ALLOCATING WEIGHT PRECISION FOR DEEP IMBALANCED REGRESSION WITH EDGE-CASE ANNEALING

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep Imbalanced Regression (DIR) addresses the mismatch between naturally imbalanced training distributions and the more balanced performance desired across the target range. Methods that resolve the underlying capacity conflict, in which parameters shared across the target range receive updates dominated by majority samples, currently do so by restructuring the model and therefore sacrifice orthogonality. We introduce Edge-Case Annealing (ECA), a training-schedule method that leaves the loss and the architecture untouched. ECA couples a data exposure curriculum one to one to the learning rate schedule: a ceiling on the target histogram tracks the learning rate, so that the effective training distribution reaches balance exactly as the step size vanishes. The design follows a minimum description length view of capacity: under the imbalanced training distribution, fine-grained structure for rare targets does not justify the weight precision it requires, whereas under the balanced evaluation distribution it does, and it can be retained once the learning rate, and with it the gradient noise, has decayed. On AgeDB-DIR, IMDB-WIKI-DIR, and the MuViS-DIR time series benchmark, ECA is the strongest orthogonal addition, improves the tail without the majority tax of static rebalancing, and composes with existing methods up to a new state of the art in the medium and few-shot regions. Under increasing weight noise, the annealing gains are erased in rare target regions, consistent with being stored as small, precision-fragile adjustments and supporting the proposed capacity view.

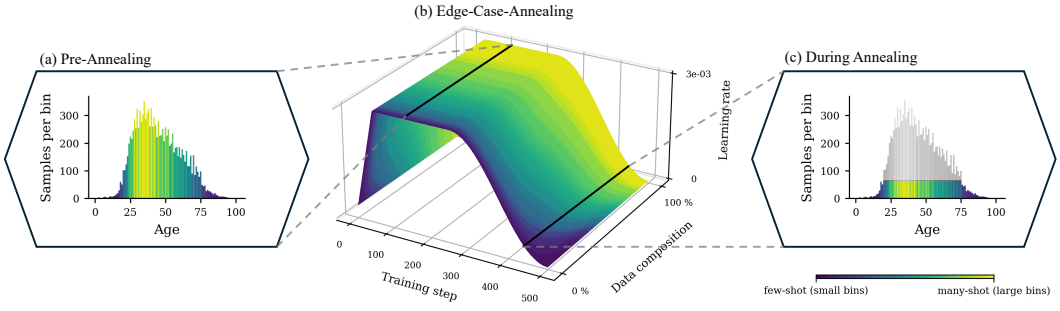


Figure 1: **Edge-Case Annealing** on AgeDB-DIR. (a) The natural training distribution, colored by bin size from few-shot (dark) to many-shot (yellow). (b) ECA over the course of training: the surface traces the learning rate schedule, and its color shows the composition of the effective training data at each step. During the stable phase, the composition is dominated by the many-shot bins of the natural distribution; whenever the learning rate is small (as shown in (c)), few-shot bins occupy a growing share, because the descending ceiling removes mass from the densest bins first. (a, c) Snapshots of the training distribution at the positions marked in (b).

1 INTRODUCTION

Regression models are routinely deployed to predict exactly the values that their training data contain the least. Extreme wind speeds (Maskey et al., 2020), unusually high biomass (Lang et al., 2023), rare physiological (Reiss et al., 2019) states: the conditions that make a prediction in these cases valuable are, almost by definition, the ones that occur rarely, while unremarkable observations accumulate in abundance. Trained on such data, deep models inherit the imbalance of their supervision. Gradient updates are dominated by densely covered target regions, and performance degrades

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.

toward the tails, precisely where errors are most consequential (Yang et al., 2021). The discrepancy between the empirical training distribution and the desired evaluation distribution amounts to a target-distribution shift (Puetz et al., 2026a), and the research field of Deep Imbalanced Regression (DIR) has emerged explicitly to address it (Yang et al., 2021). The setting is particularly challenging for continuous targets, where conventional classification-based rebalancing techniques are ill-suited.

Recent years have seen a growing body of DIR methods that improve performance in under-represented regions of the target space. We argue that a practically useful DIR method should satisfy two desiderata. It should be *data agnostic*, in that it does not rely on modality-specific techniques such as image augmentation, and it should be *model orthogonal*, in that it applies to arbitrary architectures, ideally in a black-box fashion that requires no modification of the network architecture or its latent representations.

Existing evidence points to shared capacity as a root cause of the DIR failure mode (Jiang et al., 2024). Puetz et al. (2026a) describe a cascading effect in which parameters shared across the entire target range receive updates dominated by majority samples; the accumulated bias culminates in a feature collapse, where rare targets are absorbed into the representation of the dominant region (Yang et al., 2021). Jiang et al. (2024) show that freeing up capacity through dedicated prediction heads counteracts this effect and reaches state-of-the-art (SOTA) performance. Yet allocating capacity architecturally requires changing the model, which sacrifices orthogonality. Inspired by the minimum description length (MDL) view of Hinton & Van Camp (1993), we hypothesize that the necessary capacity already exists inside the weights as residual precision. Fine-grained structure for rare targets is not worth its precision under p_{train}, but it is under p_{eval}, and it can be retained late in training, once the gradient noise has decayed with the learning rate.

Building on this view, we introduce *Edge-Case Annealing* (ECA), a training-schedule method for DIR (Figure 1). ECA couples a data exposure curriculum to the learning rate schedule: every label bin is clipped to a ceiling that tracks the learning rate, so that the effective training distribution morphs continuously between a balanced and the natural, imbalanced distribution and reaches full balance exactly while the step size goes to zero. The model thus consolidates the bulk of the data at full step size and receives its rebalanced, tail-emphasized exposure when updates are small enough to act as granular corrections. This view also yields a falsifiable prediction: if the tail gains are stored as small, precision-fragile deviations in the weights, perturbing the weights should erase the tail gains of annealing while leaving the majority mapping largely unaffected. Our contributions are:

1. We propose ECA, a continuous curriculum for DIR that preserves the continuous nature of the problem along both the target axis and the time axis. ECA is a black-box method that is data agnostic and model orthogonal. It reaches SOTA performance on established DIR benchmarks and composes with existing DIR methods.
2. We provide extensive ablations that support the weight-precision hypothesis, including weight-noise and interpolation probes, and show that ECA is agnostic to model size.

2 Related Work

The problem of imbalanced training data is not new in deep learning, but most imbalance-aware methods were developed for classification, where the output space is categorical and finite (He & Garcia, 2009). Extending these ideas to continuous and potentially infinite target spaces is non-trivial. Yang et al. (2021) introduced DIR as a distinct problem setting and showed that reweighting and resampling strategies transferred from classification perform poorly once the notion of a class boundary disappears. Since then, DIR has developed into a substantial methodological literature, broadly grouped into three families (Puetz et al., 2026a): *algorithm-level* methods that modify the loss or sample weighting, *representation-learning* methods that reshape the latent space and *architectural* methods that modify the predictor itself. Cutting across these families, methods divide into *orthogonal* approaches that compose with standard regression pipelines and *non-orthogonal* approaches that require specific architectures or substantially restructured training (a full overview can be found in Appendix C). The methods that address the root cause of the failure mode, namely shared capacity receiving majority-dominated updates, tend to do so by restructuring the model and are therefore non-orthogonal (Jiang et al., 2024). Orthogonal methods, in contrast, typically leave the capacity conflict in place and compensate for it through the loss surface or the learned repre-

sentation. ECA occupies the missing quadrant: it targets the capacity conflict directly, but from the training-schedule angle, and therefore remains fully orthogonal.

The insight that rebalancing belongs late in training has precedent in long-tailed classification. Cao et al. (2019) defer re-weighting or re-sampling until after a learning rate drop (DRW/DRS), and Kang et al. (2020) interpolate from instance-balanced to class-balanced sampling over the course of training. ECA can be read as the regression-native continuation of this idea: rather than a one-shot regime switch or an epoch-linear interpolation, exposure morphs as a smooth function of the remaining step size, over a continuous target space in which bins serve only to measure target density rather than to define semantic categories, and with the objective left untouched.

Reweighting methods for DIR, such as label distribution smoothing (LDS) and feature distribution smoothing (FDS) (Yang et al., 2021), Balanced MSE (Ren et al., 2022), and density-based weighting (Steininger et al., 2021), alter the loss surface for the entire run, so the model never fits the natural distribution and tail gains typically come at a cost in the majority region (section 4). ECA instead confines rebalancing to the phase in which updates are small enough to leave the consolidated majority mapping largely intact.

Curriculum learning orders training data by difficulty, typically easy-first, to accelerate and stabilize optimization (Bengio et al., 2009); hard-example mining emphasizes high-loss instances throughout training (Shrivastava et al., 2016). Modern pretraining pipelines reserve high-value or specially curated data for the final phase of training. Warmup-stable-decay schedules make this phase explicit (Hägele et al., 2024), and decay-phase data curation is used in MiniCPM (Hu et al., 2024), Llama 3, OLMo 2, and in domain upsampling at the end of training (Blakeney et al., 2024). ECA transfers this principle to supervised regression, with rare label regions as the notion of high-value data and a formal coupling of exposure to the learning rate schedule.

3 METHOD

3.1 PROBLEM FORMULATION

We consider a regression task in which a model $f_\theta \colon \mathcal{X} \to \mathcal{Y}$ with parameters θ learns a mapping from input $x \in \mathcal{X}$ to a continuous target $y \in \mathcal{Y} \subseteq \mathbb{R}$. The model is trained by minimizing a loss $\mathcal{L}\big(f_\theta(x), y\big)$ over a training set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{M}$ whose targets follow the empirical distribution $p_{\text{train}}(y)$. In the DIR setting, $p_{\text{train}}(y)$ is highly non-uniform, while the evaluation objective is to generalize with respect to a more balanced distribution $p_{\text{eval}}(y)$. This mismatch constitutes an implicit target-distribution shift that weakens the empirical risk minimization assumption of identically distributed training and evaluation samples. Following Yang et al. (2021), we partition the supported target range into B contiguous bins $\{\mathcal{B}_1, \ldots, \mathcal{B}_B\}$ of equal width and write m_b for the number of training samples in bin $\mathcal{B}_b$. Binning is only an auxiliary discretization for the data schedule below, neither the model nor the loss operates on discretized targets.

3.2 EDGE-CASE ANNEALING

Under imbalance, parameters shared across the target range receive updates dominated by majority samples, and the accumulated bias collapses the representation of rare targets (Puetz et al., 2026a). Allocating dedicated capacity mitigates this effect, but often requires modifying the architecture (Jiang et al., 2024). We hypothesize that the necessary capacity already exists inside the weights as residual precision. Our intuition comes from the MDL view, in which learning compresses the data through a two-part code,

$$L(\mathcal{D}) = L(\theta) + L(\mathcal{D} \mid \theta), \qquad L(\mathcal{D} \mid \theta) = \sum_{i=1}^{M} -\log_2 p_\theta(y_i \mid x_i), \tag{1}$$

where the model cost $L(\theta)$ grows with the precision to which the weights must be specified (Hinton & Van Camp, 1993) and the data cost $L(\mathcal{D} \mid \theta)$ is the encoding cost of the targets under the model's predictive distribution $p_\theta(y \mid x)$. Precision invested in θ pays off only where it reduces $L(\mathcal{D} \mid \theta)$ by more than it costs, and this reduction scales with the number of samples in a target region. Stochastic gradient methods impose a comparable precision limit implicitly: their gradient noise keeps the weights fluctuating on a scale that grows with the learning rate (Smith & Le, 2018), so structure finer than this scale is not retained stably. Under p_{train}, standard training therefore

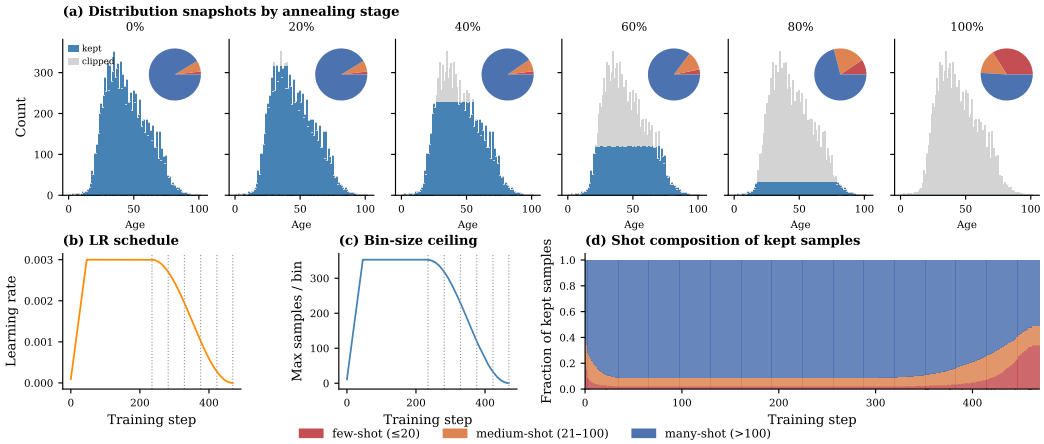


Figure 2: **ECA in depth** on the example of AgeDB-DIR. (a) Snapshots of the effective training distribution during the decay phase; the descending ceiling clips the densest bins first, and the pie chart in each corner gives the current share of the many, medium, and few-shot regions. (b, c) Coupling of the learning rate schedule $\eta(n)$ and the annealing coefficient $\xi(n)$; the dotted vertical lines mark the positions of the snapshots in (a). (d) Resulting data exposure over training, separated into the three shot regions.

consolidates the many-shot mapping, which dominates the gradient, in perturbation-robust structure and rarely writes structure specific to the tail. ECA combines the cost–benefit view of MDL with the learning-rate-dependent precision limit of stochastic gradient methods to allocate weight precision to the tail: under p_{eval}, where rare targets carry far more weight, their fine-grained structure does pay off, and it can be retained once the gradient noise is small. It therefore consolidates the natural distribution at full step size and rebalances exposure toward rare targets as the step size decays, so tail corrections are written as small deviations in residual precision, largely without overwriting the majority mapping. This yields a testable prediction: corrections stored in residual precision tolerate less perturbation than the majority mapping, so injecting noise of increasing magnitude into the weights should erase the tail gains of annealing while leaving the majority mapping largely unaffected, which we verify in subsection 4.4.

