).

5.3 The accuracy–storage frontier

On the frontier (Figure 1) no prior-work row dominates ours below 20 MB, where dominated means another row is both smaller and more accurate on the mean. Shared+LR32 beats FeCAM and RanPAC-2000 at 7.4% and 37% of their storage, by $+1.15 \pm 0.44$ and $+2.10 \pm 0.44$ points, both on every seed. On ResNet-18, RanPAC overtakes it at 107 MB but never

Table 1: Frozen expansion from 20 base classes to 100, mean±std over 5 seeds, ordered by accuracy. Columns: accuracy over all 100 classes and over the 80 arriving classes alone; AUC-EC (EC_τ averaged over $\tau \in [0.50, 0.65]$); EC_{50}; memory at 100 classes (timings in Table S29). Bold marks the best value in each performance column; storage is never bolded. FeCAM (Goswami et al., 2023), RanPAC (McDonnell et al., 2023), FeTrIL (Petit et al., 2023), streaming LDA (Hayes & Kanan, 2020) and the analytic readout (Zhuang et al., 2022) are prior work. Markers: †scalar settings chosen on the test split; *FeTrIL's published SVM head; §an upper bound that retains every feature; ‡our shared term alone; ¶variants stronger than our rank-32 headline. Each is explained in Appendix S20. ≥ marks an EC_{50} censored at the benchmark on every seed; RanPAC-10000's is censored on two of five.

Method	Acc@100	New-class acc	AUC-EC	EC_{50}	MB
Shared+LR32, uncentred prior (ours)¶	**0.5173 ± 0.0058**	**0.4855 ± 0.0039**	**74.3 ± 3.5**	**≥ 100**	7.46
RanPAC-10000	0.4973 ± 0.0052	0.4595 ± 0.0033	66.8 ± 3.7	98.3 ± 2.2	404.82
Shared+LR32, refreshed (ours)¶	0.4890 ± 0.0062	0.4492 ± 0.0048	64.7 ± 3.8	94.4 ± 3.0	7.46
Shared+LR64 (ours)¶	0.4890 ± 0.0052	0.4495 ± 0.0027	64.8 ± 3.5	94.4 ± 2.5	13.72
RanPAC-5000	0.4834 ± 0.0087	0.4382 ± 0.0069	65.3 ± 4.2	92.9 ± 3.3	107.04
Shared+LR32 (ours)	0.4760 ± 0.0058	0.4328 ± 0.0035	61.3 ± 3.3	88.2 ± 3.1	7.46
FeCAM	0.4644 ± 0.0042	0.4078 ± 0.0022	58.5 ± 3.2	82.2 ± 3.7	100.20
Shared+LR16 (ours)	0.4582 ± 0.0071	0.4095 ± 0.0043	57.3 ± 3.3	80.7 ± 3.9	4.33
RanPAC-2000	0.4550 ± 0.0078	0.3972 ± 0.0079	59.6 ± 3.4	82.2 ± 3.6	19.93
Shared+LR8 (ours)	0.4382 ± 0.0064	0.3850 ± 0.0045	53.5 ± 2.7	74.2 ± 3.0	2.76
Linear probe (joint)§	0.4294 ± 0.0091	0.3873 ± 0.0069	49.2 ± 3.1	68.3 ± 4.7	97.85
RanPAC-1000	0.4244 ± 0.0056	0.3528 ± 0.0041	54.0 ± 2.7	73.1 ± 3.3	6.15
Streaming LDA	0.4167 ± 0.0064	0.3655 ± 0.0035	49.2 ± 2.6	67.8 ± 3.9	1.20
Analytic readout (ACIL)	0.3942 ± 0.0056	0.2988 ± 0.0061	50.5 ± 2.4	65.7 ± 3.4	1.20
FeTrIL*	0.3701 ± 0.0135	0.3131 ± 0.0111	44.0 ± 0.9	53.4 ± 1.3	0.39
Shared only‡	0.3700 ± 0.0057	0.2916 ± 0.0029	44.2 ± 1.7	57.4 ± 3.0	1.20
k-NN (k=5)§	0.3616 ± 0.0132	0.2954 ± 0.0138	43.2 ± 2.4	55.6 ± 4.6	97.66
Associative prototypes†	0.3252 ± 0.0094	0.2292 ± 0.0099	40.6 ± 1.5	50.2 ± 2.6	0.59
Multi-prototype (K=4)	0.3238 ± 0.0135	0.2470 ± 0.0142	40.0 ± 1.7	49.6 ± 2.9	0.78
Nearest class mean	0.3125 ± 0.0083	0.2119 ± 0.0074	39.6 ± 1.3	48.2 ± 1.8	0.20

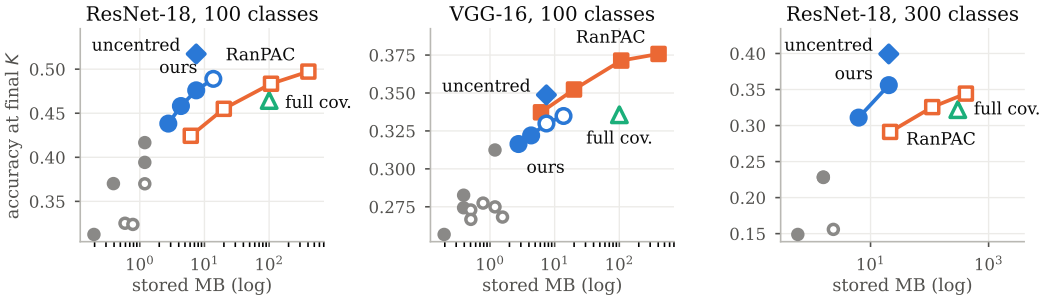


Figure 1: Accuracy against stored memory at the final label space. Ours: blue (ranks 8–64; 8 and 32 at 300 classes; diamond: rank 32 on the uncentred prior); RanPAC: orange (widths 1000–10000; from 2000 at 300 classes); full per-class covariance: green; all other baselines grey. Hollow markers are dominated by a smaller, more accurate row among those plotted. Numbers in Tables 1, S2 and S5.

overtakes our uncentred variant, which beats RanPAC-10000 by $+2.00 \pm 0.45$ at 1/54 of its storage. The one domination of our memory, by RanPAC-1000 on VGG-16, is $+0.74 \pm 1.12$ points and not significant (Appendix S41). Storage is the only axis we claim (costs in Appendix S33).

5.4 What unlocks capacity: estimating second-order structure

Second-order structure is the lever; prototype count is not. Shared+LR32 gains +16.4 points over a class mean, while $K = 2, 4, 8$ or adaptive prototypes per class gain 0.4–1.1 and associative aggregation (Ramsauer et al., 2021) 1.3, the K=4 variant not established under the class-level bootstrap and associative aggregation not surviving its multiplicity correction (Appendix S23). A single covariance estimated on the base classes before any expansion recovers 38% of the full-covariance gain at 1.2% of its memory, and spending the

Table 2: Decomposing the uncentred prior, rank 32, ridge 10^{-2}, seeds in parentheses. "FeCAM's" is the headline prior; "Within" is the within-class scatter with the ridge instead of FeCAM's shrinkage. The next three columns are paired differences from "Within", in points, with the 95% t half-width. The last column splits the between-class term's effect into base-class and arriving-class accuracy.

Representation	FeCAM's	Within	$+\mu\mu^\top$	+between	Uncentred	between: base / new
CIFAR-100 / ResNet-18 (5)	0.4760	0.4963	-0.02 ± 0.08	$+1.81 \pm 0.23$	$+2.10 \pm 0.30$	-0.92 ± 0.78 / $+2.50 \pm 0.20$
CIFAR-100 / VGG-16 (3)	0.3299	0.3351	-0.05 ± 0.41	$+1.36 \pm 0.48$	$+1.36 \pm 0.74$	-2.38 ± 0.83 / $+2.30 \pm 0.40$
CIFAR-100 / ResNet-50 (3)	0.5405	0.5513	-0.10 ± 0.05	$+1.16 \pm 0.10$	$+1.18 \pm 0.30$	-0.88 ± 1.90 / $+1.67 \pm 0.58$
Tiny-ImageNet / ResNet-18 (3)	0.3366	0.3546	$+0.14 \pm 0.62$	$+0.15 \pm 0.11$	$+0.32 \pm 0.46$	-3.67 ± 3.07 / $+0.58 \pm 0.26$
300-class union / ResNet-18 (3)	0.3561	0.3873	-0.28 ± 0.25	$+1.29 \pm 0.36$	$+1.20 \pm 0.52$	-3.12 ± 1.97 / $+1.51 \pm 0.48$
Pretrained donors (not trained on the base classes)						
ResNet-50 (ImageNet-1k) (3)	0.7506	0.7599	$+0.28 \pm 0.51$	-0.73 ± 0.05	-0.29 ± 0.24	-3.10 ± 0.78 / -0.14 ± 0.25
CLIP ViT-B/16 (3)	0.7836	0.7847	$+0.13 \pm 0.31$	-0.73 ± 0.23	-0.63 ± 0.09	-2.58 ± 0.47 / -0.26 ± 0.17
DINOv2 ViT-B/14 (3)	0.9020	0.8997	-0.00 ± 0.16	-0.42 ± 0.11	-0.43 ± 0.10	-1.62 ± 0.75 / -0.12 ± 0.26

rank on the residual after shared whitening beats spending it on per-class covariances by $+7.93 \pm 0.56$ points at comparable storage (7.46 against 6.46 MB). Shared+LR32 exceeds full FeCAM on every seed, by +0.70 to +1.65 points.

Alignment, or estimation? Class covariances are strongly aligned (Frobenius cosine 0.580 ± 0.015 between base pairs), but after the best scalar fit of the shared estimate the residual still carries 0.628 ± 0.018 of each covariance's norm (Figure S3), so "most of the structure is shared" is too strong. The shared covariance is a statistically efficient prior, not a geometric summary: a class supplies 500 images in d=512, the shared term 10,000. A control separates the two readings. At matched samples, spreading the shared term over 20 classes rather than 1 is worth $+2.18 \pm 1.04$ points, while twentyfold more images at fixed classes is worth $+0.40 \pm 0.08$ (Appendix S31): averaging over classes cancels the class-specific term Δ_c of Proposition 1, which also gives the scaling in B a mechanism. This is why the memory matches or exceeds full covariance rather than approaching it from below: the full estimator spends d^2 parameters per class on residual structure it cannot estimate.

Which prior? Two different mechanisms meet here: a lighter regulariser reduces estimation error, as Proposition 1 describes, while the between-class term below is a geometric correction. At the same d^2+Kd, an analytic readout (Zhuang et al., 2022; 2024) — a feature autocorrelation and a closed-form ridge solve — beats our shared term on every architecture (+2.43, +3.75, +4.89 points; Table S17; +0.63 on CIFAR once the shared term's shrinkage is tuned on held-out data). Its advantage is its statistic, not its solve: used as our prior, the uncentred autocorrelation with a light ridge scores 0.4486 with no residual, 90% of the full-covariance gain, and 0.5173 with rank-32 residuals, $+4.13 \pm 0.22$ over our headline and above every row of Table 1. Two parts carry the gain (Table 2; within- plus between-class scatter is the total scatter, the classical alternative in discriminant analysis (Fukunaga, 1990)): the plain ridge in place of FeCAM's shrinkage ($+2.03 \pm 0.13$), and the base classes' between-class scatter ($+1.81 \pm 0.23$ over a within-class prior with the same ridge); the global mean adds nothing (-0.02 ± 0.08; at most 0.3 points anywhere). The between-class term moves accuracy to the arriving classes (+2.50) at the base classes' expense (−0.92): it discounts the directions the backbone spent separating its training classes. The term is positive on all 17 seeds of the five backbones we train with the CIFAR pad-crop recipe (+0.15 to +1.81; least on Tiny-ImageNet, where the ridge carries nearly all of the gain) and negative on all 9 seeds of the pretrained donors, which never were (−0.73 to −0.42; sign tests p=8×10^{-6} and 0.002). The between-class scatter of 20 classes the backbone never trained on, removed from the label space and rescaled to the same trace, adds only $+0.48 \pm 0.19$ against $+1.91 \pm 0.29$, and random directions with the same spectrum add nothing (-0.01 ± 0.15), so it is the trained classes' directions, not the size of the term. Nor is it a recalibration of base against arriving classes: a single penalty on the base classes' scores, chosen on held-out data, adds $+0.01 \pm 0.07$ on CIFAR, at most +0.43 on any backbone even when read on the test split, and nothing on top of the between-class prior (Appendix S39). And it responds to an intervention: fine-tuning only the last block of the ImageNet ResNet on Food-101's 20 base

classes turns its between-class term from -1.11 ± 0.20 to $+0.36 \pm 0.20$ on every class order, while fine-tuning on 20 classes held out of the label space leaves it negative (-0.81 ± 0.34), all on the same 80-class label space (Appendix S28). A specialisation index measured without it — the base classes' between-class share of variance over other classes' — predicts its sign in 14 of 15 settings (Spearman 0.79; Appendix S29); the exception, the Food-101 backbone under stronger augmentation (index 3.0, term -0.09 ± 0.25), makes specialisation necessary but not sufficient. It does not fade with B as we predicted ($+1.06$ to $+1.81$ from B=5 to 50; Appendix S39). Chosen on held-out training data instead of the test split, the same prior and ridge win on all 5 seeds; tuning full covariance's shrinkage on the same split does not help it, so the variant's $+5.27 \pm 0.49$ over full covariance is not tuning effort. Most of its gain reaches the arriving classes ($\mathrm{EC}_{50}^{\mathrm{new}}$ 91.0 ± 3.8 against 65.8 ± 6.8). The pooling and sample-efficiency controls reproduce under it, except at n=10 (-2.03 against FeCAM's prior). We keep FeCAM's prior as the headline only because every other study here used it. For deployment we recommend the uncentred prior at rank 32: held-out data select it and it is the most accurate row of Table 1. Two untuned constants inherited from FeCAM, the power transform and the residual floor, are load-bearing (Appendix S40).

5.5 Sample efficiency

Limiting each arriving class to n labelled images, per-class full covariance inverts (Table S7): best of the covariance family at $n = 500$, it falls below a class mean at $n = 10$ (0.1763 against 0.2746), and FeCAM's few-shot shrinkage still trails the shared-prior memory by 2.4–5.8 points on every seed at $n \leq 50$ (Appendix S7). Shrinkage toward the prior loses to rank-32 residuals at $n \geq 100$ at 13× the storage (Table S13).

5.6 Where the findings stop

Architectures and datasets. The second-order claim replicates on VGG-16-BN (Simonyan & Zisserman, 2015) and ResNet-50, where it outruns prototype multiplicity by 3.5 and 13× (14× on ResNet-18; Appendices S2, S3). On Tiny-ImageNet (Le & Yang, 2015) at 200 classes rank-32 residuals exceed full covariance on all three seeds (0.3366 against 0.3196) at 14× less memory. Stitching the two datasets into one disjoint 300-class stress test lifts the ceiling that censors EC_{τ}: nothing saturates, and ours beats every other readout we ran there, RanPAC-10000 included (Appendix S4).

Depth and hostability. On ResNet-18, layer3 is 5.4 points worse than layer4 on the base classes yet serves 100 classes 6.3 points better, on every seed and at every budget from 0.5 to 8 MB, and VGG-16 peaks earlier still; on CIFAR the gap needs the second-order memory (a class mean shows +0.7, not sign-consistent). It is absent on ResNet-50, on Tiny-ImageNet and on every donor fed upsampled CIFAR, small on the one donor where it reappears at native resolution, and large again for a backbone trained on few classes at native resolution (below), so it belongs mainly to backbones trained on few classes, not to depth or resolution (Appendices S45, S17, S46). Which arrivals a representation can host is predictable at arrival from their competition with the classes then present (encroachment, Spearman -0.70 ± 0.02 with per-class accuracy), not from the base classes (separation from the base prototypes, $+0.10 \pm 0.13$; Appendices S9, S16, S18).

Representations we did not train. We repeat the protocol on ResNet-50 supervised on ImageNet-1k, CLIP ViT-B/16 (Radford et al., 2021) and DINOv2 ViT-B/14 (Oquab et al., 2024), fed the CIFAR images upsampled to 224×224, so pretraining, architecture and resolution change at once. The efficiency claim holds: rank-32 residuals match or beat full covariance at 2.6–5.3% of its memory ($+2.02 \pm 0.30$ points on the ResNet, parity on CLIP and DINOv2). Second-order estimation still outweighs prototype multiplicity (3.2 to 6.0×), but prototypes now help and on these upsampled images every donor is monotone in depth, consistent with Harun et al. (2024); donors saturate at τ=0.70, so we report accuracy at K=100 (Appendix S26). Their pretraining overlaps CIFAR-100: the deployed case, not novel categories.

At native resolution. Upsampling 32-pixel images confounds every donor result above, so we repeat the donor protocol on Food-101 (Bossard et al., 2014): 100 classes of 750 training images at about 512 pixels, viewed at 224, with 20 base classes and the same sessions (Appendix S27). The efficiency claim holds: rank-32 residuals match or beat full covariance on all three donors at 2.6–5.3% of its memory (DINOv2 a tie, -0.12 ± 0.17), and second-order estimation outweighs prototype multiplicity (1.9 to 10.6×). The memory is undominated on all six donor settings, against every comparator of Table 1 (Appendix S27). The overlap caveat is stronger here: DINOv2's pretraining set includes a million images retrieved around Food-101's training split (Oquab et al., 2024, Table 15). On the ImageNet ResNet layer3 now edges layer4 ($+0.33 \pm 0.05$, ten class orders), so upsampling may have contributed to the donors' depth reversal.

Trained at native resolution. We also train ResNet-18 from scratch on 20 Food-101 base classes at 224 pixels (3 seeds; Appendix S28). The findings of our 32-pixel backbones return: second-order estimation gains +8.68 over a class mean while more prototypes lose (-1.89 at K=8); rank-32 residuals beat full covariance by +1.56 at 7.4% of its memory, undominated; the between-class term is positive (+1.12) and the uncentred prior adds +3.27; layer3 hosts 13.5 points more than layer4 while 4.4 points worse on the base classes; and the pooled-versus-arriving gap reproduces ($\mathrm{EC}_{50} = 47.7$ against $\mathrm{EC}_{50}^{\mathrm{new}} = 38.4$). Retrained with random-resized-crop instead of the CIFAR pad-crop, all of this holds except the between-class term, which vanishes (Table S26). What the donors lack is training on the base classes, not resolution.

6 Limitations

Storage efficiency holds in ten representation–dataset settings across CIFAR-100 and Food-101, with Tiny-ImageNet and a 300-class union as further tests, with magnitudes that vary; the accuracy margin of our headline over full covariance is small and does not survive a multiplicity correction (Appendix S23). The depth and prototype findings fail on the donors, and depth also on ResNet-50 and Tiny-ImageNet (§5.6). Two baseline hyperparameters were read on the test split (held-out selection lands within 0.25 points; Appendix S42).

The backbones we train are convolutional; at native resolution we train only ResNet-18, on one dataset (Food-101) and three seeds, and the CIFAR donors see upsampled images. Our largest label space, 300 classes, is stitched, not ImageNet (Appendix S4). EC_τ depends on a stated readout and threshold, saturates at B=50, and orders memories almost identically to average incremental accuracy (Appendix S34). RanPAC-10000 is more accurate than our memory in most settings, by up to 4.6 points at 11–54× the storage; the uncentred prior overtakes it on ResNet-18, ResNet-50, the 300-class union and the Food-101 ResNet-18, but not on VGG-16 (2.7 points behind at 1/54 of its storage) or the pretrained ViTs, and removes the non-significant RanPAC-1000 domination on VGG-16. Its between-class gain is absent on one augmented backbone (§5.4). Each donor's seeds vary only the class order and base split, the hostability predictor was measured on CIFAR-100 only, and about three quarters of the variance behind our memory's quoted EC_{50} spread is arrival order rather than representation (76% for the abstract's ±6.8, Appendix S37).

7 Conclusion

Freezing turns class-incremental learning into a measurement, and the measurement says what binds: estimating second-order structure, not counting prototypes. The memory it points to holds the frontier in nine of ten representation–dataset settings, including native resolution on Food-101, and at 300 classes, and whitening it by the base classes' between-class scatter as well strengthens it on the backbones that separate those classes far more than others.

Use of Large Language Models

The research question, the expansion protocol, the capacity measure, the memory design, the mathematical claims and every experimental decision are the authors' own. The authors trained the backbones, ran and re-ran every experiment, wrote the manuscript, audited each reported number against the raw result files, and take full responsibility for every claim and artefact.

An LLM-based assistant played a supporting role in three ways. Writing: it polished and tightened prose the authors had written, and rendered figures from the result files. Retrieval and discovery: it ran literature searches that surfaced candidate related work, including Honda (2026); the authors read, selected and checked every reference against the publisher's record. Ideation and execution: it suggested a few additional controls (the pooling control of Appendix S31, the arrival-order decomposition of Appendix S37 and the four controls of Tables S8, S3, S15 and S42), which the authors evaluated and chose to run, and it helped with code implementation — drafting baselines, analysis scripts and the verification checker, which the authors reviewed, corrected and ran. It did not draft any section of the manuscript from scratch, was not used to derive or prove the mathematical claims, and generated no dataset. All AI-drafted code was tested by the authors, and every number it produces is regenerated from the raw result files by the supplementary reproduce.sh, which rebuilds every table, figure and interval and checks 2,322 printed quantities against them.