Learning rate schedule. ECA couples a data exposure curriculum (DEC) to a learning rate schedule with an explicit decay phase. We use warmup-stable-decay (WSD) (Hägele et al., 2024), which holds the learning rate constant for most of training and drives it to zero at the end (Figure 2b):

$$\eta(n) = \begin{cases} \dfrac{n}{N_{\text{warmup}}}\,\eta_{\max} & \text{if } n \le N_{\text{warmup}}, \\ \eta_{\max} & \text{if } N_{\text{warmup}} < n \le N - N_{\text{decay}}, \\ f(n, N, N_{\text{decay}})\,\eta_{\max} & \text{if } n > N - N_{\text{decay}}, \end{cases} \tag{2}$$

with peak learning rate $\eta_{\max}$, total steps N, phase lengths N_{warmup} and N_{decay}, and a monotonically decreasing decay shape f that decreases from 1 at $n = N - N_{\text{decay}}$ to 0 at $n = N$.

Data exposure curriculum. The DEC of ECA is governed by an annealing coefficient that mirrors the learning rate schedule one to one,

$$\xi(n) = \frac{\eta(n)}{\eta_{\max}} \in [0, 1], \tag{3}$$

so the data schedule inherits the warmup, stable, and decay phases of Equation 2. The coefficient lowers a ceiling on the target histogram. With $m_{\max} = \max_b\{m_b\}$ and a minimum bin size $c_{\min} \ge 1$ (set to 1 unless stated otherwise), every bin is clipped to

$$c_b(n) = \min\big(m_b,\ C(n)\big), \qquad C(n) = c_{\min} + \xi(n)\,\big(m_{\max} - c_{\min}\big), \tag{4}$$

yielding the effective training distribution $p_n(\mathcal{B}_b) \propto c_b(n)$. During the stable phase, $\xi(n) = 1$ and the model trains on the unmodified natural distribution; during warmup and decay, the ceiling

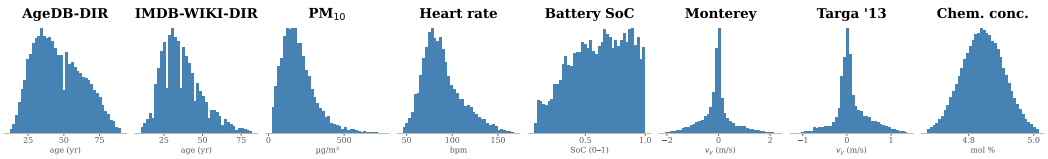


Figure 3: Target distributions of eight of the eleven datasets used in the experiments.

is lowered in step with the learning rate, so exposure is balanced whenever updates are small, at the very start and at the very end of training. As the ceiling descends, surplus mass is removed from the densest bins first, sparser bins are affected only once the ceiling reaches their count, and at the end of training every supported bin contributes equally (Figure 2a). The clipped subset is re-drawn uniformly at random at every epoch, so surplus samples rotate through the caps rather than being discarded. Exposure thus morphs continuously from the natural to a balanced distribution, synchronized with the step size, without imposing class structure on the targets and without any discrete regime switch.

Properties and choice of schedule. ECA changes which samples the model sees, nothing else: the loss remains the chosen regression objective, no gradients are rescaled, no samples are duplicated or synthesized, and neither the architecture nor the latent space is modified. Its entire interface to the training pipeline is the sampler, which makes ECA data agnostic, model orthogonal, and freely composable with existing DIR methods. ECA can also be coupled to any learning rate schedule with a decay phase. We use WSD because its constant stable phase permits open-ended training, with the decay initiated at any time, and because the sharp stable-to-decay boundary isolates the annealing mechanism and exposes the decay shape f as an explicit design choice (Hägele et al., 2024).

4 Experiments

4.1 Experimental Setup

We evaluate on AgeDB-DIR and IMDB-WIKI-DIR, the two most widely used DIR benchmarks, to cover the common computer vision use case, and on MuViS-DIR to cover health, environmental, and engineering domains in the time series modality (Figure 3). We anchor our comparison in the result set reported for UVote (Jiang et al., 2024), the strongest published method to date; a discussion of the fragmented evaluation landscape in DIR and of the systematic weaknesses identified by Puetz et al. (2026b) is provided in subsection D.1. For AgeDB-DIR and IMDB-WIKI-DIR we retain the established all, many, medium, and few-shot reporting for comparability with prior work. For MuViS-DIR we adopt the bMASE protocol of Puetz et al. (2026b) on the basis of Ren et al. (2022), since shot-based reporting would impose an arbitrary binning on each dataset (shot-based results are nevertheless given in Appendix I).

Although ECA leaves many degrees of freedom in principle, we keep its configuration deliberately minimal. The target discretization is inherited from the respective benchmark (subsection D.5), while all schedule parameters are inherited from WSD, the annealing coefficient mirrors the learning rate throughout (Equation 3) and the ceiling always decays to a minimum bin size of $c_{\min} = 1$. ECA therefore introduces no additional hyperparameters that are tuned independently of the underlying benchmark and learning-rate configuration. For WSD we use a cosine decay shape in all main experiments and vary only the phase fractions per benchmark: a warmup fraction of 0.1 for IMDB-WIKI-DIR and 0.2 for AgeDB-DIR and MuViS-DIR, and a decay fraction of 0.2 for AgeDB-DIR, 0.4 for IMDB-WIKI-DIR, and 0.1 for MuViS-DIR. The peak learning rate matches the established configuration of the respective vanilla model. All predictive-performance experiments are run across ten seeds and we report the mean (standard deviations and the full configurations are listed in Appendix G to Appendix I.)

4.2 Results

AgeDB-DIR Table 1 compares ECA against the established DIR method landscape. Three observations stand out. First, V. + ECA is the strongest single orthogonal addition across all methods, reducing the overall MAE by 10.03% and providing the largest improvements in the medium (12.12%) and few-shot (23.45%) regions among all orthogonal methods. Notably, and in contrast

Table 1: Results on AgeDB-DIR (MAE, mean over ten seeds; standard deviations in Appendix G). Improvements are relative to the reference row given in the last column. **Bold** is best and underline the second best value per column and group.; V. abbreviates Vanilla. Rows with ECA are ours.

#	Method	MAE				Improv. (%)				vs. #
		All	Many	Med	Few	All	Many	Med	Few	
Orthogonal methods										
1	Vanilla	7.68	6.67	8.99	13.52	–	–	–	–	–
2	V. + LDS	7.69	7.03	8.64	11.32	-0.13	-5.40	3.89	16.27	1
3	V. + FDS	7.69	6.67	9.09	13.50	-0.13	0.00	-1.11	0.15	1
4	V. + LDS + FDS	7.75	7.07	8.71	11.49	-0.78	-0.57	-0.81	-1.50	2
5	V. + Focal L_1	7.45	6.61	8.69	12.04	2.99	0.90	3.34	10.95	1
6	V. + SQINV	7.69	7.02	8.65	11.42	-0.13	-5.25	3.78	15.53	1
7	V. + RRT	7.36	6.58	8.30	12.15	4.17	1.35	7.68	10.13	1
8	V. + RRT + BNI	7.37	6.57	8.29	12.33	-0.14	0.15	0.12	-1.48	7
9	V. + RRT + BMC	7.30	6.53	8.19	12.17	0.82	0.76	1.33	-0.16	7
10	V. + RankSim	7.28	6.39	8.49	12.32	5.21	4.20	5.56	8.88	1
11	V. + LDS + RankSim	7.21	6.61	8.07	10.40	6.24	5.97	6.60	8.13	2
12	V. + FDS + RankSim	7.33	6.43	8.52	12.55	4.68	3.60	6.27	7.04	3
13	V. + LDS + FDS + RankSim	7.19	6.62	8.00	10.33	7.23	6.36	8.15	10.10	4
14	V. + Focal L_1 + RankSim	7.29	6.41	8.56	12.08	2.15	3.03	1.50	-0.33	5
15	V. + SQINV + RankSim	7.16	6.57	8.02	**10.27**	6.89	**6.41**	7.28	10.07	6
16	V. + RRT + RankSim	7.25	6.51	8.13	11.86	1.49	1.06	2.05	2.39	7
17	V. + ConR	7.31	6.38	8.58	12.51	4.82	4.35	4.56	7.47	1
18	V. + LDS + ConR	7.30	6.71	8.10	10.69	5.07	4.55	6.25	5.57	2
19	V. + FDS + ConR	7.32	6.41	8.59	12.41	4.81	3.90	5.50	8.07	3
20	V. + LDS + FDS + ConR	7.32	6.73	8.10	10.72	5.55	4.81	7.00	6.70	4
21	V. + ECA (ours)	**6.91**	**6.25**	**7.90**	10.35	**10.03**	6.30	**12.12**	**23.45**	1
Non-orthogonal methods										
22	RnC	**6.50**	**5.83**	7.61	9.73	**15.36**	**12.59**	15.35	28.03	1
23	UVote	6.77	6.38	7.32	8.91	11.85	4.35	**18.58**	**34.10**	1
24	VIR	6.83	6.29	7.54	9.95	11.07	5.70	16.13	26.41	1
-	V. + ECA (ours)	6.91	6.25	7.90	10.35	10.03	6.30	12.12	23.45	1
25	UVote + ECA (ours)	6.96	6.88	**6.88**	**7.98**	-2.81	-7.84	6.01	10.44	23

to static rebalancing methods such as LDS and SqInv, which pay for their tail gains with degraded many-shot performance (section 2), ECA improves the many-shot region as well (6.30%). Only the non-orthogonal methods RnC, UVote, and VIR achieve better overall scores. Second, ECA composes with existing methods: added on top of ConR, it still contributes 8.39% and 9.59% improvement in the medium and few-shot regions relative to ConR alone (shown in Table 14). Third, added on top of UVote, ECA sets a new SOTA for the medium and few-shot regions on AgeDB-DIR, reaching an MAE of 6.88 and 7.98 and improving the few-shot performance of UVote by a further 10.44%. This gain is not free: the many-shot error increases relative to UVote, and the overall MAE degrades slightly.

IMDB-WIKI-DIR On IMDB-WIKI-DIR (Table 2), V. + ECA remains the strongest single orthogonal addition, with the largest overall orthogonal improvement (8.24%) and the largest many-shot improvement (6.89%), alongside a 15.21% gain in the medium region. Its overall MAE of 7.40 is second only to the non-orthogonal UVote (7.36). The picture for composition differs from AgeDB-DIR: here, ECA shifts the gains of its partner methods toward the medium region. UVote + ECA attains the best medium-shot MAE in the table (10.62, a further 9.85% over UVote) at a nearly unchanged few-shot error (-0.96%), but pays in the many-shot and overall scores. V. + SQINV + ECA reaches the second best med.-shot value (11.51) while improving over SQINV in every region.

MuViS-DIR On MuViS-DIR (Table 3), ECA is the best orthogonal method in the geometric mean and second overall behind UVote. It achieves the best result of all methods on the PPG-DaLiA heart rate estimation task, and exhibits the smallest standard deviation of all methods across the ten seeds, making it the most stable method in the comparison. ECA is also the best orthogonal method on PM2.5, Monterey and Tyre Temp., and the second best orthogonal method on Targa '14 and Chem. Conc. Its clearest failure case is Targa '13, where it falls behind the Vanilla baseline, highlighting

Table 2: IMDB-WIKI-DIR: Baseline numbers from Jiang et al. (2024); improvements relative to the reference row (last column). **Bold** is best and underline the second best value per column and group.

#	Method	MAE				Improv. (%)				vs. #
		All	Many	Med	Few	All	Many	Med	Few	
Orthogonal methods										
1	Vanilla	8.06	7.23	15.12	26.33	–	–	–	–	–
2	V. + RankSim	7.72	6.93	14.48	25.38	4.22	4.15	4.23	3.61	1
3	V. + RankSim + RRT	7.55	6.83	13.47	24.72	2.20	1.44	6.98	2.60	2
4	V. + RankSim + SQINV	7.42	6.84	12.12	22.13	3.89	1.30	16.30	12.81	2
5	V. + RankSim + RRT + LDS + FDS	**7.37**	6.80	11.80	23.11	2.38	0.44	12.40	6.51	3
6	V. + RankSim + SQINV + LDS + FDS	7.69	7.13	12.30	**21.43**	-3.64	-4.24	-1.49	3.16	4
7	V. + SQINV	7.87	7.24	12.44	22.76	2.36	-0.14	**17.72**	**13.56**	1
8	V. + SQINV + LDS + FDS	7.78	7.20	12.61	22.19	1.14	0.55	-1.37	2.50	7
9	V. + BMSE	8.08	7.52	12.47	23.29	-0.25	-4.01	17.53	11.55	1
10	V. + ECA (ours)	7.40	**6.73**	12.82	24.00	**8.24**	**6.89**	15.21	8.85	1
11	V. + SQINV + ECA (ours)	7.56	7.05	**11.51**	21.74	3.92	2.56	7.47	4.46	7
Non-orthogonal methods										
12	UVote	**7.36**	6.81	11.78	**20.96**	**8.68**	5.81	**22.09**	**20.40**	1
-	V. + ECA (ours)	7.40	**6.73**	12.82	24.00	8.24	**6.89**	15.21	8.85	1
13	UVote + ECA (ours)	7.67	7.26	**10.62**	21.16	-4.24	-6.63	9.85	-0.96	12

Table 3: bMASE on MuViS-DIR (mean over ten seeds). **Bold** marks the best and underline the second best value per dataset. Orthogonal add-ons (left) vs. non-orthogonal methods (right)

Dataset		Vanilla	V. + ConR	V. + Focal-L1	V. + LDS	V. + SQInv	V. + ECA		RnC	UVOTE	V. + ECA
PM10	*Orthogonal methods*	0.6677	0.6793	0.6171	**0.4765**	0.6230	0.6394	*Non-orthogonal methods*	0.6663	0.6792	**0.6394**
PM2.5		0.5705	0.5507	0.5242	0.5489	0.5457	**0.5185**		0.5675	**0.5142**	0.5185
Heart Rate		0.1277	0.1925	0.0748	0.1130	0.1232	**0.0593**		0.0923	0.0782	**0.0593**
Battery SoC		0.0229	0.0306	0.0280	**0.0224**	0.0262	0.0245		0.2536	0.0251	**0.0245**
Monterey		0.2511	0.2695	0.2215	0.2401	0.2260	**0.1009**		0.3622	**0.0853**	0.1009
Targa '13		**0.1116**	0.1136	0.1933	0.1212	0.1132	0.2213		0.3452	**0.1136**	0.2213
Targa '14		0.1276	0.1319	0.1326	0.1273	**0.1183**	0.1218		0.1746	**0.1041**	0.1218
Chem. Conc.		0.6910	0.6485	0.6343	**0.5826**	0.6371	0.6295		0.6349	0.6056	0.6295
Tyre Temp.		0.3662	0.3618	0.3777	0.3534	0.3505	**0.3398**		0.3933	**0.3313**	0.3398
Geometric Mean		0.2157	0.2338	0.2135	0.1997	0.2084	**0.1870**		0.3314	**0.1729**	0.1870

a limitation of the method. Across datasets, a Friedman–Nemenyi analysis ranks ECA as the best orthogonal method on bMASE and first of all methods in the many-shot region, although most pairwise differences remain below the critical difference (Appendix M).

4.3 Ablations

Backbone size. Under the shared capacity view, backbone size should modulate the capacity conflict. We therefore repeat the V. + ECA experiment across ResNet-18, -34, -50, and -101 backbones (Table 4). Two observations follow. First, the Vanilla performance varies considerably with the backbone and is best for the smallest model. Second, ECA is effective at every size: the few-shot MAE improves for all four backbones and the overall MAE with ECA is nearly constant across backbones (6.91 to 6.99) although the Vanilla baselines vary between 7.17 and 7.68. ECA thus not only improves every configuration but also largely decouples the final performance from the backbone size.

Isolated components. To show that the coupling of the two is what makes the difference, we ablate both components separately (Table 5). Each component already helps in isolation, but in characteristically different regions. DEC alone, which amounts to progressive rebalancing of rare regions at full step size, improves the medium (10.12%) and few-shot (17.83%) regions but de-

Table 4: Backbone size ablation on AgeDB-DIR (MAE). **Bold** marks the best and underline the second best value per column.

Model	ECA	All	Many	Med	Few
ResNet18	✗	7.17	6.48	8.01	11.39
ResNet18	✓	6.91	6.29	**7.77**	10.37
ResNet34	✗	7.24	6.56	8.06	11.37
ResNet34	✓	6.97	**6.24**	8.12	10.76
ResNet50	✗	7.68	6.67	8.99	13.52
ResNet50	✓	**6.91**	6.25	7.90	10.35
ResNet101	✗	7.53	6.77	8.26	12.58
ResNet101	✓	6.99	6.39	7.90	**10.11**

Table 5: Component ablation on AgeDB-DIR (MAE). Improvements relative to Vanilla are listed in Table 27.

Method	All	Many	Med	Few
Vanilla	7.68	6.67	8.99	13.52
V. + DEC	7.44	6.86	8.08	11.11
V. + WSD	7.04	**6.20**	8.22	11.67
V. + ECA	**6.91**	6.25	**7.90**	**10.35**

grades the many-shot region (-2.85%), reproducing exactly the majority tax of static rebalancing discussed in section 2. WSD alone improves the many-shot region most (7.05%). The combination improves all regions simultaneously and dominates both components in the overall, medium and few-shot column, supporting the central design decision of ECA: rebalanced exposure is beneficial precisely when it coincides with a vanishing step size. Appendix J extends this analysis in three directions. First, WSD alone does not explain the gain: combined with WSD, LDS and SQINV end up worse overall than WSD alone (7.27 and 7.26 versus 7.04), whereas ECA is the only addition that improves on it. Second, coupling DEC to the discrete step schedule, a regression analogue of deferred re-balancing (Cao et al., 2019), falls behind ECA in every region. Third, desynchronizing the decay phases of DEC and WSD makes each variant revert toward one of the isolated components, and the synchronized configuration remains best in three of four regions.

4.4 Probing the Weight Precision Hypothesis

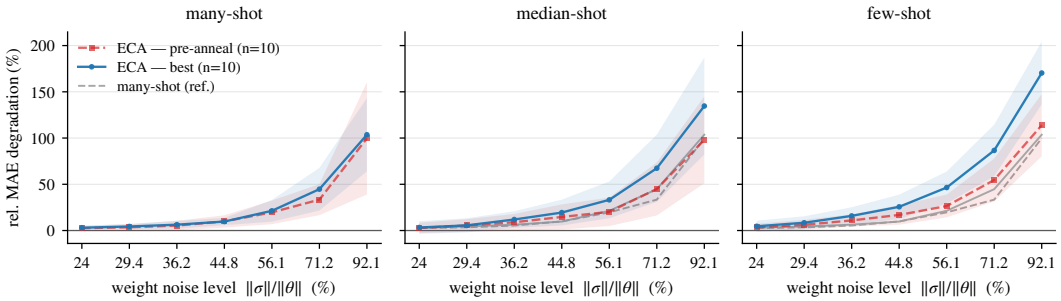


Figure 4: Relative test MAE degradation under Gaussian weight noise on AgeDB-DIR for the three shot regions many (left), medium (center), and few-shot (right). Lines show the mean over ten seeds with five noise draws per seed; bands show one σ. The pre-annealing checkpoint degrades almost uniformly across regions, whereas the annealed model degrades substantially more in the medium and few-shot regions, in which the annealing gains reside.

The capacity view of subsection 3.2 makes a falsifiable prediction. If the ECA annealing phase stores tail-specific corrections as small deviations in residual weight precision, then distorting that precision should erase the few-shot gains more than the many-shot gains. We test the prediction following Hinton & Van Camp (1993). In their MDL view, and consistent with the model cost term in Equation 1, the amount of noise a weight tolerates is a direct measure of how many bits of information it carries: weights that must be specified precisely are expensive to describe, whereas weights that can be freely perturbed are cheap. We therefore apply zero-mean Gaussian noise to the weights, scaled per layer to the weight magnitude, and increase the noise level while monitoring the relative degradation of test MAE in each shot region. Noise is applied only to weight tensors of rank two or higher. Biases are left untouched, because perturbing the regression head bias applies a uniform shift to every prediction, which interacts asymmetrically with the shot regions and would confound the measurement with an effect unrelated to weight precision.

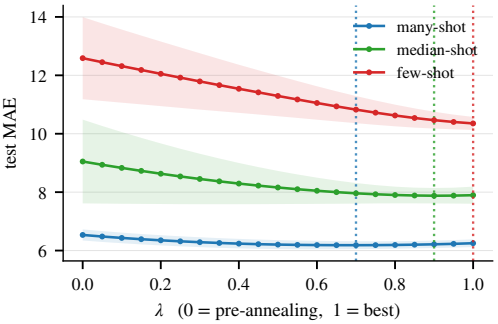


Figure 5: Weight interpolation $\theta(\lambda)$ between the pre-annealing checkpoint ($\lambda = 0$) and the annealed model ($\lambda = 1$) for V. + ECA on AgeDB-DIR. Lines show the mean over 10 seeds; bands show one σ. The shot regions attain their minima at different values of λ.

Figure 4 shows that the pre-annealing checkpoint degrades almost uniformly across shot regions. At a noise level of 71% of weight magnitude, many-shot and medium-shot follow nearly identical trajectories at roughly 42% degradation, and few-shot is only slightly worse at 46%. The annealed checkpoint behaves differently. Its many-shot trajectory is essentially unchanged at 39%, but medium-shot rises to 64% and few-shot to 85%. The separation between the two checkpoints therefore grows monotonically with the rarity of the region, from -3 points on many-shot to +21 on

medium-shot and +39 on few-shot. A control with statically rebalanced models confirms that this escalating signature does not arise from a generic sensitivity of sparse regions (Appendix L).

Interpolation. To characterize how the annealing phase writes tail knowledge, we linearly interpolate between the checkpoint taken at the start of the decay phase and the best validation checkpoint, $\theta(\lambda) = (1-\lambda)\,\theta_{\text{pre}} + \lambda\,\theta_{\text{post}}$, and evaluate many, medium, and few-shot MAE along $\lambda \in [0, 1]$ (Figure 5). The three shot regions attain their minima at different values of λ: many-shot error is minimized near $\lambda = 0.7$, medium-shot near $\lambda = 0.9$, and few-shot only at $\lambda = 1.0$. The ordering is monotone in region rarity, indicating that each additional increment of the annealing update exchanges a small amount of head accuracy for a larger gain in the tail rather than improving all regions jointly. Notably, the annealing update is not a pure trade-off: many-shot error decreased from $\lambda = 0$ up to roughly $\lambda = 0.7$ before increasing, so the majority of the update benefits the head as well. Only the final increment is tail-specific, with few-shot error still decreasing while many-shot error slightly rises. This marginal, tail-directed component is precisely the small, precision-fragile correction that the noise probe isolates. Because both endpoints lie on a single optimization trajectory, the interpolation remains inside one basin and requires no neuron alignment, analogous to weight-space interpolation between a pre-trained and a fine-tuned model (Wortsman et al., 2022).

5 Limitations and Future Work

ECA is among the strongest orthogonal additions on all three benchmarks, but its gains are not uniform. How it composes with other methods is dataset dependent: added to UVote, it trades many-shot for medium and few-shot performance on AgeDB-DIR, but mainly improves the medium region on IMDB-WIKI-DIR. The clearest failure case is Targa '13. ECA also requires control over the sampler and a schedule with a decay phase. Finally, the evidence for the weight-precision view is observational: the probes show where and when tail structure is written, not the information content of the weights. A control with statically rebalanced models (Appendix L) shows that sparse regions are mildly more perturbation-sensitive in general, but the escalating fragility of ECA appears only when tail-directed updates coincide with a decaying step size.

The interpolation result of subsection 4.4 suggests what we consider the most interesting extension. Our current practice runs the schedule to completion and selects the best checkpoint, or gives the option to halt the decay once many-shot performance degrades too far. Since every point on the segment between θ_{pre} and θ_{post} is itself a valid model, one could instead anneal deliberately past the point of many-shot degradation and choose the mixture coefficient λ post hoc on validation data. The head-tail trade-off then becomes a free, continuously tunable parameter that costs no retraining: a single run yields an entire family of models along the annealing trajectory. Taking this further, λ need not be global. A lightweight router could assign each sample its own coefficient at inference time, estimating from the input whether it belongs to a dense or a sparse target region and evaluating the corresponding interpolated weights, so that every prediction is served by the point on the trajectory that is optimal for its region.

6 Conclusion

We introduced Edge-Case Annealing, a training-schedule method for deep imbalanced regression that couples a data exposure curriculum one to one to the learning rate schedule, so that the effective training distribution reaches balance exactly as the step size vanishes. The method is grounded in a minimum description length view of network capacity: fine-grained structure for rare targets does not pay for its precision under the training distribution but does under the balanced evaluation distribution, and it can be retained, once the decaying learning rate lowers the gradient noise. Empirically, ECA is the strongest single orthogonal addition on AgeDB-DIR and IMDB-WIKI-DIR, composes with existing methods up to a new state of the art in the medium and few-shot regions, and is the best orthogonal method on MuViS-DIR with the smallest variance across seeds. Two probes support the underlying weight precision view: the annealing gains concentrate in the sparse regions, are erased first by weight noise, and accrue only in the final increments of the weight trajectory. We see these results less as another entry in the DIR leaderboard than as evidence that the training schedule is an underexplored axis for imbalanced regression: changing where and when a model sees its data can substitute for changing the model itself.

7 Reproducibility Statement

We release ECA as a pip-installable package together with the code that reproduces every main result in this paper at `https://anonymous.4open.science/r/edge-case-annealing-AAEE`; integrating ECA into an existing pipeline requires only the two steps described in Appendix E. All experiments use publicly available benchmarks with their established splits: AgeDB-DIR and IMDB-WIKI-DIR as defined by Yang et al. (2021), and the nine MuViS-DIR datasets as released by Brandt et al. (2026). Model architectures, including the per-dataset 1D-ResNet configurations for MuViS-DIR, are listed in subsection D.2. ECA introduces no hyperparameters of its own; the only tuned quantities are the warmup and decay fractions of the WSD schedule, whose search space, selection criterion, and final values are given in subsection D.3, with all remaining training settings in subsection D.4. Baseline results for AgeDB-DIR and MuViS-DIR are taken from ten-seed result tables, whereas IMDB-WIKI-DIR baselines are single runs reported by Jiang et al. (2024), as discussed in Appendix H. All our results are averaged over ten fixed seeds (42 to 51), with standard deviations reported in the appendix. Experiments ran on a single NVIDIA H100 (80 GB) GPU with CUDA 13.2.

8 Large Language Model Usage

In this work, we used generative AI tools as a general-purpose writing and coding assistant. We additionally used generative AI tools for providing feedback on research methodology and experiments, implementing methods, and supporting qualitative and thematic data analysis. We have not used generative AI tools for helping develop theoretical models or conceptual frameworks, proposing or refining hypotheses, designing research methodology or experiments, assisting with translation, cleaning and reformatting datasets, or interpreting results, and generating synthetic datasets, formulating mathematical claims, providing critical ingredients for proving mathematical claims, and assisting in the writing of proofs. Additionally, we used generative AI tools for creating or editing software code, creation of artifacts, drafting parts of the research paper, summarizing and analyzing existing literature, brainstorming, sourcing and searching for information, editing the research paper to improve readability, and identifying relevant literature. We have reviewed all AI-assisted work: we manually polished all AI-generated text, and we verified and tested all AI-generated code for correctness. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

References

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 41–48, New York, NY, USA, June 2009. Association for Computing Machinery. ISBN 978-1-60558-516-1. doi: 10.1145/1553374.1553380. URL `https://dl.acm.org/doi/10.1145/1553374.1553380`.

Cody Blakeney, Mansheej Paul, Brett W. Larsen, Sean Owen, and Jonathan Frankle. Does your data spark joy? Performance gains from domain upsampling at the end of training, June 2024. URL `http://arxiv.org/abs/2406.03476`. arXiv:2406.03476 [cs.LG].