Reproducibility Statement

Every number in this paper is produced by the code in the supplementary material and can be regenerated from it. Concretely: §3 specifies the protocol (which data each memory may touch at each session, and the two baselines that deliberately violate it); the setup paragraph of §5 gives the architecture and epoch budget, and the training recipe is SGD with Nesterov momentum 0.9, peak learning rate 0.1 under a one-cycle schedule (15% warm-up), weight decay 5×10^{-4}, batch size 128, 100 epochs (train_base.py); and all reported quantities are means over seeds — five for the CIFAR base-20 experiments and three for most others, stated in every caption — where a seed fixes the backbone initialisation, the data order and which classes form the base set. Per-seed differences or their range are given for the comparisons on which a claim of consistency rests, and every per-seed value is recomputable from the per-run JSON. The supplementary archive contains the model and memory implementations, the scripts that produced each result file (commands.sh lists the exact invocation behind every one), the raw per-run JSON results, and a script that re-derives 2,322 reported quantities — every cell of the main comparison tables and the ratios asserted in prose — from those results and checks them against the values printed here; reproduce.sh runs the whole chain end to end. Datasets are CIFAR-100, Tiny-ImageNet and Food-101, all public. Results were produced with Python 3.8.20, PyTorch 2.4.1, torchvision 0.20.0, NumPy 1.24.3 and SciPy 1.10.1 under CUDA 11.8 on A100-80GB GPUs; the single exception is DINOv2 feature extraction: its setup.py declares Python ≥ 3.9, but the released attention.py uses X | None type syntax that requires ≥ 3.10, so we ran it under Python 3.12.11 with PyTorch 2.6.0 (features are cached to disk, so every downstream number comes from the first environment). CLIP features use open_clip 2.32.0. Two details that silently change results are reported in the text rather than buried: the per-class scale normalisation required before Mahalanobis scoring (§4), and the fact that memory figures count stored float32 values rather than a nominal parameter count.

Ethics Statement

This work studies the class-expansion capacity of frozen image representations on three standard public image-classification benchmarks (CIFAR-100, Tiny-ImageNet, Food-101). It involves no human subjects, no personally identifying data, and no new data collection, and we see no direct pathway from these results to harmful application. Two indirect considerations are worth stating. First, the setting we study — adding classes to a deployed recogniser without retraining it — is exactly the setting in which a system may be extended

to categories its original developers never evaluated, and our own results show that whether a new class is well served varies sharply and predictably across classes (§5.6); a system expanded this way can therefore acquire uneven per-class reliability that aggregate accuracy hides, which is an argument for reporting the decomposed capacity we propose rather than a pooled number. Second, the novelty-detection results (Appendix S25) show that a frozen representation becomes progressively worse at recognising that an input belongs to no class it holds as its label space fills, so such a system should not be relied upon to abstain on out-of-distribution inputs.

References

Nader Asadi, MohammadReza Davari, Sudhir Mudur, Rahaf Aljundi, and Eugene Belilovsky. Prototype-sample relation distillation: Towards replay-free continual learning. In ICML, 2023.

Peyman Bateni, Raghav Goyal, Vaden Masrani, Frank Wood, and Leonid Sigal. Improved few-shot visual classification. In CVPR, 2020.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In European Conference on Computer Vision (ECCV), pp. 446–461, 2014.

SueYeon Chung, Daniel D. Lee, and Haim Sompolinsky. Classification and geometry of general perceptual manifolds. Physical Review X, 8(3):031003, 2018.

Uri Cohen, SueYeon Chung, Daniel D. Lee, and Haim Sompolinsky. Separability and geometry of object manifolds in deep neural networks. Nature Communications, 11:746, 2020.

Chandler Davis and William M. Kahan. The rotation of eigenvectors by a perturbation. III. SIAM Journal on Numerical Analysis, 7(1):1–46, 1970.

Bernhard N. Flury. Common principal components in k groups. Journal of the American Statistical Association, 79(388):892–898, 1984.

Jerome H. Friedman. Regularized discriminant analysis. Journal of the American Statistical Association, 84(405):165–175, 1989.

Keinosuke Fukunaga. Introduction to Statistical Pattern Recognition. Academic Press, second edition, 1990.

Tomer Galanti, András György, and Marcus Hutter. On the role of neural collapse in transfer learning. In ICLR, 2022.

Sixue Gong, Vishnu Naresh Boddeti, and Anil K. Jain. On the capacity of face representation. arXiv preprint arXiv:1709.10433, 2017.

Dipam Goswami, Yuyang Liu, Bartłomiej Twardowski, and Joost van de Weijer. FeCAM: Exploiting the heterogeneity of class distributions in exemplar-free continual learning. In NeurIPS, 2023.

Md Yousuf Harun, Kyungbok Lee, Jhair Gallardo, Giri Krishnan, and Christopher Kanan. What variables affect out-of-distribution generalization in pretrained models? In NeurIPS, 2024.

Tyler L. Hayes and Christopher Kanan. Lifelong machine learning with deep streaming Linear Discriminant Analysis. In CVPR Workshops, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In CVPR, 2016.

Hiroto Honda. Adversarial pseudo-replay for exemplar-free class-incremental learning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 7493–7502, 2026.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In CVPR, 2019.

Paul Janson, Wenxuan Zhang, Rahaf Aljundi, and Mohamed Elhoseiny. A simple baseline that questions the use of pretrained-models in continual learning. In NeurIPS Workshop on Distribution Shifts, 2022.

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

Ya Le and Xuan Yang. Tiny ImageNet visual recognition challenge. Technical report, Stanford University, CS231N course report, 2015.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. Journal of Multivariate Analysis, 88(2):365–411, 2004.

Marc Masana, Xialei Liu, Bartłomiej Twardowski, Mikel Menta, Andrew D. Bagdanov, and Joost van de Weijer. Class-incremental learning: Survey and performance evaluation on image classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(5):5513–5533, 2023.

Wojciech Masarczyk, Mateusz Ostaszewski, Ehsan Imani, Razvan Pascanu, Piotr Miłoś, and Tomasz Trzciński. The tunnel effect: Building data representations in deep neural networks. In NeurIPS, 2023.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In Psychology of Learning and Motivation, volume 24, pp. 109–165. Academic Press, 1989.

Mark D McDonnell, Dong Gong, Amin Parvaneh, Ehsan Abbasnejad, and Anton van den Hengel. RanPAC: Random projections and pre-trained models for continual learning. In NeurIPS, 2023.

Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013.

Saleh Momeni, Sahisnu Mazumder, and Bing Liu. Continual learning using a kernel-based method over foundation models. In AAAI, volume 39, pp. 19528–19536, 2025.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. Transactions on Machine Learning Research, 2024.

Grégoire Petit, Adrian Popescu, Hugo Schindler, David Picard, and Bertrand Delezoide. FeTrIL: Feature translation for exemplar-free class-incremental learning. In WACV, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In ICML, volume 139 of PMLR, pp. 8748–8763, 2021.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K. Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In ICLR, 2021.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. iCaRL: Incremental classifier and representation learning. In CVPR, 2017.

Grzegorz Rypeść, Sebastian Cygert, Tomasz Trzciński, and Bartłomiej Twardowski. Task-recency bias strikes back: Adapting covariances in exemplar-free class incremental learning. In NeurIPS, 2024.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In NeurIPS, 2017.

Xiaoyu Tao, Xiaopeng Hong, Xinyuan Chang, Songlin Dong, Xing Wei, and Yihong Gong. Few-shot class-incremental learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12183–12192, 2020.

Songsong Tian, Lusi Li, Weijun Li, Hang Ran, Xin Ning, and Prayag Tiwari. A survey on few-shot class-incremental learning. Neural Networks, 2024.

Michael E. Tipping and Christopher M. Bishop. Probabilistic principal component analysis. Journal of the Royal Statistical Society: Series B, 61(3):611–622, 1999.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In NeurIPS, 2014.

Da-Wei Zhou, Zi-Wen Cai, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. International Journal of Computer Vision, 2024a.

Da-Wei Zhou, Qi-Wei Wang, Zhi-Hong Qi, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Class-incremental learning: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024b.

Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In CVPR, 2021.

Huiping Zhuang, Zhenyu Weng, Hongxin Wei, Renchunzi Xie, Kar-Ann Toh, and Zhiping Lin. ACIL: Analytic class-incremental learning with absolute memorization and privacy protection. In NeurIPS, 2022.

Huiping Zhuang, Run He, Kai Tong, Ziqian Zeng, Cen Chen, and Zhiping Lin. DS-AL: A dual-stream analytic learning for exemplar-free class-incremental learning. In AAAI, volume 38, pp. 17237–17244, 2024.

S1 The capacity map

Four factors associated with expansion capacity, collected in one place. Each panel is reported quantitatively in the section it cites; the figure exists to show that capacity is not one-dimensional.

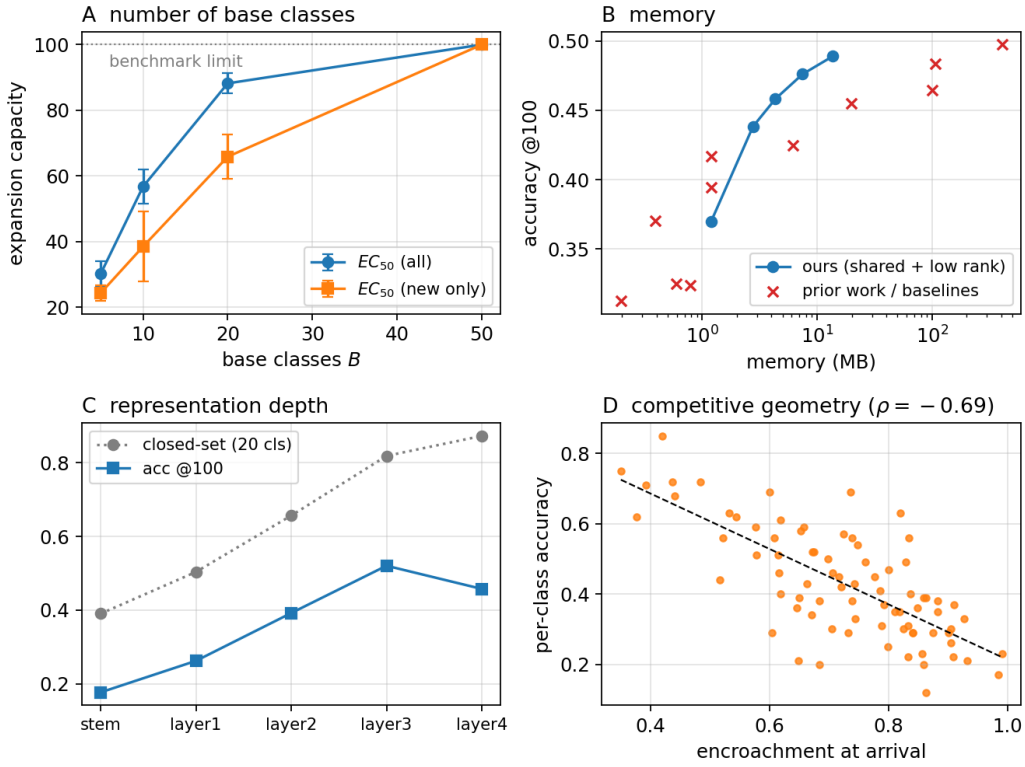


Figure S1: Factors associated with how many classes a frozen representation supports, in the settings we evaluate. Expansion capacity is not one-dimensional: it rises with the number of classes the backbone was trained on (A) and with statistically supported memory (B), is larger at an intermediate depth than at the most discriminative layer on two of three trained backbones (C), and falls with how strongly an arriving class competes with those already present (D). Panels A–C use our memory; D shows per-class outcomes against a quantity measurable the moment a class arrives; the memory is the shared-whitening low-rank residual of §4. Error bars, where shown, are ±1 standard deviation over seeds.

S2 VGG-16-BN

Table S1: VGG-16-BN, three seeds, 20 base classes expanded to 100. Closed-set accuracy on the base classes is 0.8522 ± 0.0285.

Method	Acc@100	Gain over class mean (per seed)	Memory (MB)
Full covariance	$\mathbf{0.3356 \pm 0.0159}$	+.077, +.082, +.078	100.20
Shared+LR32 (ours)	0.3299 ± 0.0161	+.072, +.075, +.073	7.46
Multi-prototype (K=4)	0.2774 ± 0.0153	+.020, +.023, +.020	0.78
Shared only	0.2749 ± 0.0155	+.017, +.020, +.018	1.20
Multi-prototype (adaptive)	0.2729 ± 0.0153	+.015, +.019, +.015	0.51
Associative prototypes	0.2667 ± 0.0156	+.008, +.014, +.009	0.51
Class mean	0.2568 ± 0.0153	—	0.20
Layer-wise (shared whitening, rank-16 residual)			
layer2 (d=256)	$\mathbf{0.4974 \pm 0.0087}$	closed-set 0.7953 ± 0.0313	
layer3 (d=512)	0.4733 ± 0.0050	closed-set $\mathbf{0.8498 \pm 0.0333}$	
layer4 (d=512)	0.3220 ± 0.0144	closed-set 0.8497 ± 0.0275	

The full VGG-16 frontier. Checking "undominated" here needs every comparator of Table 1, not just the prototype family, the shared term and full covariance. Table S2 adds RanPAC at all four widths, FeTrIL with its published head, the analytic readout, and our own rank ladder, and it changes the conclusion: on VGG-16 our rank-32 memory is dominated, by RanPAC-1000, which is both smaller (6.15 against 7.46 MB) and more accurate (0.3373 against 0.3299). Our family holds the frontier below that point, at ranks 8 and 16, and loses it above. This is the only one of the six representations where any row dominates ours (on the donors, against the comparators of Table S20). The exception belongs to FeCAM's prior, not to the residual: with the uncentred prior of Appendix S39 rank 32 reaches 0.3488 at the same 7.46 MB, and no row dominates it.

Table S2: VGG-16-BN, 20 base classes expanded to 100, mean±std over 3 seeds, ordered by storage. "Dominated" marks a row that some smaller row beats on accuracy.

Memory	MB	Acc@100	Dominated
Class mean	0.20	0.2568 ± 0.0153	—
Multi-prototype K=2	0.39	0.2744 ± 0.0145	—
FeTrIL (SVM head)	0.39	0.2826 ± 0.0196	—
Multi-prototype (adaptive)	0.50	0.2729 ± 0.0153	yes
Associative prototypes	0.51	0.2667 ± 0.0156	yes
Multi-prototype K=4	0.78	0.2774 ± 0.0153	yes
Shared only	1.20	0.2749 ± 0.0155	yes
Analytic readout (ACIL)	1.20	0.3124 ± 0.0159	—
Multi-prototype K=8	1.56	0.2682 ± 0.0181	yes
Shared+LR8	2.76	0.3164 ± 0.0134	—
Shared+LR16	4.33	0.3220 ± 0.0144	—
RanPAC M=1000	6.15	0.3373 ± 0.0184	—
Shared+LR32, uncentred prior	7.46	0.3488 ± 0.0180	—
Shared+LR32	7.46	0.3299 ± 0.0161	yes
Shared+LR64	13.72	0.3348 ± 0.0153	yes
RanPAC M=2000	19.93	0.3523 ± 0.0178	—
FeCAM (full covariance)	100.20	0.3356 ± 0.0159	yes
RanPAC M=5000	107.04	0.3714 ± 0.0174	—
RanPAC M=10000	404.82	0.3757 ± 0.0180	—

S3 ResNet-50

Table S3: ResNet-50 (d=2048), three seeds, 20 base classes expanded to 100; closed-set accuracy on the base classes is 0.8597 ± 0.0359. The layer-wise block (shared whitening, rank-16 residual; "closed-set" is the memory's own accuracy at 20 classes) is also three seeds; layer3−layer4 is +0.42, −2.40, −1.17 points per seed.

Method	Acc@100	Memory (MB)
Shared+LR32 (ours)	0.5405 ± 0.0094	41.79
FeCAM (full cov.)	0.5384 ± 0.0064	1600.78
Shared only	0.4388 ± 0.0099	16.78
Associative prototypes	0.3595 ± 0.0153	2.71
Multi-prototype (K=4)	0.3529 ± 0.0170	3.12
Multi-prototype (adaptive)	0.3500 ± 0.0143	2.71
Class mean	0.3373 ± 0.0143	0.78
Layer-wise (shared whitening, rank-16 residual), 3 seeds		
layer2 (d=512)	0.3583 ± 0.0299	closed-set 0.6035 ± 0.0450
layer3 (d=1024)	0.5166 ± 0.0125	closed-set 0.8018 ± 0.0315
layer4 (d=2048)	$\mathbf{0.5271 \pm 0.0100}$	closed-set 0.8575 ± 0.0366

The full ResNet-50 frontier. Table S4 does the same here, adding RanPAC at four widths and FeTrIL with its published head so that "undominated" is checked against the full set.

ResNet-50 is the opposite case to VGG-16: at d=2048 our rank-32 memory is undominated and itself dominates both full per-class covariance (38× its storage) and RanPAC-5000 (3.3×). Only RanPAC-10000 reaches higher accuracy, by 0.07 points at 11× the storage, which is not domination; the analytic readout dominates RanPAC-2000. With the uncentred prior (Appendix S39) rank 32 reaches 0.5631 and dominates RanPAC-10000 as well.

Table S4: ResNet-50, 20 base classes expanded to 100, mean±std over 3 seeds, ordered by storage. "Dominated" marks a row beaten on accuracy by some smaller row.

Memory	MB	Acc@100	Dominated
Class mean	0.78	0.3373 ± 0.0143	—
FeTrIL (SVM head)	1.56	0.3974 ± 0.0129	—
Multi-prototype (adaptive)	2.71	0.3500 ± 0.0143	yes
Associative prototypes	2.71	0.3595 ± 0.0153	yes
Multi-prototype K=4	3.12	0.3529 ± 0.0170	yes
RanPAC M=1000	12.01	0.4462 ± 0.0101	—
Shared only	16.78	0.4388 ± 0.0099	yes
Analytic readout (ACIL)	16.78	0.4877 ± 0.0082	—
RanPAC M=2000	31.65	0.4794 ± 0.0112	yes
Shared+LR32, uncentred prior	41.79	0.5631 ± 0.0076	—
Shared+LR32	41.79	0.5405 ± 0.0094	—
RanPAC M=5000	136.34	0.5230 ± 0.0079	yes
RanPAC M=10000	463.41	0.5412 ± 0.0083	yes
FeCAM (full covariance)	1600.78	0.5384 ± 0.0064	yes

S4 A 300-class label space

Every EC_τ in the body is bounded by a benchmark ceiling: CIFAR-100 stops at 100 classes, where Table 1's best baseline reaches 98.3 and our uncentred variant all 100. To measure a crossing rather than a bound we built a single 300-class problem from the two benchmarks we already have — CIFAR-100 upsampled to 64×64 and Tiny-ImageNet, whose label sets are disjoint — trained a ResNet-18 on 20 base classes of the union, froze it, and expanded to all 300 in six sessions.

Table S5: 20 base classes expanded to 300 on the CIFAR-100 + Tiny-ImageNet union at 64×64, mean±std over 3 seeds, ordered by storage. Closed-set accuracy on the 20 base classes is 0.8746. No EC_{50} here is censored: every memory crosses τ=0.50 strictly inside the label space, on every seed. "Dominated" marks a row beaten on accuracy by a smaller one.

Memory	MB	Acc@300	EC_{50}	Dominated
Class mean	0.59	0.1489 ± 0.0164	49.5	—
Shared only	1.59	0.2286 ± 0.0181	61.7	—
Analytic readout	1.59	0.2280 ± 0.0088	70.7	—
Multi-prototype (K=4)	2.34	0.1559 ± 0.0137	50.0	yes
Shared+LR8 (ours)	6.28	0.3110 ± 0.0116	87.3	—
Shared+LR32 (ours)	20.37	0.3561 ± 0.0075	108.9	—
Shared+LR32, uncentred prior (ours)	20.37	0.3993 ± 0.0067	140.3	—
RanPAC M=2000	21.45	0.2912 ± 0.0076	92.2	yes
RanPAC M=5000	110.86	0.3256 ± 0.0081	105.1	yes
FeCAM (full covariance)	300.59	0.3221 ± 0.0148	89.4	yes
RanPAC M=10000	412.45	0.3444 ± 0.0068	115.9	yes

Three things change and one does not. The censoring goes: at 100 classes RanPAC-10000 reached the ceiling on two of five seeds (the uncentred prior on all five) and every EC_τ we quoted was a lower bound, whereas here the largest EC_{50} is 140.3 in a space of 300, so the metric is measured throughout. The window widens: the spread between best and worst memory is 66.3 classes without the uncentred prior (90.8 with it) against 50.1 on CIFAR alone. And the advantage grows with the label space rather than shrinking —

rank-32 residuals reach 0.3561 at 20.4 MB against full covariance's 0.3221 at 300.6 MB and RanPAC-5000's 0.3256 at 110.9 MB, so at 300 classes our memory is undominated, as at 100, and now also more accurate than every other readout, RanPAC-10000 (0.3444 at 20× the storage) included; the uncentred prior adds $+4.33 \pm 0.55$ points more. What largely does not change is the mechanism's ordering: the analytic readout again separates from the shared Mahalanobis prior at identical storage, by 9.0 classes of EC_{50} (70.7 against 61.7) though not on accuracy at 300 (0.2280 against 0.2286), and prototype multiplicity again buys almost nothing (+0.7 points over a class mean).

This is one backbone on a stitched benchmark and we do not present it as the large-K regime of (2); two datasets glued at a common resolution are not ImageNet. It does remove the specific objection that our capacities are artefacts of a 100-class ceiling.

S5 Does geometric diversity explain capacity beyond B?

21 base sets ($B \in \{5, 10, 20\}$, 7 subsets each, one trained backbone per set). Capacity is EC_{50}^{new}; each diversity statistic is computed from the frozen base-class features before any expansion. A statistic earns its place only if it improves leave-one-out R^2 over B alone — adding any regressor raises in-sample R^2 mechanically.