Jens U. Brandt, Noah C. Puetz, Jobel Jose George, Niharika Vinay Kumar, Elena Raponi, Marc Hilbert, Thomas Bäck, and Thomas Bartz-Beielstein. MuViS: Multimodal Virtual Sensing Benchmark, March 2026. URL `http://arxiv.org/abs/2603.24602`. arXiv:2603.24602 [eess].

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss, October 2019. URL `http://arxiv.org/abs/1906.07413`. arXiv:1906.07413 [cs.LG].

Janez Demšar. Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, 7(1):1–30, 2006. ISSN 1533-7928. URL `http://jmlr.org/papers/v7/demsar06a.html`.

Zijian Dong, Yilei Wu, Chongyao Chen, Yingtian Zou, Yichi Zhang, and Juan Helen Zhou. Improve Representation for Imbalanced Regression through Geometric Constraints, March 2025. URL http://arxiv.org/abs/2503.00876. arXiv:2503.00876 [cs].

Yu Gong, Greg Mori, and Frederick Tung. RankSim: Ranking Similarity Regularization for Deep Imbalanced Regression, June 2022. URL http://arxiv.org/abs/2205.15236. arXiv:2205.15236 [cs].

Haibo He and Edwardo A. Garcia. Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284, September 2009. ISSN 1558-2191. doi: 10.1109/TKDE.2008.239. URL https://ieeexplore.ieee.org/document/5128907. Conference Name: IEEE Transactions on Knowledge and Data Engineering.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition, December 2015. URL http://arxiv.org/abs/1512.03385. arXiv:1512.03385 [cs.CV].

Steffen Herbold. Autorank: A Python package for automated ranking of classifiers. *Journal of Open Source Software*, 5(48):2173, April 2020. ISSN 2475-9066. doi: 10.21105/joss.02173. URL https://joss.theoj.org/papers/10.21105/joss.02173.

Geoffrey E. Hinton and Drew Van Camp. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the sixth annual conference on Computational learning theory - COLT '93*, pp. 5–13, Santa Cruz, California, United States, 1993. ACM Press. ISBN 978-0-89791-611-0. doi: 10.1145/168304.168306. URL http://portal.acm.org/citation.cfm?doid=168304.168306.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zheng Leng Thai, Kaihuo Zhang, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. MiniCPM: Unveiling the Potential of Small Language Models with Scalable Training Strategies, June 2024. URL http://arxiv.org/abs/2404.06395. arXiv:2404.06395 [cs.CL].

Alexander Hägele, Elie Bakouch, Atli Kosson, Loubna Ben Allal, Leandro Von Werra, and Martin Jaggi. Scaling Laws and Compute-Optimal Training Beyond Fixed Training Durations, October 2024. URL http://arxiv.org/abs/2405.18392. arXiv:2405.18392 [cs.LG].

Yuchang Jiang, Vivien Sainte Fare Garnot, Konrad Schindler, and Jan Dirk Wegner. Uncertainty Voting Ensemble for Imbalanced Deep Regression, October 2024. URL http://arxiv.org/abs/2305.15178. arXiv:2305.15178 [cs].

Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling Representation and Classifier for Long-Tailed Recognition, February 2020. URL http://arxiv.org/abs/1910.09217. arXiv:1910.09217 [cs.CV].

Mahsa Keramati, Lili Meng, and R. David Evans. ConR: Contrastive Regularizer for Deep Imbalanced Regression, 2023. URL https://arxiv.org/abs/2309.06651.

Nico Lang, Walter Jetz, Konrad Schindler, and Jan Dirk Wegner. A high-resolution canopy height model of the Earth. *Nature Ecology & Evolution*, 7(11):1778–1789, November 2023. ISSN 2397-334X. doi: 10.1038/s41559-023-02206-6. URL https://www.nature.com/articles/s41559-023-02206-6.

Jongin Lim, Sucheol Lee, Daeho Um, Sung-Un Park, and Jinwoo Shin. PRIME: Deep Imbalanced Regression with Proxies. In *Proceedings of the 42nd International Conference on Machine Learning*, pp. 37705–37729, Vancouver, Canada, October 2025. PMLR. URL https://proceedings.mlr.press/v267/lim25a.html.

Dekun Lin, Tailai Peng, Rui Chen, Xinran Xie, and Zhe Cui. Let Multi-classification Help Deep Imbalanced Regression. In Michael Wand, Kristína Malinovská, Jürgen Schmidhuber, and Igor V. Tetko (eds.), *Artificial Neural Networks and Machine Learning – ICANN 2024*, pp. 430–447, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-72338-4. doi: 10.1007/978-3-031-72338-4_29.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal Loss for Dense Object Detection, February 2018. URL http://arxiv.org/abs/1708.02002. arXiv:1708.02002 [cs].

Manil Maskey, Rahul Ramachandran, Muthukumaran Ramasubramanian, Iksha Gurung, Brian Freitag, Aaron Kaulfus, Drew Bollinger, Daniel J. Cecil, and Jeffrey Miller. Deepti: Deep-Learning-Based Tropical Cyclone Intensity Estimation System. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13:4271–4281, 2020. ISSN 2151-1535. doi: 10.1109/JSTARS.2020.3011907. URL https://ieeexplore.ieee.org/document/9149719.

Ismail Nejjar, Faez Ahmed, and Olga Fink. IM-Context: In-Context Learning for Imbalanced Regression Tasks, October 2024. URL http://arxiv.org/abs/2405.18202. arXiv:2405.18202 [cs].

Guangkun Nie, Gongzheng Tang, and Shenda Hong. Dist Loss: Enhancing Regression in Few-Shot Region through Distribution Distance Constraint, March 2025. URL http://arxiv.org/abs/2411.15216. arXiv:2411.15216 [cs].

Noah C. Puetz, Jens U. Brandt, Marc Hilbert, Elena Raponi, Thomas Bäck, and Thomas Bartz-Beielstein. Deconstructing deep imbalanced regression: a comprehensive review and experimental evaluation. *Artificial Intelligence Review*, 59(6):135, April 2026a. ISSN 1573-7462. doi: 10.1007/s10462-026-11570-1. URL https://doi.org/10.1007/s10462-026-11570-1.

Noah C. Puetz, Jens U. Brandt, Marc Hilbert, Elena Raponi, Thomas Bäck, and Thomas Bartz-Beielstein. Exposing Blind Spots in Deep Imbalanced Regression Evaluation, September 2026b. URL http://arxiv.org/abs/2609.25152. arXiv:2609.25152 [cs.LG].

Attila Reiss, Ina Indlekofer, Philip Schmidt, and Kristof Van Laerhoven. Deep PPG: Large-Scale Heart Rate Estimation with Convolutional Neural Networks. *Sensors*, 19(14), July 2019. ISSN 1424-8220. doi: 10.3390/s19143079. URL https://www.mdpi.com/1424-8220/19/14/3079.

Jiawei Ren, Mingyuan Zhang, Cunjun Yu, and Ziwei Liu. Balanced MSE for Imbalanced Visual Regression. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7916–7925, New Orleans, LA, USA, June 2022. IEEE. ISBN 978-1-6654-6946-3. doi: 10.1109/CVPR52688.2022.00777. URL https://ieeexplore.ieee.org/document/9879350/.

Abhinav Shrivastava, Abhinav Gupta, and Ross Girshick. Training Region-based Object Detectors with Online Hard Example Mining, April 2016. URL http://arxiv.org/abs/1604.03540. arXiv:1604.03540 [cs.CV].

Samuel L. Smith and Quoc V. Le. A Bayesian Perspective on Generalization and Stochastic Gradient Descent, February 2018. URL http://arxiv.org/abs/1710.06451. arXiv:1710.06451 [cs.LG].

Michael Steininger, Konstantin Kobs, Padraig Davidson, Anna Krause, and Andreas Hotho. Density-based weighting for imbalanced regression. *Machine Learning*, 110(8):2187–2211, August 2021. ISSN 1573-0565. doi: 10.1007/s10994-021-06023-5. URL https://doi.org/10.1007/s10994-021-06023-5.

Ziyan Wang and Hao Wang. Variational Imbalanced Regression: Fair Uncertainty Quantification via Probabilistic Smoothing, October 2023. URL http://arxiv.org/abs/2306.06599. arXiv:2306.06599 [cs, stat].

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7949–7961, June 2022. doi: 10.1109/CVPR52688.2022.00780. URL https://ieeexplore.ieee.org/document/9878622/. Conference Name: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) ISBN: 9781665469463.

Haipeng Xiong and Angela Yao. Deep Imbalanced Regression via Hierarchical Classification Adjustment. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23721–23730, June 2024. doi: 10.1109/CVPR52733.2024.02239. URL https://ieeexplore.ieee.org/document/10655108.

Yuzhe Yang, Kaiwen Zha, Ying-Cong Chen, Hao Wang, and Dina Katabi. Delving into Deep Imbalanced Regression, May 2021. URL http://arxiv.org/abs/2102.09554. arXiv:2102.09554 [cs].

Kaiwen Zha, Peng Cao, Jeany Son, Yuzhe Yang, and Dina Katabi. Rank-N-Contrast: Learning Continuous Representations for Regression, October 2023. URL http://arxiv.org/abs/2210.01189. arXiv:2210.01189 [cs].

Shihao Zhang, Linlin Yang, Michael Bi Mi, Xiaoxu Zheng, and Angela Yao. Improving Deep Regression with Ordinal Entropy, February 2023. URL http://arxiv.org/abs/2301.08915. arXiv:2301.08915 [cs].

A Appendix Outline

This appendix is organized as follows. Appendix B lists the mathematical notation used throughout the paper, and Appendix C provides additional background on the current DIR literature, including an overview of the datasets and backbone models used in recent publications (Table 7). Appendix D then details our evaluation setup, and Appendix E describes how to integrate ECA into an existing training pipeline. The remaining sections provide extended results for the individual experiments on AgeDB-DIR (Appendix G), IMDB-WIKI-DIR (Appendix H), and MuViS-DIR (Appendix I), followed by the ECA ablation (Appendix J) and the additional probes (Appendix K and Appendix L), as well as a significance analysis (Appendix M).

B Mathematical Notations

Table 6: Mathematical notation used throughout the paper.

Symbol	Meaning
Problem setting	
$\mathcal{X}$	Input space.
$\mathcal{Y}$	Continuous target space, $\mathcal{Y} \subseteq \mathbb{R}$.
x	Input features, $x \in \mathcal{X}$.
y	Continuous scalar target, $y \in \mathcal{Y}$.
$\mathcal{D}$	Training set of input–target pairs, $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{M}$.
M	Number of training samples.
i	Sample index, with $i \in \{1, \dots, M\}$.
f_θ	Deep regression model with parameters θ, $f_\theta \colon \mathcal{X} \to \mathcal{Y}$.
θ	Trainable model parameters.
$\mathcal{L}$	Training loss, $\mathcal{L}(f_\theta(x), y)$; the plain L_1 objective in all experiments.
$p_\theta(y \mid x)$	Predictive distribution of the model.
$p_{\text{train}}(y)$	Empirical target distribution of the training set, highly non-uniform in the DIR setting.
$p_{\text{eval}}(y)$	Desired evaluation distribution over the target space, more balanced than $p_{\text{train}}(y)$.
Target binning	
B	Number of equal-width bins partitioning the supported target range.
b	Bin index, with $b \in \{1, \dots, B\}$.
$\mathcal{B}_b$	The b-th target bin; the partition is $\{\mathcal{B}_1, \dots, \mathcal{B}_B\}$.
m_b	Number of training samples whose target falls into bin $\mathcal{B}_b$.
$m_{\max}$	Largest bin count, $m_{\max} = \max_b m_b$.
Capacity view (MDL)	
$L(\mathcal{D})$	Total description length of the data under a two-part code.

Continued on next page

Symbol	Meaning
$L(\theta)$	Model cost: bits required to specify the weights, growing with the precision to which they must be pinned down.
$L(\mathcal{D} \mid \theta)$	Data cost: bits required to encode the training targets under p_θ, equivalently the cost of encoding the residuals. For the L_1 objective, p_θ is a Laplace likelihood with fixed scale, and $L(\mathcal{D} \mid \theta)$ equals the L_1 loss up to a constant.
Learning rate and data schedule	
n	Training step index.
N	Total number of training steps.
N_{warmup}	Number of warmup steps.
N_{decay}	Number of decay steps.
$\eta(n)$	Learning rate at step n under the WSD schedule.
$\eta_{\max}$	Peak learning rate.
f	Monotonically decreasing decay shape, with $f(N - N_{\text{decay}}) = 1$ and $f(N) = 0$; cosine in the main experiments.
$\xi(n)$	Annealing coefficient of ECA, mirroring the learning rate schedule, $\xi(n) = \eta(n)/\eta_{\max} \in [0, 1]$.
$C(n)$	Ceiling on the target histogram at step n, descending from $m_{\max}$ to $c_{\min}$.
$c_{\min}$	Minimum bin size the ceiling decays to; $c_{\min} = 1$ unless stated otherwise.
$c_b(n)$	Clipped sample count of bin $\mathcal{B}_b$ at step n, $c_b(n) = \min(m_b, C(n))$.
$p_n(\mathcal{B}_b)$	Effective training distribution over bins at step n, with $p_n(\mathcal{B}_b) \propto c_b(n)$.
Probes	
θ_{pre}	Checkpoint taken at the start of the decay phase, before annealing.
θ_{post}	Best validation checkpoint after annealing.
λ	Interpolation coefficient between the two checkpoints, $\lambda \in [0, 1]$.
$\theta(\lambda)$	Interpolated weights, $\theta(\lambda) = (1 - \lambda)\,\theta_{\text{pre}} + \lambda\,\theta_{\text{post}}$.
$\|\Delta\theta\|/\|\theta_{\text{pre}}\|$	Relative displacement between the two checkpoints.

C Related Work

Table 7 summarizes the DIR method landscape. For each method, it lists the family it belongs to (algorithm-level, representation-learning, or architectural), the venue, the datasets and backbone models used in the original evaluation, whether the method is orthogonal, and whether public training code is available.