Diversity statistic	ρ with EC	R^2(D)	LOO-R^2(B+D)	gain over B
baseline EC $\sim B$		0.821	0.786	—
effective rank, centroids	+0.909	0.821	0.781	−0.005
effective rank, features	+0.873	0.810	0.777	−0.009
mean centroid separation	+0.795	0.677	0.731	−0.055
covariance dissimilarity	+0.570	0.249	0.755	−0.031
separation / intra-class radius	+0.271	0.044	0.771	−0.015
minimum separation	+0.208	0.012	0.755	−0.032

B	subsets	EC_{50}^{new} mean	std	range	max/min
5	7	21.8	7.3	15.3–36.1	2.35×
10	7	42.0	8.8	29.3–52.4	1.79×
20	7	62.1	5.8	57.7–73.6	1.28×

Two things follow. The hypothesis that diversity supplants B is rejected on this evidence. But which classes form a small base set matters — a 2.35× spread at B=5, of which roughly two thirds survives the arrival-order correction of Appendix S37 — and none of the statistics we tried explains it. After centring within B, only separation normalised by intra-class radius shows any association ($\rho = +0.426$; others $\leq +0.10$ in magnitude aside from one negative), the same ratio that predicts hostability for individual classes (§5.6). We regard identifying what governs the within-B spread as the most concrete open question this framework raises.

S6 Capacity against base-class count

Table S6: Expansion capacity against the number of base classes, mean±std over 3 seeds, using our memory; the B=20 row therefore differs slightly from the 5-seed values quoted in §5.2. Capacity recovery is $(A_{\text{frozen}} - A_{\text{chance}})/(A_{\text{joint}} - A_{\text{chance}})$ with $A_{\text{joint}} = 0.7782$. Entries at B=50 are lower bounds: the benchmark has only 100 classes.

Base classes B	Closed-set acc	EC_{50}	$\text{EC}_{50}^{\text{new}}$	Capacity recovery
5	0.9047	30.1 ± 3.8	24.3 ± 2.4	0.401
10	0.8803	56.7 ± 5.3	38.4 ± 10.7	0.520
20	0.8732	87.5 ± 3.3	65.5 ± 6.4	0.605
50	0.8114	≥ 100	≥ 100	0.771

S7 Sample efficiency

Proposition 1 reads the rank requirement as $n \gtrsim d/\delta_r^2$ rather than $n \gtrsim r$. This sweep does not separate the two: the useful rank we observe (8 at n=25, 8 or 0 at n=10, 32 from n=50) is also the largest tested rank below n, and rank 32 is degenerate where $n \leq 32$.

A count-dependent mixture in the spirit of Simple CNAPS (Bateni et al., 2020), weighting each class's own covariance by $n_c/(n_c{+}1)$ against the shared term, does not rescue per-class covariance: the better of our ranks 8 and 32 beats it by $+13.06 \pm 1.98$, $+13.96 \pm 0.57$, $+8.93 \pm 0.68$, $+4.49 \pm 0.41$ points at $n = 10, 25, 50, 100$.

Table S7: Accuracy at 100 classes when each arriving class supplies only n labelled images; mean over 3 seeds. Bold marks the best method in each row. The n=500 row is the full-data setting, and differs from Table 1 in the third decimal because this table uses 3 seeds and Table 1 uses 5. Full covariance uses FeCAM's many-shot shrinkage (γ_1=γ_2=1); Table S8 runs FeCAM's prescribed few-shot shrinkage, and Appendix S15 shrinkage toward the shared estimate. At n=10 ranks 8 and 0 are within 0.4 points and not sign-consistent across seeds.

n	Class mean	Full cov.	Shared	Shared+LR8	Shared+LR32	RanPAC-2000
10	0.2746	0.1763	0.3120	**0.3159**	0.2871	0.2050
25	0.2894	0.2018	0.3405	**0.3629**	0.3562	0.2631
50	0.3017	0.3008	0.3565	0.3907	**0.4015**	0.3191
100	0.3074	0.3869	0.3650	0.4151	**0.4349**	0.3710
250	0.3099	0.4433	0.3686	0.4304	**0.4614**	0.4275
500	0.3104	0.4650	0.3701	0.4365	**0.4749**	0.4557

Table S8: Per-class full covariance with FeCAM's few-shot shrinkage. FeCAM prescribes γ_1=γ_2=100 for few-shot sessions (against 1 for many-shot, used everywhere else in this paper); mean±std over 3 seeds, same few-shot draws as Table S7. Stronger shrinkage lifts the per-class estimate above the class mean, so the inversion of §5.5 is a property of the many-shot setting; the shared-prior memory still leads by 2.4–5.8 points on every seed.

n	γ=1	γ=10	γ=100	Class mean	Shared+LR8
10	0.1763 ± 0.0052	0.2452 ± 0.0096	0.2873 ± 0.0125	0.2746 ± 0.0119	$\mathbf{0.3159 \pm 0.0112}$
25	0.2018 ± 0.0079	0.3122 ± 0.0111	0.3177 ± 0.0093	0.2894 ± 0.0094	$\mathbf{0.3629 \pm 0.0090}$
50	0.3008 ± 0.0040	0.3497 ± 0.0093	0.3365 ± 0.0110	0.3017 ± 0.0073	$\mathbf{0.3907 \pm 0.0093}$

S8 Per-stage decomposition

Table S9: Per-stage decomposition for Shared+LR32, mean over 3 seeds (so the 20-class row differs from the 5-seed 0.8782 of §5); new-class accuracy is the macro-average over the arriving classes. Base-class accuracy degrades without any change to the representation or its prototypes, purely from the growing competitor set.

Label space	Base-class acc	New-class acc	All
20	0.8752	—	0.8752
30	0.7598	0.6410	0.7202
40	0.7277	0.5772	0.6524
50	0.7072	0.5382	0.6058
60	0.6918	0.5154	0.5742
80	0.6685	0.4648	0.5157
100	0.6497	0.4312	0.4749

S9 Frozen-geometry predictors

Measuring the geometry. For class c with frozen embeddings Z_c, prototype p_c, let $I_c = \mathbb{E}_{z \in Z_c}\|z - p_c\|$ be the intra-class radius and $S_c(\mathcal{R}) = \min_{r \in \mathcal{R}} \|p_c - p_r\|$ the separation from a reference set $\mathcal{R}$ of classes. We define $\mathrm{RES}_c(\mathcal{R}) = S_c(\mathcal{R})/(I_c + \epsilon)$ and the encroachment $\mathrm{enc}_c = \frac{1}{|Z_c|}\sum_{z \in Z_c} \mathbb{1}[\arg\min_{r \in \mathcal{R} \cup \{c\}} \|z - p_r\| \neq c]$, the fraction of a class's own points already claimed by a foreign prototype. The choice of $\mathcal{R}$ decides whether they work: against the base prototypes, or against the classes present when c arrives.

Table S10: Correlation of frozen-geometry measures with per-class accuracy, over expanded classes only; mean±std over 3 seeds.

Measure	Spearman ρ	Pearson r	Available at arrival
RES vs. base prototypes	$+0.103 \pm 0.125$	$+0.108 \pm 0.146$	yes
RES vs. classes present at arrival	$+0.640 \pm 0.034$	$+0.624 \pm 0.044$	yes
Encroachment enc_c	$\mathbf{-0.699 \pm 0.020}$	-0.733 ± 0.010	yes
Distance to nearest foreign samples	$+0.519 \pm 0.015$	$+0.587 \pm 0.018$	yes
RES vs. all 100 classes (oracle)	$+0.693 \pm 0.072$	$+0.704 \pm 0.049$	no
Separation alone (unnormalised)	$+0.519 \pm 0.035$	$+0.503 \pm 0.048$	yes
Intra-class radius alone	-0.482 ± 0.126	-0.450 ± 0.119	yes
Multimodality (dispersion ratio)	$+0.147 \pm 0.171$	$+0.167 \pm 0.152$	yes

S10 Covariance parameterisation

Table S11: Covariance parameterisation, mean±std over 5 seeds. "Share of gain" is relative to full per-class covariance, over the class-mean baseline. The rows shared with Table 1 agree with it exactly.

Parameterisation	Acc@100	Memory (MB)	Share of gain
None (class mean, first-order)	0.3125 ± 0.0083	0.20	0%
Per-class diagonal	0.3291 ± 0.0094	0.39	11%
One shared covariance, base classes only	0.3700 ± 0.0057	1.20	38%
+ per-class rank 4	0.4194 ± 0.0068	1.98	70%
+ per-class rank 8	0.4382 ± 0.0064	2.76	83%
+ per-class rank 16	0.4582 ± 0.0071	4.33	96%
+ per-class rank 32 (reported)	0.4760 ± 0.0058	7.46	> 100%
+ per-class rank 64 (Appendix S42)	$\mathbf{0.4890 \pm 0.0052}$	13.72	> 100%
Per-class rank 32 (no shared whitening)	0.3967 ± 0.0100	6.46	55%
Per-class rank 128 (no shared whitening)	0.4256 ± 0.0112	25.24	74%
Per-class full covariance	0.4644 ± 0.0042	100.20	100%

S11 The memory in full

§4 states the memory in prose; this is the same thing as executable steps, with every constant named. $\mathcal{Z}_c$ is the set of features of class c at the session it arrives, B the base classes, d the feature dimension and r the residual rank.

1. Feature transform (all classes, every session). L2-normalise the frozen features, then apply the Tukey power $z \mapsto \mathrm{sign}(z)|z|^{0.5}$.
2. Shared prior (once, on the base classes). Pool the centred scatter of every base class, $S = \sum_{c \in B} \sum_{z \in \mathcal{Z}_c} (z - p_c)(z - p_c)^\top$, and divide by $\sum_{c \in B}(|\mathcal{Z}_c| - 1)$. Shrink with FeCAM's rule at $\gamma_1{=}\gamma_2{=}1$: add $\gamma_1 \overline{\mathrm{diag}}\, I$, then $\gamma_2 \overline{\mathrm{off}}\, (\mathbf{1}\mathbf{1}^\top - I)$. Convert to a correlation matrix by dividing by the outer product of the standard deviations, add $10^{-6} I$, invert, and take the Cholesky factor W with $WW^\top = \Sigma_{\mathrm{sh}}^{-1}$. Store W; the inverse is transient, since scoring needs only W, and W is the single $d \times d$ term in the accounting.
3. Per class, at arrival. Store $p_c = \mathrm{mean}(\mathcal{Z}_c)$. Whiten, $\tilde{X}_c = (\mathcal{Z}_c - p_c)W$, and take its top-$r$ SVD. With λ_k the retained eigenvalues of $\tilde{X}_c^\top \tilde{X}_c/(n_c{-}1)$ and V_c their directions, set the isotropic floor $\sigma_c = (\text{tail variance})/(d{-}r) + \gamma_1 \bar{\lambda}$, then rescale: divide λ and σ_c by $(\sigma_c d + \sum_k \lambda_k)/d$, so the whitened covariance has unit mean variance. Store V_c, λ, σ_c. If $n_c < 3$ store only p_c and score class c by the shared term alone.
4. Scoring. $s_c(z)$ is (1) of §4; predict $\arg\max_c s_c(z)$.

Two steps are load-bearing and easy to omit. The rescaling in step 3 is the normalisation of §4: without it the classes are scored on incomparable scales and the memory loses about 12 points (Appendix S21; only the per-class rank-8 variant falls below a class mean). The whitening in step 3 must be the same W used at scoring time: if the shared prior is ever re-estimated, every residual fitted against the old W is invalid, which is the constraint Appendix S38 measures.

S12 Tiny-ImageNet

Table S12: Tiny-ImageNet, 20 base classes expanded to 200, mean±std over 3 seeds; new-class accuracy is macro-averaged over the 180 arriving classes. Chance is 0.005.

Method	Acc@200	New-class acc	Memory (MB)
Shared+LR32, uncentred prior (ours)	$\mathbf{0.3578 \pm 0.0040}$	$\mathbf{0.3456 \pm 0.0010}$	13.92
Shared+LR32 (ours)	0.3366 ± 0.0032	0.3186 ± 0.0007	13.92
Full covariance	0.3196 ± 0.0004	0.2976 ± 0.0018	200.39
RanPAC-2000	0.2767 ± 0.0056	0.2384 ± 0.0054	20.69
Shared only	0.2535 ± 0.0017	0.2210 ± 0.0035	1.39
Associative prototypes	0.1909 ± 0.0035	0.1491 ± 0.0015	1.39
Class mean	0.1800 ± 0.0024	0.1374 ± 0.0025	0.39
Multi-prototype (K=4)	0.1768 ± 0.0013	0.1440 ± 0.0017	1.56

S13 Expansion curves and the memory frontier

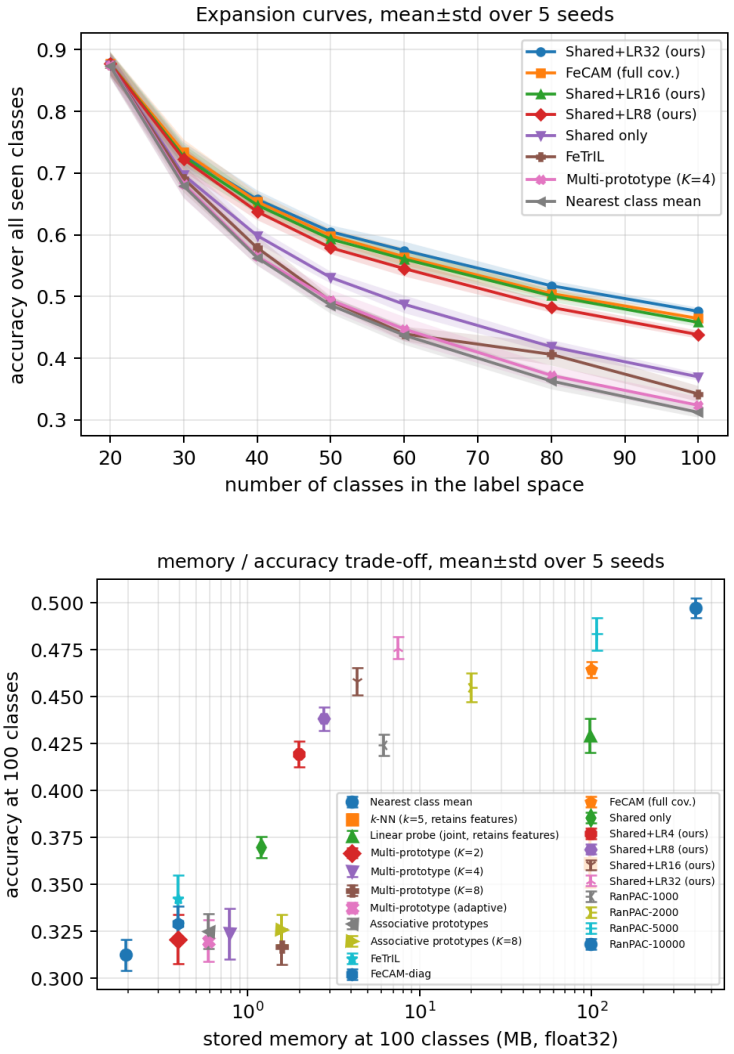


Figure S2: Top: expansion curves from 20 base classes, mean±std bands over 5 seeds, method names as in Table 1. Bottom: accuracy at 100 classes against stored memory (log scale), error bars ±1 std over 5 seeds, RanPAC at all four M included. Legend names follow the code: AAPM is the associative-prototype memory and KNN-5 the k-NN upper bound. The shared-whitening family occupies the upper-left frontier below 20 MB.

S14 Covariance structure and sample efficiency

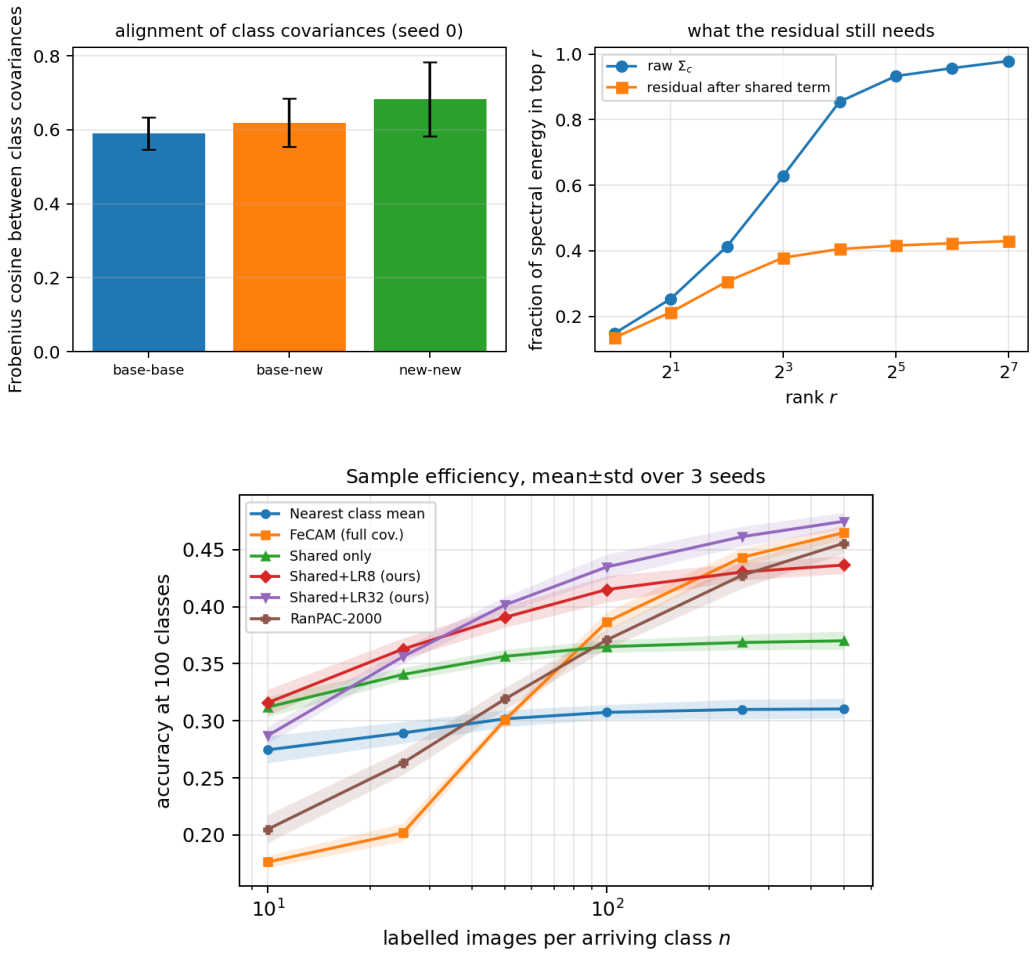


Figure S3: Top: class covariances are strongly aligned (Frobenius cosine, error bars ±1 std over class pairs), but the residual left by the shared term is close to isotropic. Read with §5.4: the shared term works as a statistically efficient prior, not as a geometric summary of what the classes share. Bottom: sample efficiency, mean over 3 seeds with ±1 std bands. Per-class full covariance (at FeCAM's many-shot shrinkage) inverts below $n \approx 50$; the shared component is estimated once from the base classes and is unaffected.

S15 Shrinkage sweep

Shrinkage behaves as theory predicts: it rescues full covariance at low n (0.2018 → 0.3607 at n=25) and its optimal λ drifts with sample size (0.25 → 0.50 → 1), so it needs per-regime tuning.

Rank-32 residuals beat the best-tuned λ at each n on every seed individually, where "best" is the λ with the highest accuracy at that n rather than a fixed one: $+0.014, +0.020, +0.018$ at n=100 (best λ=0.50) and $+0.007, +0.013, +0.010$ at n=500 (best λ=1, i.e. full covariance), while storing 13× less. At n=25 rank 32 loses to the best λ on every seed ($-0.003, -0.006, -0.004$ against λ=0.25), while rank 8, which the sample-size rule of §5.5 selects there, ties it ($+0.002, -0.001, +0.006$); rank 8, at 36× less memory, loses from n=100 upward on the mean (seed 2 at n=100: $+0.001$). The reason it is not merely interpolation is structural. Shrinkage pulls every direction toward the shared estimate, including leading

directions that n samples estimate well; the low-rank residual keeps the top-r class-specific directions at full strength and falls back on the shared estimate elsewhere. Since the spectrum divides sharply into estimable and non-estimable directions (§5.4), the hard split dominates the soft one — and it is what makes the memory $(r{+}1)d$ per class rather than d^2.

Table S13: Shrinkage toward the shared covariance versus a low-rank residual, at three per-class sample sizes; mean over 3 seeds. Every shrinkage variant stores d^2 per class (100.2 MB); ours stores $(r{+}1)d$. Bold marks a win that holds on every seed; at n=25 the "Ours" cell is rank 8, which ties the best λ (rank 32 loses there).

n	Full (λ=1)	λ=0.75	λ=0.50	λ=0.25	Shared	Ours
25	0.2018	0.2904	0.3391	0.3607	0.3405	0.3629 (LR8, 2.8 MB)
100	0.3869	0.4111	0.4176	0.4101	0.3650	**0.4349** (LR32, 7.5 MB)
500	0.4650	0.4618	0.4478	0.4255	0.3701	**0.4749** (LR32, 7.5 MB)

S16 Encroachment-driven rank allocation

Per-class rank set in proportion to arrival-time encroachment, at comparable total memory; mean over 3 seeds. Overall accuracy is essentially unchanged; what moves is the split between base and arriving classes.

Method	Acc@100	Base acc	New acc	EC_{50}	EC_{50}^{new}	Memory
Uniform rank 8	0.4365	0.6505	0.3830	74.0	49.2	2.76 MB
Allocated, mean 8	0.4363	0.5453	0.4090	72.3	**59.5**	2.83 MB
Uniform rank 16	0.4575	0.6533	0.4086	80.1	58.2	4.33 MB
Allocated, mean 16	0.4510	0.5098	**0.4363**	76.1	**68.9**	4.48 MB

S17 Nominal versus effective dimension

Participation ratio $r_{\text{eff}} = (\operatorname{tr}\Sigma)^2/\operatorname{tr}(\Sigma^2)$ of the pooled within-class covariance at each tap, against measured capacity (mean over 3 seeds, memory fixed at Shared+LR16).