Table 7: Overview of DIR methods, datasets, and backbone architectures. Datasets and models in **bold** indicate that public training code is available for the corresponding combination. "Algo." stands for algorithm-level methods, "Repr." for representation-learning methods, and "Archit." for architectural methods. The *Orth.* column indicates whether the method is orthogonal to the backbone architecture and can be combined with other methods.

Type	Method	Venue	Datasets	Models	Orth.	Repo
Algo.	LDS (Yang et al., 2021)	ICML 2021	**IMDB-WIKI-DIR** **AgeDB-DIR** **STS-B-DIR** **NYUD2-DIR** SHHS-DIR	**ResNet-50** **BiLSTM+GloVe** CNN-RNN	✓	GitHub
	Balanced MSE (Ren et al., 2022)	CVPR 2022	**IMDB-WIKI-DIR** **NYUD2-DIR** IHMR	**ResNet-50** SPIN	✓	GitHub
	VIR (Wang & Wang, 2023)	NeurIPS 2023	**IMDB-WIKI-DIR** **AgeDB-DIR** STS-B-DIR NYUD2-DIR	**ResNet-50** BiLSTM+GloVe	✗	GitHub
	DenseLoss (Steininger et al., 2021)	*Mach. Learn.* 2021	Synthetic Data	MLP	✓	GitHub

Continued on next page

Table 7 – continued from previous page

Type	Method	Venue	Datasets	Models	Orth.	Repo
	Focal-R (Yang et al., 2021; Lin et al., 2018)	ICML 2021	**IMDB-WIKI-DIR** **AgeDB-DIR** **STS-B-DIR** **NYUD2-DIR** SHHS-DIR	**ResNet-50** **BiLSTM+GloVe** CNN-RNN	✓	GitHub
	INV & SQINV (Yang et al., 2021)	ICML 2021	**IMDB-WIKI-DIR** **AgeDB-DIR** **STS-B-DIR** **NYUD2-DIR** SHHS-DIR	**ResNet-50** **BiLSTM+GloVe** CNN-RNN	✓	GitHub
	RRT (Yang et al., 2021)	ICML 2021	**IMDB-WIKI-DIR** **AgeDB-DIR** **STS-B-DIR** **NYUD2-DIR** SHHS-DIR	**ResNet-50** **BiLSTM+GloVe** CNN-RNN	✓	GitHub
	Dist Loss (Nie et al., 2025)	ICLR 2025	**IMDB-WIKI-DIR** **AgeDB-DIR** ECG-K-DIR	**ResNet-50** Net1D	✓	GitHub
Repr.	FDS (Yang et al., 2021)	ICML 2021	**IMDB-WIKI-DIR** **AgeDB-DIR** **STS-B-DIR** **NYUD2-DIR** SHHS-DIR	**ResNet-50** **BiLSTM+GloVe** CNN-RNN	✓	GitHub
	RankSim (Gong et al., 2022)	ICML 2022	**IMDB-WIKI-DIR** **AgeDB-DIR** **STS-B-DIR**	**ResNet-50** **BiLSTM+GloVe**	✓	GitHub
	ConR (Keramati et al., 2023)	ICLR 2024	**IMDB-WIKI-DIR** **AgeDB-DIR** **NYUD2-DIR** MPIIGaze-DIR	**ResNet-50** LeNet	✓	GitHub
	RnC (Zha et al., 2023)	NeurIPS 2023	IMDB-WIKI-DIR **AgeDB-DIR** TUAB MPIIGaze-DIR SkyFinder	**ResNet-18** **ResNet-50**	✗	GitHub
	Geom. Rep. (Dong et al., 2025)	CVPR 2025	UCI-DIR AgeDB-DIR IMDB-WIKI-DIR **STS-B-DIR**	ResNet-50 **BiLSTM+GloVe** MLP	✓	GitHub
	Ordinal Ent. (Zhang et al., 2023)	ICLR 2023	**Synthetic data** NYU-Depth-v2 AgeDB-DIR SHTech	**MLP** NeW-CRFs ResNet-50 DeepONet CSRNet	✓	GitHub
	PRIME (Lim et al., 2025)	ICML 2025	AgeDB-DIR IMDB-WIKI-DIR NYUD2-DIR STS-B-DIR	ResNet-50 BiLSTM+GloVe	✓	—
Archit.	UVOTE (Jiang et al., 2024)	GCPR 2024	**IMDB-WIKI-DIR** **AgeDB-DIR** Wind **STS-B-DIR**	**ResNet-50** **ResNet-18** **BiLSTM**	✗	GitHub
	HCA (Xiong & Yao, 2024)	CVPR 2024	IMDB-WIKI-DIR AgeDB-DIR NYUDv2-DIR SHTech	ResNet-50 VGG16	✗	GitHub
	Multi-Class. (Lin et al., 2024)	ICANN 2024	IMDB-WIKI-DIR AgeDB-DIR STS-B-DIR	ResNet-50 BiLSTM+GloVe	✗	—

Continued on next page

Table 7 – continued from previous page

Type	Method	Venue	Datasets	Models	Orth.	Repo
	Im-Context (Nejjar et al., 2024)	TMLR 2024	IMDB-WIKI-DIR AgeDB-DIR STS-B-DIR **Boston** **Concrete** **Abalone** **Communities** **Kin8nm** **Airfoil**	GPT2 **PFN**	✗	GitHub

D Experimental Setup

D.1 Evaluation Outline

Evaluation protocols in the DIR literature are fragmented. Recent methods such as SRL (Dong et al., 2025), PRIME (Lim et al., 2025), and Dist Loss (Nie et al., 2025) each evaluate on different datasets and report different subsets of prior techniques, which complicates direct comparison. We therefore anchor our evaluation in the result set reported for UVote (Jiang et al., 2024), which constitutes the strongest published method to date and also emerges as the best performer across the methods compared by Puetz et al. (2026a). For completeness, Appendix F compares ECA with the reported results of these three recent methods.

Beyond fragmentation, Puetz et al. (2026b) identify three systematic weaknesses of current DIR evaluation: the benchmark composition is narrowly focused on computer vision, the standard partition of the target range into many, medium, and few-shot regions is arbitrary, and most methods report results from a single seed. Our evaluation design responds to all three: We extend the benchmark suite with the MuViS-DIR time series datasets, report the binning-free bMASE protocol where no established shot partition exists, and average all results across ten seeds. For AgeDB-DIR and MuViS-DIR we additionally report the mean and standard deviation of the competing methods, since the result tables we build on were likewise evaluated across ten seeds.

D.2 Model Details

For AgeDB-DIR and IMDB-WIKI-DIR we use a standard ResNet-50 backbone (He et al., 2015) with 23.5M parameters, matching the architecture used throughout the DIR literature. For MuViS-DIR we adopt the tuned 1D-ResNet configurations of Brandt et al. (2026), listed per dataset in Table 8.

Table 8: Per-dataset 1D-ResNet configurations for MuViS-DIR, taken from Brandt et al. (2026). Dropout is applied within the residual blocks ($\text{dropout}_{\text{res}}$) and before the fully connected layer ($\text{dropout}_{\text{fc}}$). The result tables use the following short names: Beijing PM10 = PM10, Beijing PM2.5 = PM2.5, PPG DaLiA = Heart Rate, Panasonic 18650 = Battery SoC, Monterey '13 = Monterey, TEP = Chem. Conc., and Vehicle Dynamics = Tyre Temp.

Dataset	Feat. dim	Filters	FC units	Res. blocks	$\text{Dropout}_{\text{res}}$	$\text{Dropout}_{\text{fc}}$
Beijing PM10	9	156	512	4	0.154	0.028
Beijing PM2.5	9	256	64	4	0.030	0.060
PPG DaLiA	6	256	512	4	0.009	0.004
Panasonic 18650	7	256	128	2	0.165	0.00005
Monterey '13	22	256	32	4	0.036	0.0005
Targa '13	22	256	32	4	0.036	0.0005
Targa '14	22	256	32	4	0.036	0.0005
TEP	33	256	256	4	0.093	0.095
Vehicle Dynamics	11	32	512	4	0.060	0.014

D.3 ECA PARAMETERS

ECA inherits all of its hyperparameters from the WSD schedule and introduces none of its own: the annealing coefficient mirrors the learning rate throughout (`mimic_lr`), the clipped subset is re-drawn at every epoch, and the ceiling always decays to a minimum bin size of $c_{\min} = 1$. The peak learning rate is set to the established vanilla learning rate of the respective dataset, and the decay shape is cosine everywhere. The only quantities we tune are the warmup and decay fractions, selected per benchmark by a grid search over $\{0.0, 0.1, 0.2, 0.3, 0.4, 0.5\}$ for both, with the validation loss as the selection criterion. Table 9 lists the resulting configurations. The balanced exposure during warmup follows directly from mirroring the learning-rate schedule. The capacity argument motivating ECA concerns the decay phase: after training on the natural distribution at full step size, progressively rebalanced exposure is paired with increasingly small updates that can act as fine-grained corrections in residual weight precision. A longer decay phase (fraction 0.7, outside the grid used for selection) yields slightly better results on IMDB-WIKI-DIR (7.37 / 6.71 / 12.69 / 23.64), indicating that ECA is not sensitive to the exact decay length.

Table 9: ECA and WSD schedule configurations per benchmark. All remaining settings are shared across datasets.

Dataset	Peak LR	Decay shape	Warmup fract.	Decay fract.	Final LR factor	Min. bin size
AgeDB-DIR	5e-4	cosine	0.2	0.2	0.0	1
IMDB-WIKI-DIR	1e-3	cosine	0.1	0.4	0.0	1
MuViS-DIR	5e-4	cosine	0.2	0.1	0.0	1

D.4 TRAINING DETAILS

On AgeDB-DIR we train for 90 epochs with a batch size of 64, and on IMDB-WIKI-DIR for 90 epochs with a batch size of 256. We use the Adam optimizer throughout. For MuViS-DIR we adopt the tuned configuration of Brandt et al. (2026) and train for 120 epochs with a batch size of 64. All experiments were executed on a single NVIDIA H100 (80 GB) GPU with CUDA 13.2. In the case of ECA the numbers of epochs change but the total number of steps is always equivalent to the non ECA run, Table 10 shows the full overview of training steps per dataset.

Table 10: Training set sizes and total training steps for each dataset.

Dataset	n_{train}	Total steps
AgeDB-DIR	12,208	22,800
IMDB-WIKI-DIR	191,509	67,320
PM10	10,726	20,040
PM2.5	10,726	20,040
Heart Rate	46,581	87,240
Battery SoC	179,844	337,200
Monterey	108,699	203,760
Targa '13	30,168	56,520
Targa '14	42,065	78,840
Chem. Conc.	216,450	405,840
Tyre Temp.	1,245	2,280

D.5 TARGET BINNING

ECA uses equal-width target bins only to construct the data exposure schedule; neither the regression objective nor the model output is discretized. Importantly, we do not tune the number of bins for ECA. Instead, we retain the target discretization established by the respective benchmark.

For AgeDB-DIR and IMDB-WIKI-DIR, we use the binning introduced by Yang et al. (2021), in which the age target is partitioned at the resolution used by the original DIR benchmarks. This results in $B = 100$ bins for AgeDB-DIR and $B = 127$ bins for IMDB-WIKI-DIR. For the nine MuViS-DIR datasets, we retain the dataset-specific discretizations defined by Puetz et al. (2026b). The resulting numbers of bins are listed in Table 11.

The binning therefore serves as a fixed density discretization supplied by the benchmark rather than as a degree of freedom optimized for ECA. This also keeps the ECA configuration aligned

Table 11: Target discretizations used by ECA. The number of bins is inherited from the established benchmark configuration and is not tuned for ECA.

Dataset	Number of bins B	Source
AgeDB-DIR	100	Yang et al. (2021)
IMDB-WIKI-DIR	127	Yang et al. (2021)
PM10	63	Puetz et al. (2026b)
PM2.5	73	Puetz et al. (2026b)
Heart Rate	61	Puetz et al. (2026b)
Battery SoC	64	Puetz et al. (2026b)
Monterey	228	Puetz et al. (2026b)
Targa '13	109	Puetz et al. (2026b)
Targa '14	55	Puetz et al. (2026b)
Chem. Conc.	245	Puetz et al. (2026b)
Tyre Temp.	10	Puetz et al. (2026b)

with the target-density definitions used in the respective benchmark literature.

E Practitioner's Guide to ECA Integration

Integrating Edge-Case Annealing into an existing training pipeline requires only two modifications, neither of which alters the model, the loss function, or the optimizer. ECA is available as a standalone package (`pip install edge-case-annealing`).

Step 1: Wrap the dataset and dataloader. Replace the standard `DataLoader` with the ECA wrapper. The `setup_eca` convenience function accepts the original dataset, an array of per-sample labels (used to group samples into bins), and the total number of training steps:

```
from edge_case_annealing import setup_eca

eca_dataset, batch_sampler = setup_eca(
    train_dataset,
    labels=train_labels,
    batch_size=64,
    n_iterations=total_steps,
    decay_type='mimic_lr',    # tie the clip schedule to the LR decay
)

train_loader = DataLoader(eca_dataset, batch_sampler=batch_sampler)
```

Listing 1: Setting up ECA.

When using `decay_type='mimic_lr'`, ECA mirrors the learning rate schedule directly and no additional hyperparameters are needed. Alternatively, an independent schedule can be specified via `fract_decay` and one of the built-in shapes (`'log'`, `'linear'`, `'sqrt'`, `'square'`).

Step 2: Step the schedule in the training loop. After each optimizer step, advance the annealing schedule by calling `eca_dataset.step()`. This is the only addition to the inner loop; the batch sampler automatically adapts to the shrinking active set:

```
for inputs, targets in train_loader:
    loss = criterion(model(inputs), targets)
    loss.backward()
    optimizer.step()
    scheduler.step()
    eca_dataset.step(
        lr_progress=1.0 - scheduler.get_last_lr()[0] / max_lr
        )
```

Listing 2: Stepping ECA in the training loop.

No changes are needed to the validation or test pipeline. The entire mechanism is implemented as a dataset wrapper and a custom batch sampler, which keeps ECA agnostic to model architecture, loss function, and optimizer choice.