	Tap	d	r_{eff}	Acc@100	EC_{50}^{new}
ResNet-18	stem	64	4.4	0.1767	—
	layer1	64	7.8	0.2621	—
	layer2	128	13.2	0.3920	46.9
	layer3	256	**33.6**	**0.5204**	**97.9**
	layer4	512	24.0	0.4575	58.2
VGG-16	layer2	256	33.9	**0.4974**	**84.8**
	layer3	512	**37.5**	0.4733	62.0
	layer4	512	19.3	0.3220	32.4

On ResNet-18, over the three taps where EC_{50}^{new} is defined, r_{eff} orders them by capacity exactly ($\rho = +1.00$) where nominal dimension does not (+0.50); over all five taps on accuracy at 100 classes the two are closer (+1.00 against +0.87, with ties). It resolves the apparent paradox of layer4: it has twice the nominal dimension of layer3 and uses fewer directions. On VGG-16 the ordering does not hold — layer3 has the highest r_{eff} but layer2 the highest capacity ($\rho = +0.50$ over its three taps) — so effective rank is a partial explanation, not a predictor we would rely on. With three taps each these correlations are indicative only. What survives both architectures is that the final block contracts effective rank sharply (33.6 → 24.0, 37.5 → 19.3) while giving up the most capacity, consistent with that block specialising to the base classes; ResNet-50, where the final block is best (Appendix S3), is not in this table.

S18 Which arrivals are hard?

Arriving classes grouped two ways, both computable at arrival; accuracy is per-class at the full 100-class label space, pooled over 3 seeds.

Arriving class lands in	n	mean accuracy
a superclass already represented	213	0.4379 ± 0.0109
a superclass not yet represented	27	0.3781 ± 0.0321
nearest third to an existing prototype	81	0.3577 ± 0.0151
middle third	81	0.4502 ± 0.0164
farthest third	78	0.4877 ± 0.0191

Geometric crowding predicts difficulty (+0.130 from crowded to isolated, ≈ 5 standard errors if the 80×3 class–seed pairs are treated as independent, ≈ 2.9 when clustered by class identity, which is the correct unit since most classes recur across seeds). Semantic overlap does not, and trends the other way (-0.060; 1.8 standard errors unclustered, ≈ 1.0 clustered, not significant either way): a class landing in a superclass the representation already covers benefits from that coverage about as much as it suffers from the extra competitor. This is the same distinction that makes $\mathrm{RES}_{\mathrm{base}}$ uninformative while arrival-time geometry predicts well (§5.6) — what matters is proximity in the embedding to the current competitor set, not kinship with the training categories.

S19 Positioning against the closest methods

Table S14: Mechanism only. We deliberately do not tabulate "capacity metric", "capacity scaling" or "hostability predictor": those are our own definitions, so prior work would score dashes by construction and the comparison would be tautological. The distinction that matters is stated here, not ticked. This work against the closest frozen-backbone methods, all exemplar-free on a frozen extractor (RanPAC is run as its readout alone, without its first-session adaptation and with the ridge at $10^{-3}\,\mathrm{tr}\,G/M$ rather than validated; FeTrIL uses its published SVM head, Appendix S20; streaming LDA runs as published, with its covariance updated every session). All of them share the frozen extractor and one prototype per class with us. The distinction is that they treat the frozen representation as a means to an incremental classifier, whereas we treat its capacity to host unseen classes as the quantity being measured, and derive the memory from the measurement.

	2nd-order per class	shared prior	low-rank residual
FeTrIL (Petit et al., 2023)	—	—	—
FeCAM (Goswami et al., 2023)	✓	common-Σ variant	—
RanPAC (McDonnell et al., 2023)	—	Gram	—
Analytic readout (Zhuang et al., 2022)	—	autocorrelation	—
Streaming LDA (Hayes & Kanan, 2020)	—	online pooled Σ	—
This work	✓	✓	✓

Further shared-second-order readouts on frozen features: analytic class-incremental learning (Zhuang et al., 2022; 2024), which we run without its random buffer (Table S17; with a random ReLU buffer its readout is RanPAC's, which we run at four widths), stores one autocorrelation matrix for a closed-form ridge readout — the lineage RanPAC builds on — kernel LDA (Momeni et al., 2025) keeps one shared covariance over random-feature-mapped foundation-model features. AdaGauss (Rypeść et al., 2024) sits outside that group: it stores per-class Gaussians in a learned low-dimensional space and adapts them as the backbone keeps changing, so it addresses the drift our protocol removes by construction. Three older notions of capacity should be separated from ours. Closest is Gong et al. (2017), who ask

how many identities a fixed face embedding can resolve at a stated error rate and answer it with packing bounds on a low-dimensional manifold in the representation space. That is the same question in spirit, and Proposition 2 of Appendix S35 is a weaker argument of the same kind. Two differences matter here: their capacity is a bound derived from an assumed geometry and a verification threshold, ours is a number measured under a stated readout, memory budget and arrival order, so it is a property of a protocol rather than of an embedding; and their setting has no memory axis, which is the axis this paper trades along. Manifold capacity (Chung et al., 2018; Cohen et al., 2020) counts the linearly separable class manifolds a representation supports per dimension and rises with depth in ImageNet-trained networks; EC_τ differs in being a finite-sample, memory-conditioned quantity measured on classes the extractor never saw, and §5.6 recovers the monotone-depth result only for such backbones. Neural-collapse theory (Galanti et al., 2022) predicts that transfer to unseen classes improves with the number of source classes; the scaling in B of §5.2 is its measurement as a capacity, with the finding (Appendix S5) that no geometric statistic beats B itself. Finally, Honda (2026) stores a low-rank per-class covariance while the backbone is still updated through pseudo-replay, so their setting differs from ours in more than the covariance form; the per-class low-rank row of Table S11 tests only the storage form, on a frozen backbone.

Does an untuned ridge term handicap RanPAC? No. Table S15 re-runs RanPAC with the ridge parameter validated at every session on a held-out fifth of that session's own training features (a grid of multiples $10^{-7}, \ldots, 10$ of $\operatorname{tr} G/M$, the held-out fifth folded back in once chosen — protocol-legal, since only session-t data is touched). The validated version moves accuracy at 100 classes by -0.13 to $+0.42$ points across the four widths and five seeds and changes no ranking in Table 1; the default $10^{-3} \operatorname{tr} G/M$ is within a decade of what the selection lands on at $M \geq 5000$. RanPAC's first-session adaptation is not applicable here: on a backbone trained from scratch on the first session there is nothing left to adapt, and on the donors of §5.6 it would break the frozen-representation protocol every method is held to.

Table S15: RanPAC with its default ridge term ($10^{-3} \operatorname{tr} G/M$, as in Table 1) against a per-session validated one; Acc@100, mean±std over 5 seeds, and the per-seed difference.

M	default	validated	validated − default, per seed
1000	0.4244 ± 0.0056	0.4264 ± 0.0060	$+0.07, +0.42, +0.28, +0.22, +0.01$
2000	0.4550 ± 0.0078	0.4564 ± 0.0074	$+0.16, +0.09, +0.10, +0.21, +0.13$
5000	0.4834 ± 0.0087	0.4846 ± 0.0090	$+0.18, +0.10, +0.05, -0.04, +0.29$
10000	0.4973 ± 0.0052	0.4969 ± 0.0048	$+0.00, -0.01, -0.13, +0.01, -0.06$

S20 What the comparison holds fixed, and where it does not

A storage claim is only meaningful if every method is given the same information, the same tuning effort and the same accounting. Table S16 records that, method by method: what data each may revisit, what it stores, how its hyperparameters were set, and whether it is a like-for-like comparator or an upper bound that breaks the protocol on purpose.

Table S16: What each row of Table 1 is allowed and how it was tuned. "Session t only" is the protocol: when a class arrives, its features may be used once and are then discarded. Storage is what memory_floats() counts, in float32 values; d=512, K classes, r the residual rank, M RanPAC's projection width.

Method	Data access	Stored (float32)	Hyperparameters
Nearest class mean	session t only	Kd	none
Multi-prototype (K_c per class)	session t only	$K\,K_c\,d$	K_c fixed at $2, 4, 8$
Associative prototypes	session t only	$K\,K_c\,d$ + temperature	temperature on test†
FeTrIL (SVM head)	session t only*	$Kd + K(d{+}1)$	published C=1
Shared covariance (LDA-style)	session t only	$Kd + d^2$	FeCAM's γ_1, γ_2
FeCAM (full covariance)	session t only	$K(d + d^2)$	published γ_1, γ_2
Shrinkage	session t only	$K(d + d^2)$	λ swept (App. S15)
RanPAC (M)	session t only	$M^2 + KM + dM$	M swept; ridge at its default $10^{-3}\mathrm{tr}G/M$ (validated variant in App. S19)
Ours (shared + rank r)	session t only	$d^2 + K(d + rd + r{+}1)$	r=32, a storage cap
Upper bounds — these deliberately break the protocol and are labelled as such:			
k-NN (k=5)	all sessions	every training feature	k fixed at 5
Linear probe (joint)	all sessions	every training feature, then $K(d{+}1)$	60 epochs, fixed

RanPAC's $d{\times}M$ random projection is counted as stored. It could be regenerated from a seed, which would shrink RanPAC-1000 by a third (to 4.2 MB at d=512); the only domination that would change is that RanPAC-1000 would then also dominate rank 16 on VGG-16.

Two qualifications belong with that table († is explained with the markers of Table 1 below). *FeTrIL synthesises pseudo-features for earlier classes by translating the current session's real features onto stored prototypes, so it never revisits earlier data, but it does refit a K-way head from scratch each session. Storage excludes bookkeeping that every method shares — the class labels, the integer ranks, and the scalar normalisers (1 or 2 floats per class, under 0.1% of any row) — and we report the factorised form actually held: one $d \times d$ factor is stored, the Cholesky factor W of $\Sigma_{\mathrm{sh}}^{-1}$, which is the d^2 term (the inverse itself is transient: scoring in (1) needs only W); the per-class residual is held as $V_c \in \mathbb{R}^{r\times d}$, $\Lambda_c \in \mathbb{R}^r$ and σ_c, never expanded to $d \times d$. FeCAM's constants — shrinkage $\gamma_1{=}\gamma_2{=}1$ and the power-transform exponent 0.5 — are its published values and were never searched on our data.

Is anyone handicapped by the preprocessing? Only the covariance family applies FeCAM's power transform. Giving it to the others does not help them: over the same five seeds it moves the analytic readout by -2.21 points and RanPAC-1000 and RanPAC-5000 by -0.01 and -0.49. Nor does the whitening need standardised features: dividing every feature by its base-class standard deviation before the correlation-normalised prior moves our memory by $+0.06 \pm 0.10$ points.

The markers of Table 1. †Two scalar hyperparameters of the associative row (the adaptive-K threshold τ=0.85 and the aggregation temperature T=0.5) were chosen on the test split; selecting them on a held-out fifth of the training data lands within 0.25 points. The residual rank is a storage cap rather than a tuned value, since accuracy is monotone in it (Appendix S42). *FeTrIL uses its published one-vs-rest SVM head; its cross-entropy head is 2.71 ± 0.58 points worse (below). §The k-NN and joint linear probe retain every training feature and are upper bounds. ‡"Shared only" is our shared term alone: one covariance pooled over the base classes at FeCAM's constants. Re-estimated each session it reaches 0.3954

(Appendix S38), and with its shrinkage tuned on held-out data 0.3880 (Appendix S39); streaming LDA (Hayes & Kanan, 2020), which does both and scores with a linear discriminant, reaches 0.4167. ¶Three variants beat our rank-32 headline: rank 64 at the same once-estimated prior, and rank 32 with the prior re-estimated each session (these two share a 5-seed mean of 0.4890 by coincidence and differ on every seed), and rank 32 on the analytic readout's uncentred, ridge-regularised prior, the best of four prior variants read on the test split, whose EC_{50} is censored at 100 on every seed (Appendix S39). Elsewhere we report rank 32 on FeCAM's prior unless a table says otherwise. RanPAC is its readout alone; validating its ridge moves it by at most +0.42 points (Appendix S19). No baseline beats ours at or below its own budget; RanPAC-10000 beats our FeCAM-prior rows with 30–54× more storage and is itself beaten by the uncentred variant at 1/54 of it; the rank-32 margin over FeCAM is small, marginally supported and does not survive a multiplicity correction (Appendix S23), so the memory ratio is the robust part.

FeTrIL with its published head. Training FeTrIL's linear head by cross-entropy is not what Petit et al. (2023) prescribe. Fitting the published one-vs-rest linear SVM instead (LinearSVC, squared hinge, C=1) at every session, on exactly the same real and translated pseudo-features, is worth $+2.71 \pm 0.58$ points at K=100 (5 seeds, paired): 0.3701 ± 0.0135 against 0.3431 ± 0.0131, and EC_{50} 53.4 against 49.0. Table 1 reports the stronger, published version. It does not change any conclusion — FeTrIL remains 10.6 points behind the rank-32 memory at 19× less storage, and the gap to second-order methods is what §5.4 is about — so the comparison is against the method as published.

A shared second-order term, spent discriminatively. The analytic class-incremental readout (Zhuang et al., 2022; 2024) stores a feature autocorrelation matrix R and a cross-correlation C and reads out with the closed-form ridge solution $(R + \lambda I)^{-1}C$. We run it without the random buffer expansion, so it spends exactly the budget our shared covariance spends, $d^2 + Kd$, and answers a question the rest of the paper does not: is a shared second-order term better used generatively, as a Mahalanobis prior, or discriminatively, as a linear solve? Against FeCAM's prior, discriminatively, on every architecture we tried — but Appendix S39 traces the advantage to the readout's statistic rather than to its solve.

Table S17: Analytic readout against our shared covariance at identical storage, mean±std over 5 seeds for ResNet-18 and 3 for VGG-16 and ResNet-50, all on CIFAR-100. Accuracy is at the final label space. The last column is the analytic readout's advantage, in points.

Backbone	MB	Analytic readout	Shared only	Difference
ResNet-18	1.20	0.3942 ± 0.0056	0.3700	+2.43
VGG-16	1.20	0.3124 ± 0.0159	0.2749	+3.75
ResNet-50	16.78	0.4877 ± 0.0082	0.4388	+4.89

What it does not do is scale with a per-class budget: R and C are fixed in size once d and K are, so the only way to buy more is a wider random projection, which is what RanPAC does and what costs M^2. Our rank-32 residual adds +8.2 points over it at 6× the storage.

S21 The scoring rule: log-determinant and scale normalisation

Section 4 drops the class-dependent $\log\det\Sigma_c$ from the Gaussian discriminant and normalises every class covariance to unit mean variance instead. That is a design choice, and a reader is entitled to ask what the textbook score does. Table S18 answers it: the full Gaussian discriminant is far worse, at every covariance parameterisation we use.

Table S18: Accuracy at K=100, mean±std over 5 seeds, CIFAR base-20, under three scoring rules. "Full Gaussian" is $-\frac{1}{2}(z-p_c)^\top \Sigma_c^{-1}(z-p_c) - \frac{1}{2}\log\det\Sigma_c$ on the covariance as estimated; "unnormalised" drops the log-determinant but leaves the estimated scale; ours additionally rescales each covariance to $\mathrm{tr}/d = 1$.

Memory	full Gaussian	unnormalised	normalised (ours)
FeCAM (full covariance)	0.1745 ± 0.0038	0.3229 ± 0.0113	$\mathbf{0.4644 \pm 0.0042}$
Shared + rank-32 residual	0.1471 ± 0.0142	0.3508 ± 0.0145	$\mathbf{0.4760 \pm 0.0058}$
Per-class rank 8	0.0768 ± 0.0278	0.3012 ± 0.0085	$\mathbf{0.3739 \pm 0.0096}$

The gap is 29.00 ± 0.15 points for full covariance and 32.89 ± 1.40 for ours (paired over seeds). The reason is estimation, not principle: $\log\det\hat{\Sigma}_c$ is a sum of d=512 log-eigenvalues estimated from n=500 samples, so its smallest eigenvalues — the least reliable — dominate it, and the term that should calibrate classes against each other instead injects the noisiest statistic each class owns. Normalising the scale away removes both that term and the scale differences it was there to absorb. Two consequences worth stating plainly. First, our scoring rule is not a Gaussian likelihood: it is a Mahalanobis distance under a per-class covariance constrained to unit mean variance. It is not the maximum-likelihood rule of any Gaussian model we know of, since unit-trace covariances of different shape still differ in log-determinant; we justify it empirically, by the table above, not probabilistically. Second, the middle column is the silent failure of §4: dropping the log-determinant without normalising is worse than either principled option, and it is what a straightforward implementation produces.

S22 Variance from initialisation alone

A seed in this paper changes the network initialisation, the data order and which classes form the base set. Appendix S37 isolates the arrival order; this isolates the initialisation. We retrain the base-20 backbone four times with the seed-0 class order and the seed-0 base set, varying only the initialisation and the batch order, and run the identical expansion on each.

Table S19: Accuracy at K=100 over 4 initialisations of one base set and class order, against the 5-seed spread of Table 1, which also moves the base set. "Share" is (init sd/seed sd)^2, the fraction of seed variance the initialisation could account for.

Memory	mean	init-only sd	seed sd	share
Nearest class mean	0.3016	0.0018	0.0083	4.8%
Multi-prototype (K=4)	0.3097	0.0045	0.0135	11.3%
FeCAM (full covariance)	0.4568	0.0021	0.0042	23.9%
Shared + rank-32 (ours)	0.4661	0.0009	0.0058	2.2%

Initialisation accounts for a small part of the reported spread: for the proposed memory, 2.2% of the seed variance, and at most 24% for any memory. Table S19 lists them. Two cautions on those shares. They come from n=4 against n=5, so the F-based 95% intervals are very wide — class mean $[0.5\%, 73\%]$, full covariance $[2.4\%, 361\%]$, ours $[0.2\%, 33\%]$ — and the data therefore do not exclude initialisation accounting for all of full covariance's seed spread. And all four initialisations share the seed-0 base set, so "the base set dominates" is read off one base set. The point estimates put initialisation below a quarter of the seed variance; the intervals cannot bound it tightly at this n. Two consequences. First, the paired margins of Appendix S23 are conservative: the rank-32 advantage over full covariance is +0.93 points averaged over these four initialisations and positive on each, with a spread of 0.19 points, against the ±0.44 the 5-seed interval reports. Second, the depth result is not training noise: with the base set held fixed, layer3 beats layer4 by +6.60, +6.97, +6.25, +6.96 points across the four initialisations, a mean of +6.70 with sd 0.34. Its failure on ResNet-50

(§5.6) and on the donors (§5.6) is therefore a property of the architecture and the base training, not variance we failed to average over.

S23 Uncertainty on every headline comparison

Two analyses, because neither alone is adequate at this many seeds. Seed-level pairs over seeds (n=5 for the CIFAR base-20 rows, n=3 elsewhere): the seed is the true experimental unit, since it resamples the backbone, the data order and which classes form the base set, so this is the honest interval — but t^* is 2.78 at n=5 and 4.30 at n=3, wide by construction either way, and sign consistency remains the more informative statistic at this n. Class-level is a paired bootstrap over the classes of the shared test set: real power, but it conditions on the trained models and so does not capture seed variation. We call a comparison supported only when the seed-level interval excludes zero, the difference has the same sign on every seed, and the class-level interval excludes zero in that direction; the seed-level clause changes no verdict below but makes the criterion genuinely two-unit, since at n=3 sign consistency alone has p=0.25. The class-level interval is built per seed — 10,000 paired resamples of the 100 test classes, percentile bounds — and the table reports the mean of the per-seed bounds, so it keeps single-seed width; we keep this form rather than a bootstrap pooled over seeds because it never lets one seed's classes dominate the interval. No multiplicity correction is applied in the table, and we state plainly what one would do: over the twelve non-donor rows, at Bonferroni-corrected $\alpha/12$ the class-level intervals of the three boundary rows all cross zero — Shared+LR32 vs. FeCAM on CIFAR ($[-0.45, +2.82]$ points), associative aggregation vs. a class mean ($[-0.53, +3.05]$) and Shared+LR32 vs. FeCAM on VGG-16 ($[-1.33, +0.24]$) — so those three become not established, the VGG-16 reversal among them. The other nine are unaffected. Across all twelve the seed-level clause never binds, so the operative gate is the class-level bootstrap, the unit that does not capture seed variation. That interval is the mean of per-seed percentile bounds, chosen for robustness; it is a statement about the label space given these trained models, which is why we pair it with the seed interval rather than replace one by the other. Three verdicts sit within 0.0005 of the boundary. Rows from Table 1 use 5 seeds; rows marked n=3 use 3. † At three seeds this comparison was not established; the two extra seeds moved the class-level lower bound from $-.001$ to $+.001$, so it is supported but marginal, and we do not lean on it.