F Comparison with Other Methods from the Literature

Our main comparison is anchored in the ten-seed result tables of Puetz et al. (2026a) and in UVote (Jiang et al., 2024), the strongest published method (subsection D.1). For completeness, Table 12 and Table 13 additionally list three recent orthogonal methods: Dist Loss (Nie et al., 2025), SRL

(Dong et al., 2025), and PRIME (Lim et al., 2025). Their numbers are taken from the original publications. Dist Loss and SRL report single runs, PRIME reports standard deviations but provides no public code, and none of the three compares against UVote. The results are therefore not strictly comparable to our ten-seed evaluation and are not included in the main tables.

Table 12: Comparison with recent methods on AgeDB-DIR (MAE). Numbers for Dist Loss, SRL, and PRIME are taken from the original publications; all other rows are means over ten seeds. **Bold** marks the best value per column within each group; V. abbreviates Vanilla.

Method	All	Many	Med	Few
Orthogonal methods				
Vanilla	7.68	6.67	8.99	13.52
V. + Dist Loss (Nie et al., 2025)	7.64	7.57	7.32	**9.12**
V. + SRL (Dong et al., 2025)	7.22	6.64	8.28	9.81
V. + PRIME (Lim et al., 2025)	$7.06_{\pm 0.09}$	$6.67_{\pm 0.09}$	$\mathbf{7.27}_{\pm 0.25}$	$9.91_{\pm 0.10}$
V. + ECA (ours)	$\mathbf{6.91}_{\pm 0.08}$	$\mathbf{6.25}_{\pm 0.13}$	$7.90_{\pm 0.29}$	$10.35_{\pm 0.22}$
Non-orthogonal methods				
UVote	$\mathbf{6.77}_{\pm 0.08}$	$\mathbf{6.38}_{\pm 0.12}$	$7.32_{\pm 0.28}$	$8.91_{\pm 0.37}$
UVote + ECA (ours)	$6.96_{\pm 0.05}$	$6.88_{\pm 0.16}$	$\mathbf{6.88}_{\pm 0.38}$	$\mathbf{7.98}_{\pm 0.31}$

Table 13: Comparison with recent methods on IMDB-WIKI-DIR (MAE). Numbers for Dist Loss, SRL, PRIME, and Vanilla are taken from the original publications and Jiang et al. (2024); ECA rows are means over ten seeds. **Bold** marks the best value per column within each group; V. abbreviates Vanilla.

Method	All	Many	Med	Few
Orthogonal methods				
Vanilla	8.06	7.23	15.12	26.33
V. + Dist Loss (Nie et al., 2025)	8.03	7.46	12.61	22.52
V. + SRL (Dong et al., 2025)	7.69	7.08	12.65	22.78
V. + PRIME (Lim et al., 2025)	$\mathbf{7.36}_{\pm 0.05}$	$\mathbf{6.73}_{\pm 0.06}$	$12.48_{\pm 0.23}$	$23.01_{\pm 0.92}$
V. + ECA (ours)	$7.40_{\pm 0.18}$	$\mathbf{6.73}_{\pm 0.19}$	$12.82_{\pm 0.18}$	$24.00_{\pm 0.44}$
V. + SQINV + ECA (ours)	$7.56_{\pm 0.15}$	$7.05_{\pm 0.17}$	$\mathbf{11.51}_{\pm 0.32}$	$\mathbf{21.74}_{\pm 0.43}$
Non-orthogonal methods				
UVote	**7.36**	**6.81**	11.78	**20.96**
UVote + ECA (ours)	$7.67_{\pm 0.33}$	$7.26_{\pm 0.38}$	$\mathbf{10.62}_{\pm 0.42}$	$21.16_{\pm 0.62}$

On AgeDB-DIR, V. + ECA attains the best overall and many-shot MAE among all orthogonal methods, including the three recent ones. In the medium and few-shot regions, PRIME and Dist Loss are ahead. Dist Loss illustrates the trade-off discussed in section 2: its few-shot gain comes with a many-shot MAE of 7.57, clearly worse than Vanilla (6.67). On IMDB-WIKI-DIR, V. + ECA matches PRIME in the many-shot region (6.73) and is within one standard deviation of it overall (7.40 versus 7.36), while PRIME, Dist Loss, and SRL are ahead in the medium and few-shot regions. Combined with SQINV, however, ECA attains the best medium and few-shot MAE of all orthogonal methods (11.51 and 21.74), ahead of all three recent methods, while remaining fully orthogonal.

Two further observations support our choice of UVote as the reference. First, UVote alone already outperforms all three recent methods in the few-shot region of AgeDB-DIR and in the medium and few-shot regions of IMDB-WIKI-DIR. Second, UVote + ECA attains the best medium and few-shot MAE of all methods in both comparisons, except for the few-shot region of IMDB-WIKI-DIR, where it is within 0.2 MAE of UVote alone. ECA thus composes with the strongest available method into results that none of the recent single methods reaches in the sparse regions.

G AgeDB-DIR Appendix

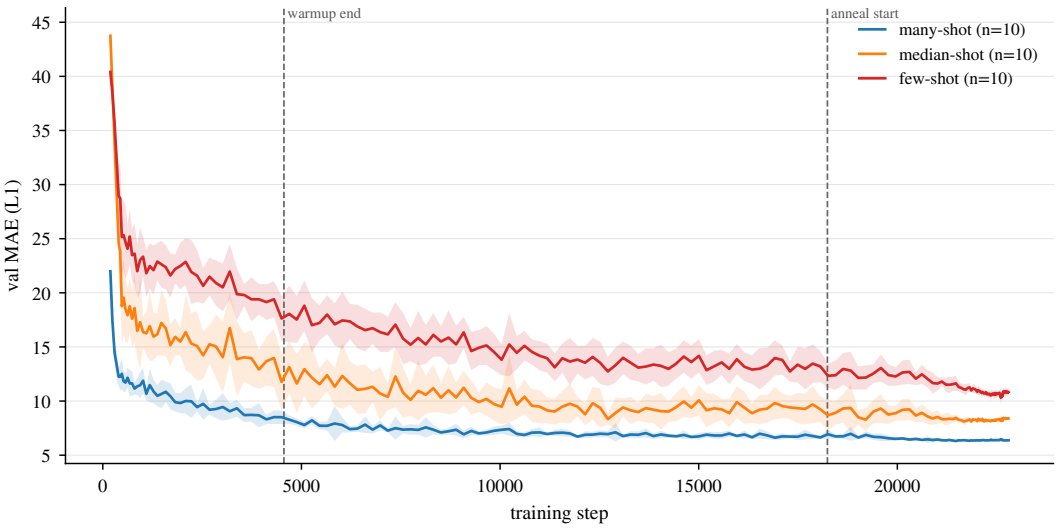


Figure 6: Course of the validation loss seperated for many-, medium and few-shot val loss for V. + ECA across ten seeds on AgeDB-DIR. The vertical dashed lines show the end of the warmup-phase the the beginning of the annealing phase.

Table 14: Results on AgeDB-DIR (MAE, mean ± std over ten seeds). Improvements are relative to the reference row given in the last column.

#	Method	MAE				Improv. (%)				vs. #
		All	Many	Med	Few	All	Many	Med	Few	
1	Vanilla	7.68 ± 0.15	6.67 ± 0.14	8.99 ± 0.18	13.52 ± 0.62	–	–	–	–	–
2	V. + LDS	7.69 ± 0.19	7.03 ± 0.20	8.64 ± 0.43	11.32 ± 0.33	-0.13	-5.40	3.89	16.27	1
3	V. + FDS	7.69 ± 0.19	6.67 ± 0.16	9.09 ± 0.26	13.50 ± 0.69	-0.13	0.00	-1.11	0.15	1
4	V. + LDS + FDS	7.75 ± 0.17	7.07 ± 0.16	8.71 ± 0.24	11.49 ± 0.55	-0.78	-0.57	-0.81	-1.50	2
5	V. + Focal L_1	7.45 ± 0.10	6.61 ± 0.13	8.69 ± 0.21	12.04 ± 0.35	2.99	0.90	3.34	10.95	1
6	V. + SQINV	7.69 ± 0.17	7.02 ± 0.16	8.65 ± 0.30	11.42 ± 0.62	-0.13	-5.25	3.78	15.53	1
7	V. + RRT	7.36 ± 0.23	6.58 ± 0.24	8.30 ± 0.20	12.15 ± 0.42	4.17	1.35	7.68	10.13	1
8	V. + RRT + BNI	7.37 ± 0.19	6.57 ± 0.17	8.29 ± 0.23	12.33 ± 0.60	-0.14	0.15	0.12	-1.48	7
9	V. + RRT + BMC	7.30 ± 0.12	6.53 ± 0.12	8.19 ± 0.16	12.17 ± 0.43	0.82	0.76	1.33	-0.16	7
10	V. + RankSim	7.28 ± 0.05	6.39 ± 0.08	8.49 ± 0.18	12.32 ± 0.24	5.21	4.20	5.56	8.88	1
11	V. + LDS + RankSim	7.21 ± 0.08	6.61 ± 0.10	8.07 ± 0.11	10.40 ± 0.28	6.24	5.97	6.60	8.13	2
12	V. + FDS + RankSim	7.33 ± 0.09	6.43 ± 0.08	8.52 ± 0.22	12.55 ± 0.47	4.68	3.60	6.27	7.04	3
13	V. + LDS + FDS + RankSim	7.19 ± 0.09	6.62 ± 0.08	8.00 ± 0.17	10.33 ± 0.37	7.23	6.36	8.15	10.10	4
14	V. + Focal L_1 + RankSim	7.29 ± 0.08	6.41 ± 0.09	8.56 ± 0.20	12.08 ± 0.32	2.15	3.03	1.50	-0.33	5
15	V. + SQINV + RankSim	7.16 ± 0.13	6.57 ± 0.13	8.02 ± 0.21	10.27 ± 0.37	6.89	6.41	7.28	10.07	6
16	V. + RRT + RankSim	7.25 ± 0.10	6.51 ± 0.16	8.13 ± 0.14	11.86 ± 0.38	1.49	1.06	2.05	2.39	7
17	V. + ConR	7.31 ± 0.13	6.38 ± 0.11	8.58 ± 0.23	12.51 ± 0.45	4.82	4.35	4.56	7.47	1
18	V. + LDS + ConR	7.30 ± 0.09	6.71 ± 0.10	8.10 ± 0.15	10.69 ± 0.37	5.07	4.55	6.25	5.57	2
19	V. + FDS + ConR	7.32 ± 0.07	6.41 ± 0.06	8.59 ± 0.17	12.41 ± 0.27	4.81	3.90	5.50	8.07	3
20	V. + LDS + FDS + ConR	7.32 ± 0.13	6.73 ± 0.10	8.10 ± 0.22	10.72 ± 0.43	5.55	4.81	7.00	6.70	4
21	RnC	6.50 ± 0.08	5.83 ± 0.14	7.61 ± 0.20	9.73 ± 0.63	15.36	12.59	15.35	28.03	1
22	UVote	6.77 ± 0.08	6.38 ± 0.12	7.32 ± 0.28	8.91 ± 0.37	11.85	4.35	18.58	34.10	1
23	VIR	6.83 ± 0.06	6.29 ± 0.13	7.54 ± 0.16	9.95 ± 0.36	11.07	5.70	16.13	26.41	1
24	V. + ECA (ours)	6.91 ± 0.08	6.25 ± 0.13	7.90 ± 0.29	10.35 ± 0.22	10.03	6.30	12.12	23.45	1
25	V. + ConR + ECA (ours)	7.50 ± 0.07	6.99 ± 0.12	7.86 ± 0.21	11.31 ± 0.40	-2.60	-9.56	8.39	9.59	17
26	UVote + ECA (ours)	6.96 ± 0.05	6.88 ± 0.16	6.88 ± 0.38	7.98 ± 0.31	-2.81	-7.84	6.01	10.44	22

Table 14 reports the standard deviations of the MAEs for all shot regions and shows the additional experiment of combining ECA with ConR (line 25). Then Figure 6 shows the course of learning across ten seeds for the V. + ECA model on AgeDB-DIR. Two things can be observed. First, roughly halfway through the annealing phase, the medium and few-shot validation MAEs enter a second descent after having plateaued for the preceding few thousand steps, while the many-shot

MAE remains stable. Second, the standard deviation across runs shrinks steadily toward the end of training.

H IMDB-WIKI-DIR APPENDIX

The evaluation on IMDB-WIKI-DIR differs from the other benchmarks in that the baseline numbers are single, non-repeated runs taken from Jiang et al. (2024). A single run takes roughly four hours on our H100, i.e., about 40 hours per method across ten seeds and more than 20 days of compute for the thirteen baselines alone. We therefore compare our seeded results against the reported single-run numbers. Table 15 lists the mean and standard deviation of our three experiments as also more reference models taken from Jiang et al. (2024).

Table 15: Results on IMDB-WIKI-DIR (MAE). Baseline numbers (rows 1 to 13) are single runs from Jiang et al. (2024); our results (rows 14 to 16) are mean $\pm$ std over ten seeds. Improvements relative to the reference row in the last column; V. abbreviates Vanilla.