Comparison	mean	95% CI (seeds)	d_z	consistent	class-level [95% CI]	verdict
Shared: second-order term vs. class mean	+.0575	[+.051, +.064]	+10.4	yes	+.058 [+.045, +.070]	supported
Shared+LR32 vs. Shared (low-rank term)	+.1060	[+.101, +.111]	+24.6	yes	+.106 [+.088, +.124]	supported
Shared+LR32 vs. RanPAC-1000 (matched MB)	+.0516	[+.049, +.054]	+25.1	yes	+.052 [+.036, +.067]	supported
Shared+LR32 vs. RanPAC-10000 (54× MB)	−.0213	[−.025, −.018]	−7.1	yes	−.021 [−.031, −.012]	supported
Shared+LR32 vs. FeCAM (CIFAR-100)	+.0115	[+.007, +.016]	+3.3	yes	+.012 [+.001, +.023]	supported†
layer3 vs. layer4 (n=3)	+.0629	[+.036, +.090]	+5.7	yes	+.063 [+.045, +.080]	supported
Shared+LR32 vs. FeCAM (Tiny-ImageNet, n=3)	+.0171	[+.010, +.024]	+6.1	yes	+.017 [+.010, +.024]	supported
Shared+LR32 vs. FeCAM (VGG-16, n=3)	−.0057	[−.010, −.002]	−3.3	yes	−.006 [−.0110, −.0002]	supported
Associative agg. vs. class mean	+.0127	[+.008, +.018]	+3.2	yes	+.013 [+.0004, +.025]	supported
Shared+LR32 vs. FeCAM (ResNet-50, n=3)	+.0021	[−.007, +.011]	+0.6	no	+.002 [−.007, +.011]	not established
Multi-prototype (K=4) vs. class mean	+.0114	[+.004, +.019]	+1.9	yes	+.011 [−.004, +.027]	not established
Multi-prototype vs. class mean (Tiny, n=3)	−.0032	[−.0061, −.0003]	−2.8	yes	−.003 [−.013, +.007]	not established
Donors (§5.6); a seed varies the class order and base set but not the backbone, so these are the weakest units here:						
Shared+LR32 vs. FeCAM (RN-50/IN-1k, n=3)	+0.0202	[+0.017, +0.023]	—	yes	+0.020 [+0.011, +0.030]	supported
Multi-prototype (K=8) vs. class mean (RN-50/IN-1k, n=3)	+0.0460	[+0.044, +0.048]	—	yes	+0.046 [+0.030, +0.061]	supported
Shared+LR32 vs. FeCAM (CLIP ViT-B/16, n=3)	+0.0025	[+0.002, +0.003]	—	yes	+0.003 [−0.005, +0.010]	not established
Multi-prototype (K=8) vs. class mean (CLIP ViT-B/16, n=3)	+0.0200	[+0.014, +0.026]	—	yes	+0.020 [+0.006, +0.034]	supported
Shared+LR32 vs. FeCAM (DINOv2 ViT-B/14, n=3)	+0.0032	[+0.002, +0.004]	—	yes	+0.003 [−0.001, +0.008]	not established
Multi-prototype (K=8) vs. class mean (DINOv2 ViT-B/14, n=3)	+0.0120	[+0.011, +0.013]	—	yes	+0.012 [+0.002, +0.022]	supported

Three consequences we have applied to the claims in the main text. Our advantage over per-class full covariance is established on Tiny-ImageNet, only marginally on CIFAR-100 and not on ResNet-50, so we rest that comparison on memory rather than accuracy. The prototype-count effect has a seed-level interval that excludes zero (+1.1 points, $[+0.4, +1.9]$) but a class-level interval that does not, so under our rule it is not established; the safe statement is that it is an order of magnitude smaller than second-order structure. And the VGG-16 counterexample, where full covariance wins, is sign-consistent on every seed and supported, though marginally at the class level.

S24 Class-arrival order

Accuracy at 100 classes under four arrival orderings, mean±std over 3 seeds. The random row is the main suite. Read this table with the caveat that each ordering has its own retrained backbones, so the spread across rows is retraining variation, not an order effect: Appendix S37 shows that accuracy at K=100 is order-invariant, to the precision we report, at a fixed backbone (sd 0.0000 over eight permutations). What the table establishes is that choosing an arrival strategy and retraining under it changes nothing beyond seed noise.

Arrival order	Class mean	Shared+LR32	Full covariance
random	0.3104 ± 0.0085	0.4749 ± 0.0073	0.4650 ± 0.0058
near classes first	0.3094 ± 0.0122	0.4737 ± 0.0129	0.4642 ± 0.0110
near classes last	0.3105 ± 0.0096	0.4733 ± 0.0094	0.4632 ± 0.0080
superclass-blocked	0.3106 ± 0.0082	0.4777 ± 0.0079	0.4672 ± 0.0056
spread across orders	0.0012	0.0044	0.0041

S25 Novelty detection degrades as the label space fills

A genuinely expandable system would also need to notice that an input belongs to no class it holds. Using the negative top-1 memory score as a novelty score, and measuring separation between held-out images of seen and unseen classes (mean±std over 3 seeds):

Classes in memory	AUROC	FPR at 95% TPR
20	0.7956 ± 0.0217	0.6345 ± 0.0367
50	0.6299 ± 0.0181	0.8385 ± 0.0219
80	0.5909 ± 0.0557	0.8530 ± 0.0559

Novelty detection degrades sharply as the label space fills, and by 80 classes it is weak enough to be unusable as a gate. The obstacle is the one that also blocks a transferable hostability threshold (§5.6): both depend on how crowded the label space has become, so neither supports a fixed rule though both rank reliably. This is coherent with the encroachment result of §5.6: as classes accumulate, an arriving class stops looking distinguishable from those already present, and the same crowding that makes it hard to host makes it hard to detect. It bounds the ambition of an autonomous expandable system built on a frozen representation, and we report it because that bound is not visible from the accuracy numbers.

S26 Representations we did not train

Three externally pretrained extractors replace f_θ; everything else is unchanged. The class order, base split and session boundaries still come from the seed-s checkpoint, so each donor run is paired class-for-class with the from-scratch run of the same seed. A "seed" here therefore varies the class order and base set, not the backbone — the donor is fixed, which is why the standard deviations in Table S20 are much smaller than those on our own backbones. Images are bicubically resized to 224×224 and normalised with each donor's own statistics; ViT taps are the CLS token at blocks 3, 6, 9 and 12, CNN taps are the four residual stages as before.

Table S20: All methods on all three donors, CIFAR-100 base-20 → 100 classes, mean over 3 class orders; the largest standard deviation anywhere in the table is 0.0048, and deterministic methods have none because the donor and the per-class statistics are fixed once the class set is. Memory (MB) is float32 values stored at K=100. Paired intervals for the headline comparisons are in §5.6.

	RN-50 / IN-1k		CLIP ViT-B/16		DINOv2 ViT-B/14	
Memory	Acc@100	MB	Acc@100	MB	Acc@100	MB
NCM	0.5931	0.8	0.6641	0.3	0.8633	0.3
FeCAM-diag	0.6231	1.6	0.6724	0.6	0.8581	0.6
FeTrIL	0.6445	1.6	0.6983	0.6	0.8382	0.6
Multi-prototype K=4	0.6322	3.1	0.6767	1.2	0.8714	1.2
Multi-prototype K=8	0.6391	6.3	0.6841	2.3	0.8753	2.3
Shared only	0.6786	16.8	0.7226	2.5	0.8623	2.5
Shared+LR8	0.7367	23.0	0.7704	4.9	0.8959	4.9
Shared+LR32	**0.7506**	41.8	0.7836	11.9	0.9020	11.9
FeCAM (full cov.)	0.7304	1600.8	0.7811	225.3	0.8988	225.3
RanPAC M=2000	0.7110	31.7	0.7735	21.9	0.8921	21.9
RanPAC M=5000	0.7450	136.3	**0.7919**	111.9	**0.9040**	111.9

Depth. Table S21 runs the same expansion at each tap. Every donor is monotone: the deepest tap is the best one, in contrast to the from-scratch ResNet-18 of Table S44, where layer3 wins. The gap between layer3 and layer4 on the ImageNet ResNet-50 is −4.3 points, against +6.3 on our ResNet-18.

Table S21: Expansion at each tap on each donor, shared whitening with a rank-32 residual, mean±std over 3 class orders. Accuracy at K=100. ViTs have no spatial stages, so their taps are CLS tokens at four evenly spaced blocks.

Tap	RN-50 / IN-1k	CLIP ViT-B/16	DINOv2 ViT-B/14
stem	0.1667 ± 0.0004	—	—
quarter (layer1/b3)	0.2536 ± 0.0027	0.2862 ± 0.0026	0.2637 ± 0.0017
half (layer2/b6)	0.4061 ± 0.0022	0.5105 ± 0.0013	0.5166 ± 0.0006
three-quarter (layer3/b9)	0.7077 ± 0.0004	0.5914 ± 0.0025	0.6817 ± 0.0006
final (layer4/b12)	$\mathbf{0.7506 \pm 0.0012}$	$\mathbf{0.7836 \pm 0.0003}$	$\mathbf{0.9020 \pm 0.0005}$

What this does and does not establish. It establishes that the memory-efficiency claim is not an artefact of small from-scratch CNNs: it holds, more strongly, on a supervised CNN, a language-supervised ViT and a purely self-supervised ViT. It does not establish expansion to categories the backbone has never seen, since ImageNet-1k and LVD-142M overlap CIFAR-100 semantically, and it does not measure EC_τ at all on these donors, which saturate the benchmark. The honest summary is that the donors test whether the measurement and its conclusions transfer to representations of a kind we cannot train here, and two of the four conclusions examined (storage efficiency and second-order over prototypes transfer; intermediate depth and the prototype null do not) fail.

S27 Food-101 at native resolution

The CIFAR donors of Appendix S26 see 32-pixel images upsampled to 224, which confounds resolution with pretraining. Food-101 (Bossard et al., 2014) has 101 classes of 750 training and 250 test images at about 512 pixels. We keep the first 100 classes in name order, so the label space matches CIFAR-100, view every image at 224 (shorter side to 256, centre crop), and run the donor protocol unchanged: 20 base classes, sessions 20 → 30 → 40 → 50 → 60 → 80 → 100, 3 class orders (food101_suite.sh). Table S22 gives every memory. No row is both smaller and more accurate than our rank-32 memory on any donor. The uncentred prior moves the memory by ResNet-50 $+0.61 \pm 0.74$, CLIP -1.08 ± 0.14, DINOv2

-0.26 ± 0.17 against FeCAM's prior. Depth, rank-32 memory at each tap: ResNet-50: stem 0.1560, layer1 0.2817, layer2 0.4620, layer3 0.6968, layer4 0.6934; CLIP: b3 0.2512, b6 0.4301, b9 0.4298, b12 0.8967; DINOv2: b3 0.1784, b6 0.4581, b9 0.6274, b12 0.9027. A class mean's accuracy at 100 classes does not depend on the class order, so its spread over orders is zero; the other memories vary only through the base split. The ResNet's layer3 advantage holds beyond those three orders: over ten class orders it is $+0.33 \pm 0.05$ points, positive on every one, and at B=10 and B=50 (three orders each) it is $+0.43 \pm 0.63$ and $+0.22 \pm 0.23$, again positive on every order; under a class mean layer4 wins by 9.2 points on every order.

Table S22: Food-101 at native resolution, 20 base classes expanded to 100, accuracy at 100 classes, mean±std over 3 class orders, and stored MB. The two indented rows decompose the prior as in Table 2.

	ResNet-50 (ImageNet-1k)		CLIP ViT-B/16		DINOv2 ViT-B/14	
Memory	Acc@100	MB	Acc@100	MB	Acc@100	MB
Class mean	0.5219 ± 0.0000	0.8	0.8506 ± 0.0000	0.3	0.8309 ± 0.0000	0.3
Multi-prototype K=4	0.5344 ± 0.0022	3.1	0.8617 ± 0.0029	1.2	0.8613 ± 0.0010	1.2
Multi-prototype K=8	0.5380 ± 0.0015	6.2	0.8631 ± 0.0007	2.3	0.8695 ± 0.0007	2.3
FeTrIL (cross-entropy head)	0.5934 ± 0.0018	1.6	0.8564 ± 0.0033	0.6	0.8553 ± 0.0034	0.6
Streaming LDA	0.6433 ± 0.0000	16.8	0.8769 ± 0.0000	2.5	0.8842 ± 0.0000	2.5
Analytic readout	0.6486 ± 0.0000	16.8	0.8689 ± 0.0000	2.5	0.8815 ± 0.0000	2.5
Shared only	0.6367 ± 0.0024	16.8	0.8702 ± 0.0006	2.5	0.8580 ± 0.0041	2.5
Shared+LR8 (ours)	0.6853 ± 0.0028	23.0	0.8921 ± 0.0005	4.9	0.8942 ± 0.0010	4.9
Shared+LR32 (ours)	0.6934 ± 0.0008	41.8	0.8967 ± 0.0003	11.9	0.9027 ± 0.0007	11.9
Shared+LR32, uncentred prior (ours)	0.6995 ± 0.0029	41.8	0.8859 ± 0.0006	11.9	0.9001 ± 0.0004	11.9
within-class prior, ridge	0.7051 ± 0.0024	41.8	0.8917 ± 0.0009	11.9	0.9039 ± 0.0007	11.9
+ between-class scatter	0.6955 ± 0.0044	41.8	0.8853 ± 0.0008	11.9	0.9001 ± 0.0004	11.9
RanPAC M=2000	0.6361 ± 0.0009	31.6	0.8878 ± 0.0007	21.9	0.9015 ± 0.0016	21.9
RanPAC M=5000	0.6784 ± 0.0015	136.3	0.9005 ± 0.0009	111.9	0.9180 ± 0.0005	111.9
FeCAM (full covariance)	0.6610 ± 0.0000	1600.8	0.8903 ± 0.0000	225.3	0.9039 ± 0.0000	225.3

The complete comparator set on the donors. The donor suites above omit four comparators of Table 1; Table S23 adds them. None is both smaller and more accurate than our rank-32 memory in any of the six donor settings. RanPAC-10000 is more accurate everywhere, at 11–35× the storage.

Table S23: The four comparators missing from the donor suites, accuracy at 100 classes (mean over 3 class orders) with stored MB in parentheses, next to our rank-32 memory.

Donor	Shared+LR32 (ours)	Streaming LDA	Analytic readout	RanPAC-1000	RanPAC-10000
CIFAR (upsampled), ResNet-50	0.7506 (41.8)	0.6973 (16.8)	0.7235 (16.8)	0.6840 (12.0)	0.7589 (463.4)
CIFAR (upsampled), CLIP	0.7836 (11.9)	0.7456 (2.5)	0.7577 (2.5)	0.7498 (7.1)	0.8031 (414.6)
CIFAR (upsampled), DINOv2	0.9020 (11.9)	0.8808 (2.5)	0.8845 (2.5)	0.8800 (7.1)	0.9104 (414.6)
Food-101, ResNet-50	0.6934 (41.8)	0.6433 (16.8)	0.6486 (16.8)	0.5902 (12.0)	0.7010 (463.4)
Food-101, CLIP	0.8967 (11.9)	0.8769 (2.5)	0.8689 (2.5)	0.8745 (7.1)	0.9074 (414.6)
Food-101, DINOv2	0.9027 (11.9)	0.8842 (2.5)	0.8815 (2.5)	0.8829 (7.1)	0.9259 (414.6)

S28 Food-101 with backbones trained at native resolution

Every backbone we train elsewhere sees 32×32 images, and the donors are never trained on the base classes, so the two effects are confounded in the body of the paper. Here ResNet-18 with the standard ImageNet stem (7×7 stride-2 convolution and max-pool) is trained from scratch on the 20 base classes of Food-101 at 224 pixels with the recipe of §5 (train_base.py --dataset food101 --arch resnet18_im; closed-set 0.815), frozen, and run through the suite with the class orders of Appendix S27 (food101_scratch.sh). Table S24 gives every memory, ordered by storage. No row is both smaller and more accurate than our rank-32 memory. Food-101's training split is deliberately uncleaned (Bossard et al., 2014), which favours pooled estimators over per-class ones; CIFAR-100's is clean and shows the same ordering.

Table S24: Food-101, ResNet-18 trained from scratch at 224 pixels on 20 base classes, expanded to 100; accuracy at 100 classes, mean±std over 3 seeds. The indented rows decompose the prior as in Table 2. "—" marks an EC_{50}^{new} undefined on at least one seed (arriving-class accuracy below 0.50 from the first session).

Memory	MB	Acc@100	EC_{50}	EC_{50}^{new}
Class mean	0.20	0.2553 ± 0.0036	40.0	—
FeTrIL (cross-entropy head)	0.39	0.2549 ± 0.0028	38.1	32.5
Multi-prototype K=4	0.78	0.2451 ± 0.0013	37.4	—
Streaming LDA	1.20	0.3224 ± 0.0016	45.2	—
Analytic readout	1.20	0.2547 ± 0.0039	43.6	—
Shared only	1.20	0.2896 ± 0.0044	42.7	—
Multi-prototype K=8	1.56	0.2363 ± 0.0046	35.7	—
Shared+LR8 (ours)	2.76	0.3168 ± 0.0036	44.5	32.9
RanPAC M=1000	6.15	0.3244 ± 0.0026	48.6	36.8
Shared+LR32 (ours)	7.46	0.3421 ± 0.0052	47.7	38.4
within-class prior, ridge	7.46	0.3636 ± 0.0036	50.6	42.7
+ between-class scatter	7.46	0.3748 ± 0.0019	54.1	45.9
Shared+LR32, uncentred prior (ours)	7.46	0.3747 ± 0.0020	54.6	44.4
RanPAC M=2000	19.93	0.3449 ± 0.0012	50.1	40.9
FeCAM (full covariance)	100.20	0.3265 ± 0.0047	46.1	—
RanPAC M=5000	107.04	0.3589 ± 0.0009	50.8	43.9
RanPAC M=10000	404.82	0.3627 ± 0.0033	49.5	43.6

Depth behaves as on CIFAR-100 (Table S25): layer3 is worse on the base classes but hosts $+13.51 \pm 0.55$ points more at 100 classes under the rank-16 memory, and $+4.63 \pm 1.31$ under a class mean.

Table S25: Food-101, ResNet-18 trained at 224 pixels: the rank-16 memory at every tap, closed-set accuracy on the 20 base classes and accuracy at 100 classes, mean over 3 seeds.

Tap	Closed-set (20)	Acc@100
stem	0.3173	0.1400
layer1	0.4163	0.2036
layer2	0.6117	0.3462
layer3	0.7694	0.4660
layer4	0.8133	0.3309

An intervention on the base-class training. The between-class term hurts every donor, which the paper attributes to the donors never having been trained to separate the base classes. To test that causally we take the ImageNet ResNet-50, freeze everything up to layer3, and fine-tune layer4 with a linear head on Food-101's 20 base classes for 5 epochs (finetune_donor.py; nothing sees an arriving class), then freeze it again and rerun the decomposition. The between-class term moves from -0.96 ± 0.51 to $+0.43 \pm 0.20$, a change of $+1.39 \pm 0.56$ with the same sign on every class order, and our rank-32 memory rises from 0.6934 to 0.7199.

Is it the training recipe? The backbones above use the CIFAR augmentation, a 4-pixel reflect-pad crop, which at 224 pixels is almost no augmentation (training accuracy reaches 0.99). Retrained with random-resized crops (--aug rrc, scale $[0.25, 1]$), the backbones overfit less and reach a higher closed-set accuracy. Table S26 compares the two recipes: every finding holds, and the intermediate-layer advantage shrinks to about its CIFAR size, except the between-class term, which vanishes although this backbone still separates its base classes three times more than other classes (Appendix S29).

Table S26: Food-101 at 224 pixels, ResNet-18 trained from scratch with the CIFAR pad-crop or with random-resized-crop augmentation, 3 seeds each; differences in points, paired, with the 95% t half-width.

	Pad-crop	Random-resized-crop
Closed-set accuracy (20 classes)	0.815	0.854
Second-order: Shared+LR32 − class mean	$+8.68 \pm 1.87$	$+15.89 \pm 0.78$
Prototypes: K=8 − class mean	-1.89 ± 0.25	-0.76 ± 1.00
Shared+LR32 − full covariance	$+1.56 \pm 0.62$	$+1.77 \pm 0.35$
Uncentred prior − FeCAM's prior	$+3.27 \pm 1.24$	$+1.65 \pm 1.15$
Between-class term	$+1.12 \pm 1.13$	-0.09 ± 0.25
layer3 − layer4 (rank 16)	$+13.51 \pm 0.55$	$+6.53 \pm 1.88$
Dominated by a smaller, more accurate row	no	no

A control for the intervention. Fine-tuning changes a backbone in many ways; the claim is that training on the base classes is what makes their between-class scatter useful. We therefore also fine-tune layer4 on the last 20 classes of the arrival order (--classes heldout), which --holdout_classes 20 removes from the label space, and compare all three backbones on the same 80-class space. The base classes' between-class term is -1.11 ± 0.20 on the frozen donor, $+0.36 \pm 0.20$ after fine-tuning on the base classes, and -0.81 ± 0.34 after fine-tuning on the held-out classes: base minus held-out is $+1.17 \pm 0.17$, the same sign on every class order.

S29 Measuring base-class specialisation

The between-class term helps some backbones and hurts others. To state when without appealing to the outcome, we measure specialisation from the frozen features alone (specialisation.py), before relating it to the term. For a set of classes, let its between share be the trace of its between-class scatter over the trace of its total scatter, on training features in the memory's preprocessed space; the specialisation index is the base classes' between share over that of 20 other classes of the same dataset (the last 20 of the arrival order). An index near 1 means the base classes are no more separated than any others. A second, generic measure is the class-mean train-minus-test accuracy gap on the base classes.

Table S27 gives both for every setting in which we measured the term. The index is about 1 on every donor (0.99–1.06) and at least 1.72 on every backbone trained on the base classes, so any threshold between the two separates them; with one at 1.5 it predicts the term's sign in 14 of 15 settings, both arms of the intervention included, and it ranks the settings as the term does (Spearman 0.79 over settings, 0.77 over 47 seeds; the train–test gap gives 0.76). The exception is the Food-101 backbone trained with random-resized-crop: it separates its base classes about three times more than other classes, yet the term vanishes. Specialisation in this sense is therefore necessary for the gain but not sufficient, and what else the stronger augmentation changes is open.

Table S27: Specialisation index, class-mean train−test gap on the base classes, and the between-class term (points; within-class prior plus the base classes' between-class scatter, minus the within-class prior), mean over seeds. The last column says whether an index above 1.5 predicts a positive term. Fine-tuned donors are compared on the 80-class space of Appendix S28.