#	Method	MAE				Improv. (%)				vs. #
		All	Many	Med	Few	All	Many	Med	Few	
1	Vanilla	8.06	7.23	15.12	26.33	–	–	–	–	–
2	V. + RankSim	7.72	6.93	14.48	25.38	4.22	4.15	4.23	3.61	1
3	V. + RankSim + RRT	7.55	6.83	13.47	24.72	2.20	1.44	6.98	2.60	2
4	V. + RankSim + SQINV	7.42	6.84	12.12	22.13	3.89	1.30	16.30	12.81	2
5	V. + RankSim + RRT + LDS + FDS	7.37	6.80	11.80	23.11	2.38	0.44	12.40	6.51	3
6	V. + RankSim + SQINV + LDS + FDS	7.69	7.13	12.30	21.43	-3.64	-4.24	-1.49	3.16	4
7	V. + RRT	7.81	7.07	14.06	25.13	3.10	2.21	7.01	4.56	1
8	V. + RRT + LDS + FDS	7.65	7.06	12.41	23.51	2.05	0.14	11.74	6.45	7
9	V. + SQINV	7.87	7.24	12.44	22.76	2.36	-0.14	17.72	13.56	1
10	V. + SQINV + LDS + FDS	7.78	7.20	12.61	22.19	1.14	0.55	-1.37	2.50	9
11	V. + BMSE	8.08	7.52	12.47	23.29	-0.25	-4.01	17.53	11.55	1
12	V. + DenseWeight	7.85	7.14	13.70	25.38	2.61	1.24	9.39	3.61	1
13	UVote	7.36	6.81	11.78	20.96	8.68	5.81	22.09	20.40	1
14	V. + ECA (ours)	7.40 $\pm$ 0.177	6.73 $\pm$ 0.186	12.82 $\pm$ 0.182	24.00 $\pm$ 0.444	8.24 $\pm$ 2.19	6.89 $\pm$ 2.57	15.21 $\pm$ 1.20	8.85 $\pm$ 1.68	1
15	V. + SQINV + ECA (ours)	7.56 $\pm$ 0.145	7.05 $\pm$ 0.172	11.51 $\pm$ 0.322	21.74 $\pm$ 0.427	3.92 $\pm$ 1.84	2.56 $\pm$ 2.38	7.47 $\pm$ 2.59	4.46 $\pm$ 1.88	9
16	UVote + ECA (ours)	7.67 $\pm$ 0.329	7.26 $\pm$ 0.380	10.62 $\pm$ 0.420	21.16 $\pm$ 0.623	-4.24 $\pm$ 4.47	-6.63 $\pm$ 5.58	9.85 $\pm$ 3.57	-0.96 $\pm$ 2.97	13

In addition, Table 16 compares our best runs against the reported numbers. Which run is best is not uniquely determined, since the per-region MAEs of the protocol by Yang et al. (2021) do not induce a single ranking (see subsection D.1); we therefore select the run with the lowest few-shot MAE. This selects seed 47 for V. + ECA, seed 50 for V. + SQINV + ECA, and seed 47 for UVote + ECA. Selecting by few-shot MAE naturally favors the runs that sacrifice the most many and medium-shot performance, as is visible for UVote + ECA; the seed-49 run of the same configuration, added as row 17, yields a more balanced improvement.

Table 16: IMDB-WIKI-DIR, best-run comparison. Rows 14 to 16 show our single best run per configuration, selected by the lowest few-shot MAE (seeds 47, 50, and 47); row 17 adds the seed-49 run of UVote + ECA for comparison. Baseline numbers (rows 1 to 13) from Jiang et al. (2024); improvements relative to the reference row in the last column. **Bold** marks the best and underline the second best value per column; V. abbreviates Vanilla.

#	Method	MAE				Improv. (%)				vs. #
		All	Many	Med	Few	All	Many	Med	Few	
1	Vanilla	8.06	7.23	15.12	26.33	–	–	–	–	–
2	V. + RankSim	7.72	6.93	14.48	25.38	4.22	4.15	4.23	3.61	1
3	V. + RankSim + RRT	7.55	6.83	13.47	24.72	2.20	1.44	6.98	2.60	2
4	V. + RankSim + SQINV	7.42	6.84	12.12	22.13	3.89	1.30	16.30	12.81	2
5	V. + RankSim + RRT + LDS + FDS	7.37	**6.80**	11.80	23.11	2.38	0.44	12.40	6.51	3
6	V. + RankSim + SQINV + LDS + FDS	7.69	7.13	12.30	21.43	-3.64	-4.24	-1.49	3.16	4
7	V. + RRT	7.81	7.07	14.06	25.13	3.10	2.21	7.01	4.56	1
8	V. + RRT + LDS + FDS	7.65	7.06	12.41	23.51	2.05	0.14	11.74	6.45	7
9	V. + SQINV	7.87	7.24	12.44	22.76	2.36	-0.14	17.72	13.56	1
10	V. + SQINV + LDS + FDS	7.78	7.20	12.61	22.19	1.14	0.55	-1.37	2.50	9
11	V. + BMSE	8.08	7.52	12.47	23.29	-0.25	-4.01	17.53	11.55	1
12	V. + DenseWeight	7.85	7.14	13.70	25.38	2.61	1.24	9.39	3.61	1
13	UVote	**7.36**	6.81	11.78	20.96	8.68	5.81	22.09	20.40	1
14	V. + ECA (ours)	7.44	**6.80**	12.68	23.18	7.69	5.95	16.14	11.96	1
15	V. + SQINV + ECA (ours)	7.76	7.29	11.55	21.00	1.40	-0.69	7.15	7.73	9
16	UVote + ECA (ours)	8.08	7.77	**10.06**	**20.02**	-9.78	-14.10	14.60	4.48	13
17	UVote + ECA (ours, seed 49)	7.52	7.10	10.60	20.81	-2.17	-4.26	10.02	0.72	13

I MuViS-DIR Appendix

Table 17 reports the bMASE results of the main paper together with their standard deviations, and the subsequent tables stratify each dataset by the established shot partition. Entries marked "–" correspond to shot regions that are empty under the partition thresholds, illustrating the arbitrariness of shot-based reporting discussed in subsection D.1. The shot-stratified view of Targa '13 further locates the failure case of section 4.2 in the few-shot region.

Table 17: Balanced MASE (bMASE) on the MuViS-DIR benchmark (mean ± std over ten seeds; lower is better). **Bold** marks the best and underline the second-best value per dataset. Orthogonal add-ons (left of the rule) vs. standalone methods (right); V. abbreviates Vanilla.

Dataset	Vanilla	V. + ConR	V. + Focal-L1	V. + LDS	V. + SQInv	V. + ECA	RnC	UVote
PM10	0.6677± 0.0418	0.6793± 0.0458	0.6171± 0.0160	**0.4765**± 0.0441	0.6230± 0.0381	0.6394± 0.0227	0.6663± 0.0138	0.6792± 0.0616
PM2.5	0.5705± 0.0500	0.5507± 0.0499	0.5242± 0.0213	0.5489± 0.0940	0.5457± 0.0483	0.5185± 0.0133	0.5675± 0.0368	**0.5142**± 0.0137
Heart Rate	0.1277± 0.0065	0.1925± 0.1751	0.0748± 0.0048	0.1130± 0.0141	0.1232± 0.0462	**0.0593**± 0.0025	0.0923± 0.0089	0.0782± 0.0068
Battery SoC	0.0229± 0.0006	0.0306± 0.0163	0.0280± 0.0086	**0.0224**± 0.0006	0.0262± 0.0107	0.0245± 0.0008	0.2536± 0.1592	0.0251± 0.0014
Monterey	0.2511± 0.1784	0.2695± 0.1991	0.2215± 0.1285	0.2401± 0.1649	0.2260± 0.1556	0.1009± 0.0023	0.3622± 0.3951	**0.0853**± 0.0028
Targa '13	**0.1116**± 0.0079	0.1136± 0.0043	0.1933± 0.0155	0.1212± 0.0065	0.1132± 0.0048	0.2213± 0.0061	0.3452± 0.2981	0.1136± 0.0087
Targa '14	0.1276± 0.0591	0.1319± 0.0729	0.1326± 0.0171	0.1273± 0.0477	0.1183± 0.0389	0.1218± 0.0049	0.1746± 0.0544	**0.1041**± 0.0228
Chem. Conc.	0.6910± 0.1245	0.6485± 0.0994	0.6343± 0.0236	**0.5826**± 0.0590	0.6371± 0.1305	0.6295± 0.0421	0.6349± 0.0306	0.6056± 0.0047
Tyre Temp.	0.3662± 0.0509	0.3618± 0.0392	0.3777± 0.0478	0.3534± 0.0443	0.3505± 0.0376	0.3398± 0.0372	0.3933± 0.0616	**0.3313**± 0.0435
Geometric Mean	0.2157± 0.0252	0.2338± 0.0488	0.2135± 0.0201	0.1997± 0.0257	0.2084± 0.0370	0.1870± 0.0073	0.3314± 0.0590	**0.1729**± 0.0099

Table 18: Shot-stratified MAE on PM10 (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	54.30± 1.896	38.38± 1.251	85.43± 7.023	272.1± 17.66
Vanilla + ConR	55.11± 1.772	38.58± 0.740	87.12± 6.137	282.1± 25.60
Vanilla + Focal-L1	53.49± 0.848	38.90± 1.084	81.71± 2.864	254.0± 6.167
Vanilla + LDS	90.07± 6.428	82.82± 7.322	107.6± 8.501	**179.7± 16.18**
Vanilla + SQInv	57.62± 2.833	43.64± 1.874	84.21± 6.130	251.1± 17.70
Vanilla + ECA	**52.82± 0.658**	**37.88± 1.299**	**79.42± 2.557**	264.8± 11.40
RnC	54.04± 0.649	38.29± 0.898	82.86± 1.864	275.1± 6.170
UVote	58.90± 4.454	42.86± 3.472	88.86± 9.210	282.4± 29.34

Table 19: Shot-stratified MAE on PM2.5 (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	38.39± 3.604	28.59± 2.364	84.57± 10.09	159.8± 22.90
Vanilla + ConR	37.74± 4.774	28.47± 3.244	82.34± 13.05	150.3± 21.82
Vanilla + Focal-L1	**34.68± 0.313**	**26.00± 0.310**	76.80± 1.504	139.4± 6.858
Vanilla + LDS	46.91± 2.891	37.81± 2.184	89.87± 12.73	159.4± 42.03
Vanilla + SQInv	38.17± 2.541	29.14± 1.838	80.37± 7.146	150.6± 24.48
Vanilla + ECA	35.20± 0.705	26.48± 0.662	77.42± 1.882	140.5± 3.365
RnC	35.25± 0.968	26.00± 0.680	**74.05± 2.446**	160.3± 18.82
UVote	35.45± 0.642	26.89± 0.585	78.64± 1.997	**135.0± 3.625**

Table 20: Shot-stratified MAE on Heart Rate (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	4.127± 0.193	3.935± 0.184	6.353± 0.344	5.760± 0.733
Vanilla + ConR	4.887± 1.904	4.490± 1.459	8.244± 4.422	21.85± 49.96
Vanilla + Focal-L1	2.862± 0.178	2.825± 0.171	3.298± 0.286	3.204± 0.256
Vanilla + LDS	4.827± 0.584	4.790± 0.527	5.456± 1.357	3.137± 0.192
Vanilla + SQInv	5.232± 1.934	5.135± 1.720	6.612± 4.749	3.368± 0.432
Vanilla + ECA	2.332± 0.110	2.321± 0.107	**2.457± 0.197**	2.542± 0.191
RnC	**1.894± 0.068**	**1.714± 0.062**	3.486± 0.249	8.835± 2.045
UVote	3.397± 0.279	3.393± 0.276	3.577± 0.383	**1.989± 0.227**

Table 21: Shot-stratified MAE on Battery SoC (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	**0.005± 0.000**	**0.005± 0.000**	–	–
Vanilla + ConR	0.007± 0.004	0.007± 0.004	–	–
Vanilla + Focal-L1	0.006± 0.002	0.006± 0.002	–	–
Vanilla + LDS	**0.005± 0.000**	**0.005± 0.000**	–	–
Vanilla + SQInv	0.006± 0.003	0.006± 0.003	–	–
Vanilla + ECA	**0.005± 0.000**	**0.005± 0.000**	–	–
RnC	0.056± 0.038	0.056± 0.038	–	–
UVote	0.006± 0.000	0.006± 0.000	–	–

Table 22: Shot-stratified MAE on Monterey (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	0.129± 0.054	0.111± 0.042	0.206± 0.104	0.444± 0.334
Vanilla + ConR	0.132± 0.057	0.111± 0.041	0.223± 0.120	0.503± 0.388
Vanilla + Focal-L1	0.128± 0.048	0.114± 0.040	0.188± 0.071	0.371± 0.222
Vanilla + LDS	0.160± 0.079	0.139± 0.064	0.277± 0.159	0.431± 0.304
Vanilla + SQInv	0.152± 0.084	0.135± 0.072	0.244± 0.149	0.400± 0.280
Vanilla + ECA	0.081± 0.001	0.075± 0.001	**0.115± 0.002**	0.172± 0.005
RnC	0.177± 0.156	0.134± 0.097	0.430± 0.500	0.716± 0.872
UVote	**0.079± 0.001**	**0.073± 0.001**	0.118± 0.002	**0.144± 0.006**

Table 23: Shot-stratified MAE on Targa '13 (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	0.050± 0.001	0.047± 0.001	**0.064± 0.003**	0.111± 0.008
Vanilla + ConR	**0.050± 0.001**	**0.046± 0.001**	0.064± 0.002	0.109± 0.005
Vanilla + Focal-L1	0.058± 0.002	0.051± 0.002	0.076± 0.002	0.196± 0.018
Vanilla + LDS	0.058± 0.001	0.054± 0.001	0.078± 0.004	0.115± 0.006
Vanilla + SQInv	0.052± 0.001	0.048± 0.001	0.067± 0.004	**0.108± 0.004**
Vanilla + ECA	0.062± 0.001	0.055± 0.001	0.077± 0.003	0.231± 0.009
RnC	0.143± 0.137	0.131± 0.143	0.191± 0.198	0.332± 0.345
UVote	0.051± 0.001	0.047± 0.001	0.065± 0.002	0.117± 0.010

Table 24: Shot-stratified MAE on Targa '14 (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	0.073± 0.023	0.072± 0.021	0.079± 0.040	0.084± 0.045
Vanilla + ConR	0.070± 0.023	0.068± 0.019	0.084± 0.053	0.096± 0.073
Vanilla + Focal-L1	0.072± 0.008	0.071± 0.007	0.078± 0.013	0.098± 0.020
Vanilla + LDS	0.072± 0.018	0.071± 0.017	0.077± 0.027	0.091± 0.049
Vanilla + SQInv	0.067± 0.014	0.067± 0.013	0.069± 0.026	0.087± 0.035
Vanilla + ECA	0.069± 0.002	0.069± 0.002	0.072± 0.003	0.079± 0.008
RnC	0.088± 0.017	0.085± 0.016	0.106± 0.042	0.151± 0.062
UVote	**0.062± 0.012**	**0.061± 0.012**	**0.064± 0.011**	**0.067± 0.018**

Table 25: Shot-stratified MAE on Chem. Conc. (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	0.056± 0.010	0.055± 0.010	0.124± 0.019	0.148± 0.028
Vanilla + ConR	0.052± 0.007	0.052± 0.007	0.118± 0.015	0.143± 0.036
Vanilla + Focal-L1	0.050± 0.003	0.050± 0.003	0.117± 0.007	0.134± 0.008
Vanilla + LDS	0.050± 0.004	0.050± 0.004	**0.105± 0.012**	**0.122± 0.018**
Vanilla + SQInv	0.052± 0.004	0.052± 0.004	0.116± 0.027	0.134± 0.032
Vanilla + ECA	**0.046± 0.001**	**0.046± 0.001**	0.120± 0.010	0.138± 0.012
RnC	0.055± 0.004	0.055± 0.004	0.113± 0.011	0.134± 0.011
UVote	0.048± 0.000	0.047± 0.000	0.113± 0.001	0.131± 0.002

Table 26: Shot-stratified MAE on Tyre Temp. (mean ± std over ten seeds; lower is better; best per column in bold).