Setting	Index	Train−test gap	Between term	Sign right
CIFAR ResNet-18	2.67	+0.125	+1.81	yes
CIFAR VGG-16	3.28	+0.149	+1.36	yes
CIFAR ResNet-50	2.51	+0.144	+1.16	yes
Tiny ResNet-18	2.77	+0.219	+0.15	yes
300-class ResNet-18	1.72	+0.125	+1.29	yes
CIFAR donor ResNet-50	1.02	+0.006	−0.73	yes
CIFAR donor CLIP	1.02	+0.001	−0.73	yes
CIFAR donor DINOv2	1.00	+0.000	−0.42	yes
Food-101 donor ResNet-50	0.99	−0.047	−0.96	yes
Food-101 donor CLIP	1.01	−0.026	−0.64	yes
Food-101 donor DINOv2	1.06	−0.026	−0.38	yes
Food-101 ResNet-18 pad-crop	4.14	+0.185	+1.12	yes
Food-101 ResNet-18 rrc	3.00	+0.143	−0.09	no
Food-101 donor fine-tuned on base	2.05	+0.059	+0.36	yes
Food-101 donor fine-tuned on held-out	0.48	−0.041	−0.81	yes

S30 EC_τ as a curve, and where it stops measuring

EC_τ is reported at two thresholds in the main text. Sweeping τ shows how much of the apparent spread between memories is a property of the threshold (Figure S4). Across eleven memories (the class mean, K=4 prototypes, FeTrIL with its cross-entropy head as in Tables S31–S32, the shared term, ranks 8 and 32, full covariance and the four RanPAC widths), the gap between the best and worst EC_τ is 50.1 classes at τ=0.50, 19.7 at 0.60, 5.5 at 0.70 and 1.6 at 0.80. The spread peaks at τ=0.49, and the window in which the metric both separates the field by more than ten classes and has not saturated the benchmark is $\tau \in [0.50, 0.65]$. At τ=0.70, six of the eleven are within their own seed standard deviation of the best.

We report this because it bounds the claim of §5.2. "Capacity depends strongly on the readout" is a statement about the regime where the memory is the binding constraint. Above it the representation binds instead: EC_{70} lies between 29.0 and 34.5 (B+9 to B+14.5) for each of the eleven, which is the subsidy of §5.2 in its purest form — at that accuracy the label space served is essentially the base classes plus a handful. Below $\tau \approx 0.45$ everything reaches 100 and the benchmark binds. A 100-class benchmark and a 20-class backbone leave a window about 0.15 wide; a larger label space would widen it, which is the concrete cost of the scale limitation in §6.

$\mathrm{EC}_\tau^{\mathrm{new}}$ is narrower still. Because arriving-class accuracy is lower throughout, the metric is undefined for weak memories at every threshold where the strong ones are informative: at τ=0.50 it is defined on all five seeds for eight of eleven memories but never for a class-mean classifier, and at τ=0.70 on at most two seeds of any memory. It ranks the memories that work; it cannot rank the whole field. The decomposition we report ($\mathrm{EC}_{50}^{\mathrm{new}}$=65.8 ± 6.8 against 88.2 ± 3.1, macro-averaged per class; the curve here uses session-level arriving-class accuracy and reads 65.6 ± 7.1 at τ=0.50) is therefore a statement at one threshold, and we do not generalise it.

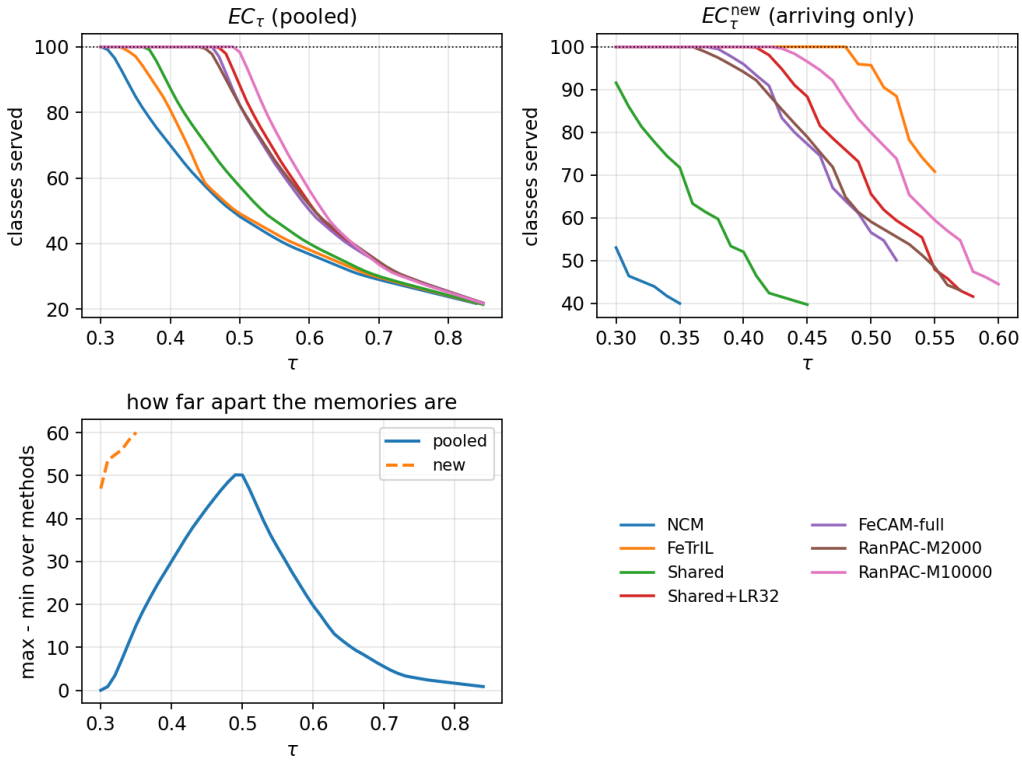


Figure S4: Top: EC_τ and EC_τ^{new} against τ, 5-seed means; curves stop where the metric becomes undefined for a seed. Bottom: the distance between the best and worst memory at each τ — the quantity that decides whether EC_τ can rank anything. Dotted line: the 100-class benchmark ceiling.

S31 Pooling, or sample count?

§5.4 attributes the shared covariance's benefit to statistical efficiency, which invites a sample-count reading: the shared term sees $B \times 500$=10,000 images where a class sees 500. That confounds two things. Pooling over B classes both multiplies the sample count and averages away class-specific deviation, and only the second requires the classes to be different. We separate them by restricting the shared term — and nothing else in the memory — to k base classes and m images of each, holding $k \times m$ fixed (Table S28).

The sample-count reading is wrong. Holding the class count at 20 and multiplying the images per class twenty-fold (500 → 10,000 samples) is worth $+0.40 \pm 0.08$ points; spreading the same 500 samples over 20 classes instead of 1 is worth $+2.18 \pm 1.04$, five times more, and consistent in sign on every seed. What the shared term needs is many classes, not many images.

This is the bias term of Proposition 1, not the variance term: averaging over B classes cancels Δ_c, and a single class's covariance — however well estimated — is a biased estimate of what the arriving classes share. It is also not a return to the "most structure is shared" reading that §5.4 rejects. Individual class covariances remain far apart (63% of each one's norm survives the best scalar fit of the shared term); it is their average that is the useful prior, and recovering an average needs many terms. Finally it explains the capacity scaling of §5.2 mechanistically: larger B raises capacity in part because it buys a better-centred shared prior, which is why the two curves have the same shape.

Table S28: The shared covariance estimated from k classes $\times$ m images, everything else unchanged; the per-class residual always sees all 500 images of its own class. Mean±std over 5 seeds, Acc@100. The three N=500 arms are matched on samples and differ only in how many classes supply them.

Shared term estimated from	N	Shared only	Shared+LR32
1 class × 500 images	500	0.3441 ± 0.0061	0.4531 ± 0.0075
4 classes × 125 images	500	0.3591 ± 0.0070	0.4660 ± 0.0074
20 classes × 25 images	500	0.3659 ± 0.0063	0.4705 ± 0.0067
20 classes × 500 images (default)	10,000	$\mathbf{0.3700 \pm 0.0057}$	$\mathbf{0.4760 \pm 0.0058}$
more classes, matched N (20×25 − 1×500)		$+2.18 \pm 1.04$	$+1.74 \pm 0.92$
more samples, matched classes (20×500 − 20×25)		$+0.40 \pm 0.08$	$+0.55 \pm 0.14$

S32 Does the shared term transfer to classes it was not estimated on?

Proposition 1 assumes $\mathbb{E}_c[\Delta_c]=0$, but Σ_{sh} is estimated on base classes and applied to arriving ones, so the assumption is being used off the set it holds on by construction. Measuring it directly (3 seeds, layer4): the mean arriving-class deviation is $\|\mathbb{E}[\Delta_c]\|/\|\Sigma_{\text{sh}}\| = 1.78 \pm 0.04$, so the assumption fails, and not by a little.

It fails almost entirely in scale. Arriving classes are more dispersed in the frozen embedding than the classes the backbone was trained on — their covariance trace is 2.31 ± 0.03 times larger — and once every covariance is put on a common trace the deviation drops to 0.359 ± 0.017. The per-class residual after the best scalar fit is the same on both sides (0.630 ± 0.011 base, 0.628 ± 0.022 arriving; computed per class against the pooled prior, so it agrees with the all-class 0.628 ± 0.018 of §5.4, which pools base and arriving classes), i.e. the shape of the mismatch transfers and only the size does not.

That is why the estimator survives an assumption it violates: the memory maps every covariance to a correlation matrix and normalises each class to unit mean variance before scoring (§4), so a common scale factor is divided out twice over and never reaches the Mahalanobis distance. The residual 36% shape mismatch is real and is what the rank-r term is for. We state this rather than leaving the proposition to be read as covering the arriving classes, which it does not.

S33 What the memory costs to fit and to run

The paper argues storage, and storage is the only axis on which we claim an advantage. For completeness, Table S29 gives the other two. Our memory is 13× smaller than full per-class covariance and 1.11× slower at inference; it is 54× smaller than RanPAC-10000 and 5.1× faster. Against a class mean it is 76× slower, which is the honest price of second-order scoring. Fitting is our worst axis: a rank-r SVD per arriving class costs 7.9s at 100 classes against 1.1s for a full covariance. Inference is implemented per class; computing zW once per test point and storing p_cW would cost $O(d^2 + Krd)$ instead of $O(Kd^2)$, so the inference figure is an upper bound for this memory.

These are wall-clock numbers from a shared GPU node and the seed spread is large (up to ±48% relative); they are indicative of ordering, not benchmarks. The asymptotic picture is cleaner: scoring is $O((r+1)d)$ per class against $O(d^2)$ for a full covariance, provided zW is formed once per test point at $O(d^2)$ and each p_cW is stored, so the measured near-parity reflects BLAS efficiency on a single dense matrix–vector product rather than arithmetic.

Table S29: Cost at K=100, mean±std over 5 seeds, CIFAR-100 base-20, shared A100 node. Inference is milliseconds per 1000 test images.

Memory	Storage (MB)	Fit (s)	Inference (ms/1k)
Nearest class mean	0.20	0.40 ± 0.18	$\mathbf{0.11 \pm 0.10}$
FeCAM-diag	0.39	0.17 ± 0.15	1.56 ± 0.77
Shared only	1.20	0.14 ± 0.12	6.83 ± 2.81
Shared+LR8	2.76	7.82 ± 4.63	8.31 ± 3.80
Shared+LR32 (ours)	7.46	7.90 ± 4.62	8.62 ± 4.14
RanPAC M=2000	19.93	0.58 ± 0.50	4.94 ± 2.17
FeCAM (full covariance)	100.20	1.10 ± 0.65	7.76 ± 3.62
RanPAC M=5000	107.04	1.04 ± 0.68	15.03 ± 6.95
RanPAC M=10000	404.82	2.33 ± 1.14	44.11 ± 20.46

S34 Does EC_τ rank anything that accuracy does not?

EC_τ inverts a monotone curve, so it cannot carry information the curve does not. The honest question is whether it orders memories differently from the two summaries already in use: accuracy at the final label space, and average incremental accuracy, the mean over sessions that has been the standard curve summary since Rebuffi et al. (2017). Table S30 puts all three side by side on the twenty memories of Table 1.

Table S30: AUC-EC against the two accuracy summaries, 5 seeds, CIFAR base-20. Average incremental accuracy is the mean of the seven session accuracies. Rank correlations: AUC-EC against average incremental accuracy $\rho = 0.992$; AUC-EC against Acc@100 $\rho = 0.977$; the two accuracy summaries against each other $\rho = 0.985$. Four of the twenty rows sit at a different rank under AUC-EC than under average incremental accuracy.

Memory	AUC-EC	Avg. incremental acc	Acc@100
Shared+LR32, uncentred prior (ours)	74.3	0.6617 ± 0.0117	0.5173
RanPAC-10000	66.8	0.6435 ± 0.0135	0.4973
RanPAC-5000	65.3	0.6416 ± 0.0142	0.4834
Shared+LR64 (ours)	64.8	0.6419 ± 0.0113	0.4890
Shared+LR32, refreshed (ours)	64.7	0.6423 ± 0.0126	0.4890
Shared+LR32 (ours)	61.3	0.6342 ± 0.0113	0.4760
RanPAC-2000	59.6	0.6293 ± 0.0130	0.4550
FeCAM	58.5	0.6282 ± 0.0108	0.4644
Shared+LR16 (ours)	57.3	0.6235 ± 0.0119	0.4582
RanPAC-1000	54.0	0.6119 ± 0.0116	0.4244
Shared+LR8 (ours)	53.5	0.6117 ± 0.0110	0.4382
Analytic readout (ACIL)	50.5	0.5973 ± 0.0120	0.3942
Linear probe (joint)	49.2	0.5968 ± 0.0134	0.4294
Streaming LDA	49.2	0.5948 ± 0.0117	0.4167
Shared only	44.2	0.5682 ± 0.0114	0.3700
FeTrIL	44.0	0.5704 ± 0.0118	0.3701
k-NN (k=5)	43.2	0.5626 ± 0.0155	0.3616
Associative prototypes	40.6	0.5407 ± 0.0132	0.3252
Multi-prototype (K=4)	40.0	0.5371 ± 0.0152	0.3238
Nearest class mean	39.6	0.5307 ± 0.0129	0.3125

The answer is: barely. At $\rho = 0.992$ against average incremental accuracy, EC_τ is close to an order-preserving reparametrisation of the curve, and we do not claim it as a discriminating statistic. What it is for is different. It reports a label-space size rather than a percentage, so a budget holder can read off how many classes a representation serves before a stated accuracy is lost; it forces the protocol conditions — threshold, memory, arrival order, benchmark ceiling — into the definition rather than into a caption; and it decomposes into base and arriving classes (§5.2), which is what makes the 65.8 versus 88.2 subsidy and the scaling in B of §5.2 statable at all. None of those is a claim that it re-ranks methods, and the reader

should not take the four rank changes as one: three of the four pairs involved are within a seed standard deviation of each other.

S35 A finite-range capacity model, and how far it can be tested

Proposition 1 explains covariance estimation; it says nothing about EC_τ itself. Proposition 2 is the smallest model that does; we use it as a finite-range diagnostic — a shape to test the measured curves against — not as a validated account of large-K behaviour. Its isotropy assumption does not hold after shared whitening: 63% of each class covariance's norm survives as residual (§5.4).

Proposition 2. Put K prototypes in a whitened embedding, score by nearest prototype, and let $\Delta_{\min}$ be the minimum pairwise separation in Mahalanobis units. A union bound over the $K-1$ competitors with a sub-Gaussian tail gives $P(\text{error}) \le (K-1)\exp(-\Delta_{\min}^2/8)$, and requiring accuracy τ gives

$$\mathrm{EC}_\tau \gtrsim 1 + (1-\tau)\, e^{\Delta_{\min}^2/8}. \tag{2}$$

Two predictions follow. P1: at a fixed representation and memory, $\mathrm{EC}_\tau - 1$ is proportional to $1-\tau$, with a slope that does not depend on τ. P2: across representations, log of that slope is linear in $\Delta_{\min}^2$ with coefficient $1/8$.

P1: affine, but not through the origin. Fitting EC_τ on a 0.01 grid of τ gives $R^2 > 0.94$, but this is not evidence, for two reasons. The grid is interpolated between seven measured sessions, so thirty-one "points" pseudo-replicate seven; and linearity is not diagnostic, because EC_τ inverts $A(n)$ and $A(n)$ is itself linear on this band at $R^2 \in [0.933, 0.994]$ — higher than the EC fit it was offered as evidence for. Any model with a locally linear accuracy curve passes that test.

We therefore fit on the measured sessions only, dropping the 100-class point where every curve is censored, and test the prediction that does discriminate: (2) puts the line through the origin, whereas a locally linear $A(n) = a - bn$ gives intercept $(a-1-b)/b$, free. Table S31 reports both. The affine fit is good ($R^2 \in [0.852, 0.909]$) and the intercept is negative for nine of twelve memories, with a magnitude that grows with capacity. Since the intercept is negative the measured curve lies below $1 + (1-\tau)e^{\Delta_{\min}^2/8}$. Because (2) is a sufficient condition, that is a shortfall against an idealised guarantee rather than a violated bound: real readouts do not reach what a model with isotropic whitened noise, known means and a fixed $\Delta_{\min}$ would permit, which is unsurprising since none of those three holds here — $\Delta_{\min}$ in particular is a minimum over a competitor set that grows with K. The equality therefore fails and the fitted slope cannot be read as $e^{\Delta_{\min}^2/8}$. We report the slope as a fitted quantity and attach no $\Delta_{\min}$ to it.

A summary that stays inside the data. Extrapolating the fitted line to τ=0 would give a single "capacity constant", but for the stronger memories that point lies 2–3× beyond the largest EC the benchmark can express, so the separation it reports would be pure extrapolation. Instead we integrate EC_τ over the window where it discriminates, $\tau \in [0.50, 0.65]$, every point of which is measured (Table S32). AUC-EC ranks the memories closely with accuracy (Spearman 0.977, Appendix S34) but not identically — RanPAC-2000 outranks full covariance on it while scoring lower at K=100 — and it is free of the threshold choice that §5.2 shows EC_{50} depends on.

P2 cannot be tested here, and that is informative. EC_τ is defined only where accuracy at the base session is at least τ, and informative only where it has not reached the benchmark ceiling. Those two bounds move in opposite directions with representation quality:

	stem	layer1	layer2	layer3	layer4
τ where EC_τ is measurable	[.20, .36]	[.28, .49]	[.41, .64]	[.53, .79]	[.47, .80]

The intersection over the five taps is empty. Any regression of the fitted slope on $\Delta^2_{\min}$ across representations therefore compares slopes fitted on disjoint τ ranges. We ran it anyway and obtained a coefficient of -0.096 against the predicted $+0.125$ — a sign reversal we attribute to that truncation rather than to the model, because the ordering it produces (stem highest, layer4 lowest) is exactly what truncation predicts: a weak representation is measured over its whole range, a strong one only near its ceiling. We therefore report P2 as not testable on a 100-class benchmark rather than as falsified. This is the sharpest form of the scale limitation in §6: the ceiling does not merely weaken the scaling claims, it removes the window in which the theory's central prediction could be checked at all.

Table S31: P1, fitted on the six measured sessions below 100 classes (the 100-class point dropped as censored), 5 seeds. (2) predicts intercept 0; the interval is the paired 95% CI over seeds. Nine of twelve exclude zero, and the offset grows with capacity.

Memory	slope	intercept	95% CI	through origin?
Nearest class mean	109.5	-1.75	$[-4.4, +0.9]$	yes
Multi-prototype K=4	111.2	-1.86	$[-4.2, +0.5]$	yes
FeTrIL (CE head)	112.4	-1.49	$[-4.9, +2.0]$	yes
Shared only	123.4	-3.55	$[-7.071, -0.029]$	no
Shared+LR8	143.1	-5.78	$[-9.7, -1.9]$	no
RanPAC M=1000	144.4	-5.90	$[-10.2, -1.6]$	no
Shared+LR16	150.3	-6.79	$[-11.0, -2.6]$	no
FeCAM (full cov.)	151.8	-6.66	$[-10.9, -2.4]$	no
RanPAC M=2000	155.3	-7.41	$[-11.9, -2.9]$	no
Shared+LR32 (ours)	156.6	-7.53	$[-11.9, -3.1]$	no
RanPAC M=5000	164.4	-8.94	$[-14.2, -3.7]$	no
RanPAC M=10000	165.6	-9.35	$[-14.4, -4.3]$	no

Table S32: AUC-EC: EC_τ averaged over $\tau \in [0.50, 0.65]$, the window in which the metric discriminates, mean±std over 5 seeds. Values are interpolated between measured sessions and none is extrapolated past the benchmark, but RanPAC-10000 reaches the 100-class ceiling at τ=0.50 on two of five seeds, so its entry is a censored mean. FeTrIL is the cross-entropy head here, as in Table S31; Table 1 reports the published SVM head (AUC-EC 44.0 ± 0.9).

Memory	AUC-EC	Memory (MB)
RanPAC M=10000	$\mathbf{66.8 \pm 3.7}$	404.82
RanPAC M=5000	65.3 ± 4.2	107.04
Shared+LR32 (ours)	61.3 ± 3.3	7.46
RanPAC M=2000	59.6 ± 3.4	19.93
FeCAM (full cov.)	58.5 ± 3.2	100.20
Shared+LR16 (ours)	57.3 ± 3.3	4.33
RanPAC M=1000	54.0 ± 2.7	6.15
Shared+LR8 (ours)	53.5 ± 2.7	2.76
Shared only	44.2 ± 1.7	1.20
FeTrIL (CE head)	40.8 ± 1.4	0.39
Multi-prototype K=4	40.0 ± 1.7	0.78
Nearest class mean	39.6 ± 1.3	0.20

S36 Can base classes be chosen for future capacity?

Base sets of the same size differ by up to 2.35× in the capacity they later deliver (§5.2) — though the draws differ in arrival order as well as membership, and Appendix S37 attributes about a third of that to order at B=5 and essentially all of it at B=20. What remains at small B is worth more than most of the memory choices in this paper, so we asked whether it is usable: is there a statistic, computable from the candidate classes alone before the backbone is trained, that ranks base sets by the capacity they will eventually deliver?

We tested seven, on the seven independently drawn base sets at each $B \in \{5, 10, 20\}$: mean, minimum and spread of prototype separation; pooled effective rank; and three derived from this paper's own mechanism, since §5.4 says the shared prior improves by averaging away class-specific deviation — the mean and maximum shift of the pooled covariance when any one class is dropped, and the trace-normalised diversity of the class covariances. A base set whose pooled covariance has converged should be a better prior.

None of them works. Within a fixed B every Spearman correlation with $\mathrm{EC}_{50}^{\mathrm{new}}$ falls in $[-0.750, +0.643]$, short of the 0.786 needed for significance at n=7 but not by much, and the sign of the same statistic flips across B for five of the seven — min_sep is $+0.643, -0.500, +0.286$ at $B = 5, 10, 20$, and cov_diversity $+0.071, -0.750, +0.357$.

A leave-one-out selection looks better than that and is not. Choosing the best of the fourteen rules (seven statistics × two signs) reaches 33.14 against a random base set's 21.79 at B=5. But running the identical procedure on randomly permuted capacities reaches 33.84 on average, and the permutation p-values are 0.981, 0.838 and 0.847 (Table S33). The apparent gain is the search over rules, not the geometry. We report the control because without it this table would have read as a positive result.

Appendix S37 supplies the mechanism. The draws differ in arrival order as well as membership, and that term accounts for about a third of the variance at B=5 and B=10 and essentially all of it at B=20. A selection rule sees only the base classes; at larger B the quantity it would have to predict is mostly a property of a sequence it cannot observe. The headroom at small B is real and we cannot reach it; the headroom at B=20 is largely not there. We state the bar so future work can clear it: a usable rule must beat its own permutation null, and the target must first be shown to carry base-set signal at the B in question.

Table S33: Selecting base classes for future capacity. Capacity is $\mathrm{EC}_{50}^{\mathrm{new}}$ with our memory, computed from per-class accuracy on the arriving classes, and agreeing with Appendix S5 row for row; "best rule" is the best of fourteen leave-one-out selection rules, and "shuffled" applies the identical procedure to permuted capacities (2000 draws).

B	random	oracle	spread	best rule	shuffled	p
5	21.79	36.11	2.35×	33.14	33.84	0.981
10	42.01	52.43	1.79×	49.31	50.61	0.838
20	62.08	73.56	1.28×	71.29	70.81	0.847

S37 What the seed spread actually measures

A seed changes the backbone, the data order and which classes form the base set, so the reported spread mixes those sources. EC_{τ} additionally depends on the order in which the remaining classes arrive, and that source can be isolated: hold the seed-0 backbone and its base set fixed and permute the 80 arriving classes eight ways.

Accuracy at K=100 has zero order-only spread (sd 0.0000 for every memory and tap tried; the four arrival strategies of Appendix S24 differ at K=100, but each has its own backbone, so that spread is not order-only) — at the final session all classes are present whatever the order, and the memories are order-independent given the same data. Every claim we state in accuracy is therefore untouched by this, including the depth result: across six permutations of the same kind (run separately at each tap) layer3 beats layer4 by +7.21 points with sd 0.00. EC_{τ} is a different matter:

	order-only sd (n=8)	seed sd (n=5)	share of variance
Nearest class mean, EC_{50}	0.88	1.79	24%
FeCAM (full cov.), EC_{50}	2.96	3.67	65%
Shared+LR32 (ours), EC_{50}	2.76	3.10	80%
Shared+LR32 (ours), $\mathrm{EC}_{50}^{\mathrm{new}}$	5.97	6.84	**76**%

The last row is the one that matters for the abstract, which quotes $\mathrm{EC}_{50}^{\mathrm{new}} = 65.8 \pm 6.8$: about three quarters of that variance is the arrival sequence. Every ± on an EC_τ in this paper is therefore already a combined interval — each seed draws its own base set, initialisation and arrival order, so the spread contains all three, in the proportions this table and Appendix S22 estimate — and the interpolation between measured sessions adds to it only where a curve crosses τ between two sessions, which Appendix S35 bounds. It does not affect the paired comparisons — every method sees the identical order within a seed, so the order term cancels in the difference, which is why the intervals in Appendix S23 are so much tighter than the marginal spreads. But a single EC_τ with its error bar should be read as "this representation, this memory, under a random arrival order". The shares are point estimates from n=8 orders against n=5 seeds and carry wide intervals — an F-based 95% interval on the Shared+LR32 EC_{50} ratio spans roughly 10% to 440% — so they establish that order is a large component of the spread, not its exact size.

It also confounds the base-set variation. The seven base sets drawn at each B (§5.2, Appendix S36) differ in their arrival order as well as in their membership — consecutive draws share about one class in ten in the first session — so the spread attributed to the base set contains this term. Measuring order-only spread on two independent base sets per B, eight permutations each:

B	order only, set A	set B	across the 7 draws	
5	3.66	5.35	7.30	order is 38% of the variance
10	4.46	5.32	8.84	order is 31%
20	5.97	6.53	5.78	order alone exceeds the spread

At B=20 permuting arrivals at a fixed base set produces more variance than the seven different base sets do, so no base-set effect is detectable there at all; the 1.28× we report is inside ordering noise. At B=5 and B=10 a base-set effect survives — roughly 60–70% of the variance — but it is smaller than the raw ratios suggest. This also explains the selection null of Appendix S36 mechanistically: at larger B the quantity a selection rule would have to predict is mostly a property of the arrival sequence, which no statistic of the base classes can see.

S38 Should the shared covariance be updated?

§4 estimates Σ_{sh} once, on the base classes, and never updates it. That is a design choice, not a constraint: accumulating the pooled covariance across sessions is protocol-legal, since it is a $d \times d$ statistic rather than retained features. What is not legal is refitting the residuals of earlier classes — they are defined in the old whitened space, and refitting needs their features again. A refreshed shared-whitening memory therefore necessarily holds residuals fitted against several different whitenings. We measured that rather than assuming it is harmless.

Refreshing helps, and by more than the mismatch costs (5 seeds, paired):

	estimated once	refreshed each session	Δ
Shared only	0.3700	0.3954	$+2.55 \pm 0.48$
Shared+LR32	0.4760	0.4890	$+1.31 \pm 0.22$

Both are sign-consistent on every seed. The refreshed variant must keep the running scatter, a $d \times d$ statistic, and derives W from it by one Cholesky factorisation when scoring, so one d^2 term is stored either way (1.20 and 7.46 MB); caching W as well would add 1.05 MB. Fitting costs 6% more. Against full per-class covariance the refreshed memory reaches $+2.46 \pm 0.61$ points at 7.4% of its storage, against $+1.15 \pm 0.44$ for the version we report — so the CIFAR-100 comparison that Appendix S23 marks supported-but-marginal (class-level lower bound +0.001) would be comfortably supported under the refreshed memory. We flag that rather than claim it: the entry in Table 1 is the marginal one, and it is the one to argue with.

Refreshing does not disturb the depth result: at rank 16 the layer3−layer4 gap is +6.29±2.72 once and $+5.04 \pm 2.26$ refreshed, sign-consistent either way. It narrows because the deeper tap gains more from an updated prior (+1.54 at layer4 against +0.28 at layer3), which is what the account of §5.6 predicts — a layer specialised to 20 classes has more to gain from statistics of the other 80.

We report the once-only variant throughout anyway, because it is the variant every other number in this paper was measured with and re-running the suite would not change any conclusion it supports. The consequence is that our headline numbers understate the memory by about a point, and that "estimate the prior before expansion and freeze it" is a simplification we adopted rather than a property the method needs.

S39 A stronger shared prior: the analytic readout's statistic

Table S17 shows the analytic readout beating our shared Mahalanobis term at the same d^2 budget. We asked whether the per-class residual stacks on that readout's statistic, and changed only the whitening. FeCAM's prior, used everywhere else in the paper, is the base-class covariance (each class centred on its own mean), shrunk with $\gamma_1{=}\gamma_2{=}1$ and correlation-normalised. The analytic readout's statistic is the uncentred autocorrelation of the base-class features, $R \propto \sum zz^\top$; we add a ridge of $\lambda\,\mathrm{mean}(\mathrm{diag}\,R)$ with $\lambda{=}10^{-2}$, a value fixed before any of these runs. The residual, the score (1) and the storage, $d^2 + K(d + rd + r + 1)$, are unchanged. Differences below are paired over seeds, with the half-width of the 95% t-interval.

Table S34: The shared prior, CIFAR-100, ResNet-18, 5 seeds, rank-32 residuals unless stated, all at the storage of the corresponding Table 1 row. Centring and the regulariser are varied separately; both matter.

Prior	Centring	Regulariser	Acc@100	Δ (points)
FeCAM's (headline)	per class	shrink + corr.-norm.	0.4760 ± 0.0058	—
Uncentred, FeCAM's reg.	none	shrink + corr.-norm.	0.4913 ± 0.0058	$+1.53 \pm 0.22$
Centred, ridge	per class	ridge 10^{-2}	0.4963 ± 0.0066	$+2.03 \pm 0.13$
Uncentred, ridge	none	ridge 10^{-2}	0.5173 ± 0.0058	$+4.13 \pm 0.22$
Uncentred ridge prior at smaller budgets				
Shared only (1.20 MB)	none	ridge 10^{-2}	0.4486 ± 0.0093	
Rank 8 (2.76 MB)	none	ridge 10^{-2}	0.5002 ± 0.0067	
Rank 16 (4.33 MB)	none	ridge 10^{-2}	0.5083 ± 0.0065	

Four things follow. The gain is in the statistic, not the solve. With no residual, the uncentred prior used as a Mahalanobis term scores 0.4486 against the analytic readout's 0.3942 at the same 1.20 MB, so the ordering of Table S17 belongs to FeCAM's prior, not to generative scoring. The residual still pays: rank 32 adds +6.9 points over that shared term. The choice survives held-out selection (below): the four priors of Table S34 were first read on the test split, and the held-out fifth picks the same one; the other two alternatives to FeCAM's are also above the headline. The ridge is not a knife edge: over $\lambda \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1\}$ rank 32 reaches 0.5115, 0.5125, 0.5173, 0.5107 and 0.4728, so three orders of magnitude lie within 0.7 points of the value we fixed, and only $\lambda{=}1$, which washes the prior out towards isotropy, falls below the headline.

Does it transfer? On every backbone we trained, yes; on pretrained ViTs, no (Table S35). It removes the one VGG-16 domination of Appendix S2 and at 300 classes widens our lead over RanPAC-10000 to 5.5 points at 1/20 of its storage. The two ViT donors lose 0.5 to 0.7 points; Table 2 (§5.4) traces this to the between-class term. We therefore keep FeCAM's prior as the headline — every mechanism study in the paper was run with it — and report this variant as a stronger configuration of the same memory rather than as the method.

Table S35: Uncentred ridge prior against FeCAM's, rank 32, same storage, 3 seeds per representation. Δ is paired, with the 95% t half-width; every difference has the same sign on every seed.

Representation	FeCAM's prior	Uncentred ridge	Δ (points)
VGG-16-BN, CIFAR-100	0.3299	0.3488	$+1.89 \pm 0.96$
ResNet-50, CIFAR-100	0.5405	0.5631	$+2.26 \pm 0.48$
ResNet-18, Tiny-ImageNet	0.3366	0.3578	$+2.12 \pm 0.30$
ResNet-18, 300-class union	0.3561	0.3993	$+4.33 \pm 0.55$
ResNet-50 (ImageNet-1k donor)	0.7506	0.7570	$+0.63 \pm 0.29$
CLIP ViT-B/16 donor	0.7836	0.7784	-0.52 ± 0.15
DINOv2 ViT-B/14 donor	0.9020	0.8954	-0.66 ± 0.22

Score fusion is not a substitute: summing the z-scored scores of the analytic readout and our headline memory gives 0.4177, 5.83 ± 0.42 points below the memory alone. One numerical note: at d=2048 the uncentred prior is ill-conditioned enough that the float32 inverse overflowed on one ResNet-50 seed; the code then recomputes the whitening, and if needed the residual SVD, in float64. The fallback fires only when the float32 computation fails, so no run that succeeded in float32 changes.

What in the uncentred statistic carries the gain? The base-class autocorrelation decomposes as $R \approx \Sigma_{\text{within}} + \Sigma_{\text{between}} + \mu\mu^\top$, with μ the global mean of the base features. Table 2 (in §5.4) adds the terms one at a time, every prior with the same ridge (10^{-2}) and the same rank-32 residual: the within-class scatter alone; plus $\mu\mu^\top$; plus the between-class scatter (the covariance centred on the global mean); and all three (the uncentred prior). The mean contributes nothing anywhere. The between-class scatter carries the gain on every backbone of this table trained on the base classes (all with the CIFAR pad-crop recipe), and it lands on the arriving classes while costing the base classes; on the three pretrained donors, which were never trained to separate the base classes, the same term hurts. Whitening by it discounts the directions a backbone spent separating its training classes — useful exactly when those directions were learned at the arriving classes' expense.

Chosen on held-out data. The four priors of Table S34 and the five ridge strengths were also selected on the held-out fifth of the training data of Appendix S42. The held-out fifth picks the uncentred prior on 5 of 5 seeds and λ=10^{-2} on 5 of 5, so the test-read choice is also the held-out one: the test accuracy of the held-out choice is 0.5173, against 0.5173 for the test-read one.

The paper's controls under the uncentred prior. The pooling control of Appendix S31 reproduces: at matched samples, spreading the shared term over 20 classes rather than 1 is worth $+6.92 \pm 0.95$ points for the shared term alone, against $+2.04 \pm 0.31$ for twentyfold more images at fixed classes (FeCAM's prior: +2.18 and +0.40), so the prior still needs many classes, not many images. The per-class low-rank store of Table S11 now trails the residual by $+12.05 \pm 0.59$ points (FeCAM's prior: +7.93), and the shared term alone recovers 90% of the full-covariance gain (38% under FeCAM's). Sample efficiency changes at the extremes: with n images per arriving class, the best of ranks 8 and 32 on the uncentred prior beats the FeCAM-prior memory by +1.73, +2.49 and +2.41 points at $n = 25, 50, 100$ but trails it by 2.03 at n=10, and the useful rank stays at 8 up to n=50.

Whose between-class scatter? Table 2 shows the base classes' between-class scatter helping on backbones trained on those classes, but the donors differ from our backbones in more than that. A within-backbone test removes the confound: we take 20 classes out of the label space entirely (the last 20 of the arrival order), expand over the remaining 80, and whiten with the within-class scatter of the base classes plus the between-class scatter of either the 20 base classes or the 20 removed classes, on the same scale and with the same ridge (--holdout_classes, Table S36). If the gain came from discounting class-separating directions in general, the two would match; if it comes from discounting the directions the backbone was trained to separate, the base classes' scatter should help more. On the backbones we

trained, the base classes' scatter helps more on 4 of 4 (sign-consistent over seeds on 4); on CIFAR-100 it adds $+1.91 \pm 0.29$ points against $+0.41 \pm 0.19$ for the held-out classes'. On the pretrained donors neither kind of scatter helps (ResNet-50 -0.71 / -0.13, CLIP ViT-B/16 -0.84 / -0.31, DINOv2 ViT-B/14 -0.41 / -0.01).

Table S36: Whitening with the between-class scatter of the 20 trained base classes or of 20 classes the backbone never trained on (removed from the label space), rank 32, ridge 10^{-2}, accuracy on the remaining 80 classes. Differences from the within-class prior in points, paired, with the 95% t half-width; seeds in parentheses.

Representation	Within	+base between	+held-out between	Base − held-out
CIFAR-100 / ResNet-18 (5)	0.5366	$+1.91 \pm 0.29$	$+0.41 \pm 0.19$	$+1.50 \pm 0.31$
CIFAR-100 / VGG-16 (3)	0.3824	$+1.38 \pm 1.11$	$+0.56 \pm 0.43$	$+0.82 \pm 0.69$
CIFAR-100 / ResNet-50 (3)	0.5833	$+1.07 \pm 0.89$	$+0.17 \pm 0.67$	$+0.91 \pm 0.26$
Tiny-ImageNet / ResNet-18 (3)	0.3657	$+0.09 \pm 0.52$	-0.38 ± 0.61	$+0.46 \pm 0.86$
ResNet-50 (ImageNet-1k) (3)	0.7803	-0.71 ± 0.46	-0.13 ± 0.47	-0.58 ± 0.50
CLIP ViT-B/16 (3)	0.8037	-0.84 ± 0.39	-0.31 ± 0.39	-0.53 ± 0.53
DINOv2 ViT-B/14 (3)	0.9118	-0.41 ± 0.07	-0.01 ± 0.11	-0.40 ± 0.05

Is it the size of the term? The base classes' between-class scatter is larger than the held-out classes' (its trace is 2.9–3.8× theirs on CIFAR-100, between_trace.py), so the control above could reflect magnitude rather than direction. Table S37 removes that: the held-out classes' scatter is rescaled to the base classes' trace, or given the base classes' eigenvalues on its own eigenvectors, and a third arm uses random orthonormal directions with the base spectrum. Matching the size recovers almost none of the gap, and random directions of the same energy do nothing, so what helps is the trained classes' directions. Pooled over seeds, the between-class term is positive on 17 of 17 seeds of the five pad-crop backbones of Table 2 and negative on 9 of 9 donor seeds (one-sided sign tests $p = 7.6 \times 10^{-6}$ and 0.002); adding the Food-101 backbones trained from scratch, it is positive on 21 of 23 seeds of backbones trained on the base classes ($p = 3.3 \times 10^{-5}$), the two exceptions under stronger augmentation.

Table S37: The held-out-classes control with the magnitude confound removed, rank 32, ridge 10^{-2}, 80-class label space. Differences from the within-class prior in points, paired, with the 95% t half-width; seeds in parentheses. "Trace" rescales the held-out classes' between-class scatter to the base classes' trace; "spectrum" keeps its eigenvectors with the base eigenvalues; "random" uses random directions with the base eigenvalues.

Representation	+base	+held-out	trace	spectrum	random	base − trace
CIFAR-100 / ResNet-18 (5)	$+1.91 \pm 0.29$	$+0.41 \pm 0.19$	$+0.48 \pm 0.19$	$+0.27 \pm 0.21$	-0.01 ± 0.15	$+1.43 \pm 0.38$
CIFAR-100 / VGG-16 (3)	$+1.38 \pm 1.11$	$+0.56 \pm 0.43$	$+0.61 \pm 0.43$	$+0.57 \pm 0.27$	$+0.05 \pm 0.57$	$+0.77 \pm 0.73$
CIFAR-100 / ResNet-50 (3)	$+1.07 \pm 0.89$	$+0.17 \pm 0.67$	$+0.17 \pm 0.59$	$+0.17 \pm 0.55$	$+0.28 \pm 0.42$	$+0.91 \pm 0.37$
Tiny-ImageNet / ResNet-18 (3)	$+0.09 \pm 0.52$	-0.38 ± 0.61	-0.62 ± 0.50	-0.71 ± 0.79	$+0.01 \pm 0.23$	$+0.71 \pm 0.89$

Is it calibration? The between-class term moves accuracy from the base classes to the arriving ones, which a simple bias correction against the well-served base classes would also do. We test that directly: the base classes' scores are multiplied by $1+\kappa$, $\kappa \in \{0, 0.01, 0.02, 0.05, 0.1, 0.2\}$ (a penalty, since scores are negative distances), with κ chosen on the held-out fifth of Appendix S42 (valselect.py, family calib). Table S38 shows that calibration recovers almost none of the gain, even with κ read on the test split, and adds nothing once the between-class scatter is in the prior: the two are not the same correction.

Table S38: A base-class score penalty against the between-class term, rank 32, ridge 10^{-2}, paired differences in points with the 95% t half-width; seeds in parentheses. "Oracle" picks κ on the test split.

Backbone	Penalty (held-out)	Penalty (oracle)	Between-class term	Between − oracle penalty	Penalty on top of between
CIFAR-100 / ResNet-18 (5)	$+0.01 \pm 0.07$	$+0.03$	$+1.81 \pm 0.23$	$+1.78 \pm 0.21$	-0.05 ± 0.15
CIFAR-100 / VGG-16 (3)	$+0.25 \pm 0.30$	$+0.43$	$+1.36 \pm 0.48$	$+0.93 \pm 0.78$	$+0.08 \pm 0.22$
CIFAR-100 / ResNet-50 (3)	$+0.00 \pm 0.00$	$+0.00$	$+1.16 \pm 0.10$	$+1.16 \pm 0.10$	$+0.00 \pm 0.00$

Does it fade as the backbone sees more base classes? If the discounted directions are over-represented because the backbone was trained on few classes, the term should shrink as B grows. Table S39 does not show that: the term is positive at every B and smallest at B=50, but the trend over four points is weak (Spearman -0.40), so the prediction is not established.

Table S39: The between-class term (within-class prior plus the base classes' between-class scatter, minus the within-class prior) and the uncentred prior's gain over FeCAM's, against the number of base classes; ResNet-18, CIFAR-100, points, paired 95% half-width; seeds in parentheses.

Base classes	Between-class term	Uncentred − FeCAM's prior
$B = 5$ (3)	$+1.67 \pm 0.64$	$+3.04 \pm 1.43$
$B = 10$ (3)	$+1.54 \pm 0.78$	$+3.39 \pm 1.12$
$B = 20$ (5)	$+1.81 \pm 0.23$	$+4.13 \pm 0.22$
$B = 50$ (3)	$+1.06 \pm 0.50$	$+2.39 \pm 0.39$

Are the baselines tuned as hard? The uncentred prior's ridge was confirmed on the held-out fifth, so we gave the baselines the same search on the same split (valselect.py, families fecam_gamma, fecam_ridge, shared_gamma). Full covariance does not improve: over $\gamma_1{=}\gamma_2 \in \{0.03, \ldots, 3\}$ the held-out fifth picks FeCAM's published $\gamma{=}1$ on 4 of 5 seeds, for a test accuracy of 0.4645 against 0.4644, and a pure ridge in its place is far worse (0.3802). The uncentred variant therefore leads the tuned full covariance by $+5.27 \pm 0.49$ points. The shared-only row does improve: FeCAM's constants over-shrink a covariance pooled from 10,000 samples, and a held-out γ lifts it from 0.3700 to 0.3880, which narrows the analytic readout's lead over it on CIFAR to $+0.63 \pm 0.57$ points. Under the uncentred prior the rank-32 memory reaches $\mathrm{EC}_{50}^{\mathrm{new}} = 91.0 \pm 3.8$ on the arriving classes alone, against 65.8 ± 6.8 for the headline.

S40 Components inherited from FeCAM

Three pieces of the memory come from FeCAM with its published constants and were never tuned here: the power transform, the residual floor's γ_1 (§4), and the prior's shrinkage. Table S40 removes each, rank 32, 5 seeds, paired against the unablated memory with the same prior. The power transform is load-bearing (-4.39 ± 0.30 points without it), and the per-class scale normalisation of §4 more so under the uncentred prior: without it that memory falls by -34.30 ± 5.67 points. The floor matters in both directions: dropping it costs the uncentred prior -5.04 ± 0.70, but a smaller floor ($\gamma_1{=}0.1$) lifts the FeCAM-prior memory by $+2.31 \pm 0.10$ while lowering the uncentred one (-1.07 ± 0.17). We report the published constant rather than this test-read improvement, so the FeCAM-prior headline is conservative in one more way. Of the three changes the ridge makes to FeCAM's prior, only the shrinkage strength matters: dropping the off-diagonal shrinkage (-0.14 ± 0.05) or the correlation normalisation (-0.00 ± 0.08) changes almost nothing, and a weaker shrinkage alone gives $+1.23 \pm 0.23$ of the full swap's $+2.03 \pm 0.13$.

Table S40: Ablations of the components inherited from FeCAM, CIFAR-100 ResNet-18, rank 32, 5 seeds. Δ is paired against the unablated memory with the same prior, in points, with the 95% t half-width.

Prior	Ablation	Acc@100	Δ
FeCAM's	floor γ_1=0	0.4644	-1.16 ± 0.51
FeCAM's	floor γ_1=0.1	0.4991	$+2.31 \pm 0.10$
uncentred	floor γ_1=0	0.4669	-5.04 ± 0.70
uncentred	floor γ_1=0.1	0.5065	-1.07 ± 0.17
FeCAM's	no power transform	0.4321	-4.39 ± 0.30
uncentred	no power transform	0.4916	-2.57 ± 0.56
FeCAM's	prior shrinkage γ=0.01	0.4882	$+1.23 \pm 0.23$
FeCAM's	no off-diagonal shrinkage (γ_2=0)	0.4746	-0.14 ± 0.05
FeCAM's	no correlation normalisation	0.4759	-0.00 ± 0.08

S41 Which dominations are significant?

Figure 1 and the frontier tables call a row dominated when a smaller row is more accurate on the mean. Storage is deterministic, so only the accuracy side needs a test: Table S41 gives, for every domination that involves one of our memories, the paired accuracy difference over the seeds both rows share, whether it has the same sign on every seed, and whether the 95% paired t-interval excludes zero (frontier_sig.py). Of the 33 dominations by one of our memories, 30 have their interval above zero; the exceptions are Shared+LR32, refreshed over Shared+LR64 on ResNet-18 ($+0.01 \pm 0.33$); Shared+LR16 over RanPAC-2000 on ResNet-18 ($+0.32 \pm 0.32$); Shared+LR32 over full covariance on ResNet-50 ($+0.21 \pm 0.91$). The one domination of our headline memory, by RanPAC-1000 on VGG-16, does not: it is $+0.74 \pm 1.12$ points on three seeds, so VGG-16 is an exception on the mean, not a demonstrated one.

Table S41: Dominations involving our memories, paired over seeds (5 on CIFAR-100 ResNet-18, 3 elsewhere). Δ is the dominating row's accuracy minus the dominated row's, in points, with the 95% t half-width. Upper-bound rows (k-NN, joint probe) are excluded.

Setting	Smaller and more accurate	Dominated	Δ	Every seed	CI > 0
CIFAR-100 / ResNet-18	Shared+LR32, uncentred prior	RanPAC-10000	$+2.00 \pm 0.45$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32, uncentred prior	Shared+LR64	$+2.83 \pm 0.24$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32, uncentred prior	RanPAC-5000	$+3.38 \pm 0.48$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32, uncentred prior	FeCAM	$+5.28 \pm 0.48$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32, uncentred prior	RanPAC-2000	$+6.23 \pm 0.48$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32, refreshed	Shared+LR64	$+0.01 \pm 0.33$	no	no
CIFAR-100 / ResNet-18	Shared+LR32, refreshed	RanPAC-5000	$+0.56 \pm 0.44$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32, refreshed	FeCAM	$+2.46 \pm 0.61$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32, refreshed	RanPAC-2000	$+3.40 \pm 0.62$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR64	RanPAC-5000	$+0.55 \pm 0.45$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR64	FeCAM	$+2.45 \pm 0.38$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR64	RanPAC-2000	$+3.40 \pm 0.38$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32	FeCAM	$+1.15 \pm 0.44$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR32	RanPAC-2000	$+2.10 \pm 0.44$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR16	RanPAC-2000	$+0.32 \pm 0.32$	no	no
CIFAR-100 / ResNet-18	Shared+LR16	RanPAC-1000	$+3.38 \pm 0.26$	yes	yes
CIFAR-100 / ResNet-18	Shared+LR8	RanPAC-1000	$+1.38 \pm 0.39$	yes	yes
CIFAR-100 / VGG-16	RanPAC-1000	Shared+LR32	$+0.74 \pm 1.12$	yes	no
CIFAR-100 / VGG-16	RanPAC-1000	Shared+LR64	$+0.24 \pm 1.17$	no	no
CIFAR-100 / VGG-16	Shared+LR32, uncentred prior	FeCAM (full cov.)	$+1.31 \pm 0.65$	yes	yes
CIFAR-100 / VGG-16	Shared+LR32, uncentred prior	Shared+LR64	$+1.39 \pm 1.09$	yes	yes
CIFAR-100 / ResNet-50	Shared+LR32	FeCAM (full cov.)	$+0.21 \pm 0.91$	no	no
CIFAR-100 / ResNet-50	Shared+LR32	RanPAC-5000	$+1.76 \pm 0.56$	yes	yes
CIFAR-100 / ResNet-50	ACIL	RanPAC-2000	$+0.83 \pm 1.09$	yes	no
CIFAR-100 / ResNet-50	Shared+LR32, uncentred prior	FeCAM (full cov.)	$+2.47 \pm 0.43$	yes	yes
CIFAR-100 / ResNet-50	Shared+LR32, uncentred prior	RanPAC-5000	$+4.01 \pm 0.54$	yes	yes
CIFAR-100 / ResNet-50	Shared+LR32, uncentred prior	RanPAC-10000	$+2.19 \pm 0.44$	yes	yes
300-class union	Shared+LR8	RanPAC-2000	$+1.99 \pm 1.31$	yes	yes
300-class union	Shared+LR32	FeCAM (full cov.)	$+3.40 \pm 1.97$	yes	yes
300-class union	Shared+LR32	RanPAC-2000	$+6.49 \pm 0.64$	yes	yes
300-class union	Shared+LR32	RanPAC-5000	$+3.05 \pm 0.34$	yes	yes
300-class union	Shared+LR32	RanPAC-10000	$+1.17 \pm 0.69$	yes	yes
300-class union	Shared+LR32, uncentred prior	FeCAM (full cov.)	$+7.73 \pm 2.47$	yes	yes
300-class union	Shared+LR32, uncentred prior	RanPAC-2000	$+10.82 \pm 1.14$	yes	yes
300-class union	Shared+LR32, uncentred prior	RanPAC-5000	$+7.37 \pm 0.90$	yes	yes
300-class union	Shared+LR32, uncentred prior	RanPAC-10000	$+5.50 \pm 1.11$	yes	yes

S42 Selection on a held-out split

The adaptive-K threshold τ and the associative aggregation temperature were chosen by reading test accuracy at CIFAR base-20; the residual rank was not, as the rank result below shows. Table S42 repeats the choice legitimately: one fifth of every class's training images is held out (fixed per seed), every candidate is run through the full expansion protocol on the remaining four fifths and scored on the held-out fifth at 100 classes, and the winner is compared with the candidate a test-split reader would have picked. Held-out accuracy is higher than test accuracy throughout because the backbone itself was trained on the held-out images of the base classes, so only the ordering matters. The held-out choice is within 0.25 points of the test choice on every seed for both scalar hyperparameters, and the rank family is monotone on both splits: the largest rank in the grid wins everywhere, so rank 32 is where we chose to stop paying storage (rank 64 costs 13.7 MB against 7.5), not a value tuned to the test set. The five-seed test means of the Shared+LR rows coincide with Table 1 by construction.

Table S42: Hyperparameter selection on a held-out fifth of the training data; mean±std over 5 seeds of accuracy at 100 classes on the held-out fifth (fitted on the other four fifths) and on the test set (fitted on all training images). "Chosen" counts the seeds on which each candidate won its family on the held-out split / on the test split.

Candidate	held-out	test	chosen (held-out / test)
Residual rank			
Shared only	0.4231 ± 0.0040	0.3700 ± 0.0057	0 / 0
Shared+LR4	0.4738 ± 0.0037	0.4194 ± 0.0068	0 / 0
Shared+LR8	0.4906 ± 0.0028	0.4382 ± 0.0064	0 / 0
Shared+LR16	0.5102 ± 0.0030	0.4582 ± 0.0071	0 / 0
Shared+LR32 (reported)	0.5263 ± 0.0034	0.4760 ± 0.0058	0 / 0
Shared+LR64	$\mathbf{0.5397 \pm 0.0036}$	$\mathbf{0.4890 \pm 0.0052}$	5 / 5
Adaptive-K dispersion threshold τ			
τ=0.50	0.3609 ± 0.0071	0.3126 ± 0.0086	0 / 0
τ=0.70	0.3580 ± 0.0090	0.3074 ± 0.0175	0 / 1
τ=0.85 (reported)	0.3688 ± 0.0082	0.3201 ± 0.0110	2 / 2
τ=0.95	$\mathbf{0.3693 \pm 0.0075}$	$\mathbf{0.3209 \pm 0.0131}$	3 / 2
Associative aggregation temperature			
T=0.1	0.3710 ± 0.0073	0.3227 ± 0.0121	0 / 0
T=0.25	$\mathbf{0.3751 \pm 0.0076}$	0.3245 ± 0.0117	2 / 2
T=0.5 (reported)	0.3744 ± 0.0073	$\mathbf{0.3252 \pm 0.0094}$	3 / 3
T=1.0	0.3636 ± 0.0062	0.3197 ± 0.0109	0 / 0
T=2.0	0.3472 ± 0.0074	0.3054 ± 0.0093	0 / 0

S43 Which conclusions transfer, and which do not

The experiments span three architectures we train, three we do not, two datasets and several sample-size and ordering regimes, and the conclusions do not all survive that spread equally. Scattering the scope conditions through the sections that report them makes them easy to miss, so we collect them here. A claim is robust if it holds on every setting in which we could test it, conditional if it holds only under a stated condition, and not established if our own criterion does not support it.

Table S43: Scope of each conclusion. "Tested on" names the settings the claim was exercised in; the verdict is against the two-unit criterion of Appendix S23.

Conclusion	Tested on	Verdict
Shared whitening + low-rank residual is more storage-efficient than full per-class covariance	ResNet-18/50, VGG-16, Tiny-ImageNet, 3 donors on CIFAR and on Food-101	robust
No readout is both smaller and more accurate than ours	5 of 6 representations on the mean; on VGG-16 RanPAC-1000 edges rank 32 by $+0.74 \pm 1.12$, not significant (Table S41)	conditional
The uncentred prior strengthens the memory	every backbone trained on the base classes (+1.7 to +4.3); ImageNet ResNet +0.6; CLIP and DINOv2 negative	conditional on training on the base classes
Its between-class term helps through the trained classes' directions	positive on 17/17 pad-crop-backbone seeds (21/23 of all trained backbones), negative on 9/9 donor seeds; beats held-out classes' scatter at matched trace on 4/4 backbones; not reproduced by a base-class score penalty (3/3 backbones); turns positive when a donor's last block is fine-tuned on the base classes, not on other classes; vanishes on a backbone trained with stronger augmentation; a specialisation index predicts its sign in 14/15 settings	conditional on base-class specialisation (necessary, not sufficient)
That term fades as B grows	$B \in \{5, 10, 20, 50\}$: +1.06 to +1.81, Spearman -0.40	not estab.
Pooled EC_τ overstates what arriving classes get	CIFAR-100, all memories	robust
Second-order estimation outweighs prototype multiplicity	3 trained architectures, 3 donors, Tiny-ImageNet, Food-101	robust; size varies 1.9–14×
Per-class covariance inverts under sample starvation	CIFAR-100, $n \in \{10, \ldots, 500\}$; not at FeCAM's few-shot shrinkage (Table S8), which still trails by 2.4–5.8	conditional on the shrinkage
Arrival-time geometry predicts hostability; base-class geometry does not	CIFAR-100 only	conditional — untested elsewhere
Capacity is larger at intermediate depth	ResNet-18, VGG-16 yes; ResNet-50 no (-1.1 ± 3.5, 3 seeds); Tiny-ImageNet and all 3 upsampled-CIFAR donors reversed; Food-101 at native resolution: ImageNet ResNet layer3 +0.33, ViTs reversed; ResNet-18 trained on Food-101 at native resolution: +13.5 (+6.5 with random-resized-crop)	conditional on architecture and narrow base training
Prototype multiplicity is not the lever	from-scratch yes; donors show +1.2 to +4.6 pts	conditional on from-scratch training
Our memory beats full covariance on accuracy	CIFAR-100 marginal; Tiny-ImageNet and ImageNet ResNet yes; VGG-16 reversed; ResNet-50, CLIP, DINOv2 tied	not estab.; storage is the claim
Base-set choice determines future capacity	B=5, 10 partly; at B=20 inside arrival-order noise	conditional on small B
EC_τ separates memories	only for $\tau \in [0.50, 0.65]$ here, and it orders them almost exactly as average incremental accuracy does ($\rho = 0.992$, Table S30)	conditional; it is a unit and a decomposition, not a re-ranking

The robust rows are what we would defend; the conditional rows are reported because they were surprising, and because knowing where a finding stops is part of what the measurement is for. Two of them — depth and prototype multiplicity — hold only for backbones trained from scratch on few classes, which the donor experiments establish.

S44 Minimal adaptation ladder

How much adaptation does it take to close the gap once frozen expansion is exhausted? Each level unlocks a strictly larger trainable set; all levels see the full 100-class training data, so this is an upper bound on what adaptation buys, not an incremental protocol. Mean±std over 3 seeds. The levels are not matched on optimisation budget, and we state the budgets: L1 trains 40 epochs of one-cycle SGD on pixels, L3–L5 train 30, and L2's linear projection takes 200 single-minibatch AdamW steps. L1 is therefore also not the same object as Table 1's joint linear probe, which gets 60 epochs on cached features and reaches 0.4294 ± 0.0091 against L1's 0.4152 ± 0.0100; the honest margin for our memory over a trained linear head is the 0.4294 one.

Level	What is trained	Acc@100	Trainable params
L0	nothing (frozen + our memory)	0.4749 ± 0.0073	0
L1	linear head	0.4152 ± 0.0100	51,300
L2	linear projection + memory	0.4585 ± 0.0051	262,144
L3	BatchNorm scale/shift + head	0.5146 ± 0.0062	60,900
L4	final residual block + head	0.6608 ± 0.0053	8,445,028
L5	everything	0.7552 ± 0.0025	11,230,392
—	joint training on all 100 classes	0.7782	11,230,392

Two observations. Our memory, which adds zero trainable parameters (though it does store class statistics, counted in the memory column of Table 1), reaches 0.4749 — beating both a trained linear head and a trained projection, so modelling the second-order structure of frozen features is worth more than learning a readout on top of them. And recalibrating normalisation statistics (L3, 60,900 parameters, 0.54% of the network) beats the 262,144-parameter projection by 5.6 points: the cheapest useful adaptation adjusts feature scaling rather than learning new features. Closing the remaining gap to 0.7552 costs 185× more parameters than L3.

S45 Expansion at each backbone tap

Table S44: Expansion at each backbone tap, mean±std over 3 seeds, using shared whitening with a rank-16 residual. Retention is Acc@100 / closed-set.

Tap	d	Closed-set (20 cls)	Acc@100	Retention
stem	64	0.3903 ± 0.0278	0.1767 ± 0.0052	0.454 ± 0.020
layer1	64	0.5037 ± 0.0202	0.2621 ± 0.0065	0.521 ± 0.024
layer2	128	0.6567 ± 0.0124	0.3920 ± 0.0060	0.597 ± 0.019
layer3	256	0.8187 ± 0.0281	$\mathbf{0.5204 \pm 0.0038}$	$\mathbf{0.636 \pm 0.024}$
layer4	512	$\mathbf{0.8730 \pm 0.0245}$	0.4575 ± 0.0092	0.524 ± 0.005

The advantage does not survive a change of dataset and input resolution. On Tiny-ImageNet at 64×64, with the identical architecture and protocol expanded to 200 classes, the final block is best for every memory we ran, though the rank-32 gap (-0.35, two seeds) is within noise:

Memory	layer2	layer3	layer4	L3−L4 (pts)
Class mean (3 seeds)	0.1197 ± 0.0015	0.1652 ± 0.0029	$\mathbf{0.1800 \pm 0.0024}$	−1.47
Shared+LR16 (3 seeds)	0.2395 ± 0.0052	0.3171 ± 0.0107	$\mathbf{0.3242 \pm 0.0019}$	−0.72
Shared+LR32 (2 seeds)	0.2502 ± 0.0059	0.3340 ± 0.0127	$\mathbf{0.3374 \pm 0.0040}$	−0.35

Against $+6.70 \pm 0.34$ points for layer3 on CIFAR (Appendix S22), this is a reversal, and it moves dataset, label space and input resolution together, exactly as the donor comparison of §5.6 does. We therefore state the depth result as a property of CIFAR-scale backbones trained from scratch on few classes at 32×32, and not as a property of depth. Harun et al. (2024) implicate resolution and data diversity alongside class count in the same effect, and our evidence cannot separate the three.

S46 Matched-budget layer frontier

Table S45: Best accuracy at 100 classes reachable under a memory cap, per tap, searching the rank ladder of the shared-whitening memory and the class mean only (not the prototype or diagonal variants); mean over 3 seeds. Each cell names the parameterisation that achieves it.

Memory cap	layer2 (d=128)	layer3 (d=256)	layer4 (d=512)
$\leq$ 0.5 MB	0.3454 (LR4)	**0.4225** (shared)	0.3104 (class mean)
$\leq$ 1.0 MB	0.3920 (LR16)	**0.4818** (LR4)	0.3104 (class mean)
$\leq$ 2.0 MB	0.4108 (LR32)	**0.5204** (LR16)	0.4194 (LR4)
$\leq$ 4.0 MB	0.4267 (LR64)	**0.5338** (LR32)	0.4365 (LR8)
$\leq$ 8.0 MB	0.4267 (LR64)	**0.5481** (LR64)	0.4749 (LR32)

S47 Accuracy against base-class count

Table S46 varies the size of the base class set, holding the final label space fixed at 100 classes. Capacity rises steeply with the number of base classes: from $B = 5$ to $B = 50$ the accuracy at 100 classes nearly doubles, and the fraction of the joint-training ceiling recovered (chance-corrected, Table S6) rises from 0.40 to 0.77. Our memory is the strongest or tied-strongest readout at every base size; at B=50 the margin over full covariance is +0.33 points (0.6025 against 0.5992, 3 seeds, no interval reported), too small to call a win.

Closed-set accuracy moves the other way, from 0.905 to 0.811. We are careful not to over-read this: the closed-set task is a different problem at each row — 5-way versus 50-way — so the decline says nothing about representation quality. That is exactly the point. Closed-set accuracy cannot be used to rank frozen representations by how much room they have left, because it is not measured on a common task; expansion capacity has to be measured directly, which is what EC_τ is for.

Table S46: Accuracy at the full 100-class label space as a function of the number of base classes the frozen backbone was trained on; mean±std over 3 seeds; FeTrIL here uses its cross-entropy head. Joint training on all 100 classes reaches 0.7782.

Base	Closed-set	NCM	FeTrIL	Linear	FeCAM	Ours
5	0.9047 ± 0.0333	0.1166 ± 0.0049	0.2083 ± 0.0015	0.2609 ± 0.0051	0.2812 ± 0.0143	$\mathbf{0.3184 \pm 0.0168}$
10	0.8803 ± 0.0502	0.1958 ± 0.0220	0.2470 ± 0.0132	0.3488 ± 0.0088	0.3848 ± 0.0062	$\mathbf{0.4094 \pm 0.0061}$
20	0.8732 ± 0.0254	0.3104 ± 0.0085	0.3416 ± 0.0135	0.4292 ± 0.0098	0.4650 ± 0.0058	$\mathbf{0.4749 \pm 0.0073}$
50	0.8114 ± 0.0062	0.5266 ± 0.0043	0.5192 ± 0.0062	0.5689 ± 0.0007	0.5992 ± 0.0014	$\mathbf{0.6025 \pm 0.0036}$

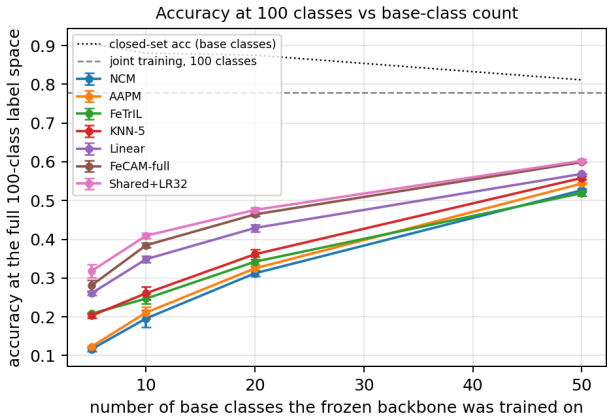


Figure S5: Accuracy at 100 classes against base-class count, mean±std over 3 seeds. It rises steeply with the number of base classes while closed-set accuracy (dotted) falls — but on a different task at each point, which is why capacity must be measured directly rather than inferred from closed-set skill.