Method	All	Many	Med	Few
Vanilla	3.106± 0.424	–	2.953± 0.403	4.894± 0.768
Vanilla + ConR	2.983± 0.250	–	2.805± 0.221	5.065± 0.968
Vanilla + Focal-L1	3.131± 0.325	–	2.944± 0.277	5.321± 0.994
Vanilla + LDS	3.102± 0.363	–	2.992± 0.342	4.385± 0.663
Vanilla + SQInv	3.074± 0.328	–	2.961± 0.326	4.399± 0.599
Vanilla + ECA	2.900± 0.298	–	2.757± 0.277	4.587± 0.605
RnC	3.262± 0.470	–	3.068± 0.456	5.538± 1.451
UVote	**2.875± 0.330**	–	**2.755± 0.299**	**4.289± 0.761**

J ECA Ablation

ECA rests on three design decisions: the data exposure curriculum, a learning rate schedule with an explicit decay phase (in our case WSD), and a continuous, synchronized coupling of the two. This section ablates each decision in turn, to show that the intuition behind ECA is backed by evidence and not merely by the combination happening to work. subsection J.1 extends the component ablation of the main paper: DEC and WSD each improve over Vanilla on their own, but only their combination improves every region. Since WSD alone is a strong schedule, subsection J.2 controls for it by also training LDS and SQINV with WSD. WSD improves every method, yet ECA still outperforms both in every shot region. subsection J.3 asks whether the coupling has to be continuous by tying DEC to the discrete step schedule, a regression analogue of deferred re-balancing (Cao et al., 2019). This variant improves over Vanilla but falls clearly short of ECA. Finally, subsection J.4 breaks the synchronization of the two components: the synchronized configuration is best in three of four regions, and each asynchronous variant reverts toward the behavior of one isolated component.

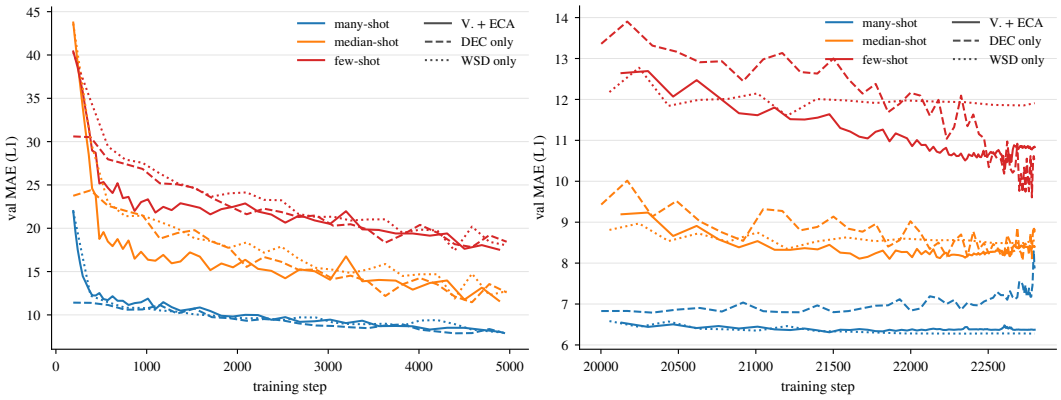
J.1 Isolated Components

This section provides additional context for the isolated-components ablation of subsection 4.3. Table 27 repeats the numbers of Table 5 and adds the relative improvement values.

Table 27: Component ablation on AgeDB-DIR (MAE, mean over ten seeds). Improvements relative to Vanilla. **Bold** marks the best and underline the second best value per column; V. abbreviates Vanilla.

Method	MAE				Improv. (%)			
	All	Many	Med	Few	All	Many	Med	Few
Vanilla	7.68 ± 0.15	6.67 ± 0.14	8.99 ± 0.18	13.52 ± 0.62	–	–	–	–
V. + DEC	7.44±0.08	6.86±0.15	8.08±0.29	11.11±0.84	3.13	-2.85	10.12	17.83
V. + WSD	7.04±0.07	**6.20**±0.09	8.22±0.30	11.67±0.50	8.33	**7.05**	8.57	13.68
V. + ECA (ours)	**6.91** ± 0.08	6.25 ± 0.13	**7.90** ± 0.29	**10.35** ± 0.22	**10.03**	6.30	**12.12**	**23.45**

Figure 7 shows the course of training for all three variants, V. + ECA (solid), DEC only (dashed), and WSD only (dotted), across ten seeds: the beginning of training, shaped by the warmup phase (a), and the end of training, shaped by the annealing phase (b). In (a), V. + ECA outperforms both isolated variants in the medium and few-shot regions over the first 2000 steps, while DEC only and WSD only are barely distinguishable from each other; the many-shot MAEs of all three variants are similar apart from the first 200 steps. The differences are most pronounced in (b). DEC only shows the typical behavior of oversampling: it strongly sacrifices many-shot performance while improving the few-shot region, and is visibly unstable. WSD only is the opposite: its curves are very stable in all three regions, but show no notable improvement over the last 1000 steps in any region except many-shot. V. + ECA shows no further many-shot improvement over the last 2000 steps, but improves the medium and, above all, the few-shot performance substantially in the final steps of training. These curves mirror the test results of Table 27.



(a) The first 5000 training steps.

(b) The last 3000 training steps.

Figure 7: Validation MAE for the many, medium, and few-shot regions of V. + ECA (solid), DEC only (dashed), and WSD only (dotted) across ten seeds on AgeDB-DIR. For readability, only the means are shown.

J.2 Isolating the Contribution of WSD

The isolated-components experiment form subsection J.1 shows that WSD is itself a substantially stronger learning rate schedule than the step schedule established for AgeDB-DIR. This raises the question of how much of the improvement of ECA over existing DIR methods originates from WSD rather than from DEC. We therefore retrain two static rebalancing methods, LDS and SQINV, with the same WSD schedule used by ECA. The original configurations use the established step schedule, which reduces the learning rate by a factor of 0.1 at epochs 60 and 80. All configurations are evaluated across the same ten seeds (Table 28).

Table 28: Isolating WSD on AgeDB-DIR (MAE, mean ± std over ten seeds). "Step" denotes the established $[60, 80]$ step schedule, "Const" a constant learning rate. **Bold** marks the best and underline the second best value per column; V. abbreviates Vanilla.

#	Method	All	Many	Med	Few
1	Vanilla + Step (Orig.)	$7.68_{\pm 0.15}$	$6.67_{\pm 0.14}$	$8.99_{\pm 0.18}$	$13.52_{\pm 0.62}$
2	Vanilla + WSD	$\underline{7.04}_{\pm 0.06}$	$\mathbf{6.20}_{\pm 0.09}$	$8.22_{\pm 0.28}$	$11.67_{\pm 0.47}$
3	V. + LDS + Step (Orig.)	$7.69_{\pm 0.19}$	$7.03_{\pm 0.20}$	$8.64_{\pm 0.43}$	$11.32_{\pm 0.33}$
4	V. + LDS + WSD	$7.27_{\pm 0.15}$	$6.68_{\pm 0.16}$	$8.12_{\pm 0.20}$	$10.55_{\pm 0.47}$
5	V. + SQINV + Step (Orig.)	$7.69_{\pm 0.17}$	$7.02_{\pm 0.16}$	$8.65_{\pm 0.30}$	$11.42_{\pm 0.62}$
6	V. + SQINV + WSD	$7.26_{\pm 0.12}$	$6.67_{\pm 0.09}$	$8.09_{\pm 0.24}$	$\underline{10.48}_{\pm 0.43}$
7	V. + DEC + Const	$7.44_{\pm 0.07}$	$6.86_{\pm 0.15}$	$\underline{8.08}_{\pm 0.27}$	$11.11_{\pm 0.80}$
8	V. + ECA (DEC + WSD)	$\mathbf{6.91}_{\pm 0.08}$	$\underline{6.25}_{\pm 0.13}$	$\mathbf{7.90}_{\pm 0.28}$	$\mathbf{10.35}_{\pm 0.21}$

Table 28 confirms that WSD is a stronger learning rate schedule independently of ECA. It improves Vanilla from 7.68 to 7.04 overall, LDS from 7.69 to 7.27, and SQINV from 7.69 to 7.26. Comparisons against the original configurations therefore mix two effects: the rebalancing method and the learning rate schedule.

Controlling for the schedule, however, does not remove the advantage of ECA. Relative to V. + WSD, ECA further reduces the overall MAE from 7.04 to 6.91, the medium-shot MAE from 8.22 to 7.90, and the few-shot MAE from 11.67 to 10.35. WSD alone retains the lowest many-shot error at 6.20, compared with 6.25 for ECA. This is consistent with the interpolation result of subsection 4.4: the final part of the annealing trajectory exchanges a small amount of many-shot accuracy for a larger gain in progressively sparser regions.

LDS and SQINV show a related effect. Under the step schedule, both obtain their tail improvements at the expense of the many-shot region. With WSD, their many-shot MAEs drop to 6.68 and 6.67,

back to the level of the original Vanilla model, while their medium and few-shot errors improve further. Against the appropriate baseline, V. + WSD, both methods still lose about 0.47 MAE in the many-shot region, however, and both end up worse overall than WSD alone (7.27 and 7.26 versus 7.04). ECA is the only addition that improves on WSD alone, and it remains the strongest method overall as well as in the medium and few-shot regions. The relevant difference is therefore not only whether rebalancing coincides with a decaying learning rate, but how the influence of the target regions develops during this phase: LDS and SQINV apply a fixed weighting throughout training, whereas DEC changes the effective data composition continuously together with the remaining step size.

This distinction does not require the mechanism to be unique to sampling. A weighting method whose strength changes continuously with the learning rate could, in principle, produce a similar trajectory. Indeed, the SQINV + WSD control of Appendix L shows that static weighting can develop an ECA-like perturbation signature when its effective gradient composition becomes increasingly tail-directed late in training, albeit with smaller gains. ECA constructs this regime explicitly through data exposure, while leaving the optimization objective unchanged.

Bounded rebalancing. DEC also differs from inverse-frequency weighting in how it increases the influence of sparse target regions. Both approach equal mass per bin, but inverse-frequency weighting gets there by amplifying individual tail samples, on AgeDB-DIR by a factor of more than 350 relative to head samples. DEC never duplicates or upweights a sample. It only removes surplus samples above the current ceiling, so every retained sample carries the same weight at every step. At the endpoint of the schedule, with $c_{\min} = 1$, every occupied bin contributes exactly one sample per epoch, and the effective distribution is uniform over the discretized supported target range. For AgeDB-DIR, each of the B occupied one-year bins then contributes a share of $1/B$.

Samples from dense bins rotate through the single retained position between epochs, so they are not permanently discarded. DEC thus interpolates between the empirical training distribution and uniform support over the target range, which is exactly the shift from $p_{\text{train}}(y)$ to $p_{\text{eval}}(y)$ that motivates DIR.

J.3 Discrete versus Continuous Coupling

The previous experiment establishes that a decaying learning rate benefits rebalancing methods more generally. We next ask whether the continuous coupling of ECA is necessary, or whether a small number of deferred distribution changes is sufficient.

The step schedule established for AgeDB-DIR provides a natural discrete control, reducing the learning rate by a factor of 0.1 at epochs 60 and 80. We couple DEC to this schedule, so that every learning rate drop is accompanied by a corresponding drop of the histogram ceiling. This yields a discrete analogue of ECA that is closely related to deferred re-balancing (Cao et al., 2019): the model first trains on the natural distribution and then switches abruptly to an increased relative exposure of sparse target regions.

Table 29: Discrete coupling of DEC to the established step schedule on AgeDB-DIR (MAE, mean $\pm$ std over ten seeds). **Bold** marks the best and underline the second best value per column; V. abbreviates Vanilla.

#	Method	All	Many	Med	Few
1	Vanilla + Step (Orig.)	$7.68_{\pm 0.15}$	$\underline{6.67}_{\pm 0.14}$	$8.99_{\pm 0.18}$	$13.52_{\pm 0.62}$
2	V. + DEC + Const	$7.44_{\pm 0.07}$	$6.86_{\pm 0.15}$	$8.08_{\pm 0.27}$	$\underline{11.11}_{\pm 0.80}$
3	V. + DEC + Step	$\underline{7.40}_{\pm 0.11}$	$6.75_{\pm 0.13}$	$\underline{8.00}_{\pm 0.23}$	$11.86_{\pm 0.38}$
4	V. + ECA (DEC + WSD)	$\mathbf{6.91}_{\pm 0.08}$	$\mathbf{6.25}_{\pm 0.13}$	$\mathbf{7.90}_{\pm 0.28}$	$\mathbf{10.35}_{\pm 0.21}$

The discrete coupling improves over Vanilla in the overall, medium, and few-shot regions, but its behavior differs markedly from ECA. The medium-shot MAE decreases from 8.99 to 8.00 and the few-shot MAE from 13.52 to 11.86, while the many-shot MAE slightly increases from 6.67 to 6.75. The variant remains behind ECA in all four regions, its few-shot gain is smaller than that of DEC

with a constant learning rate (11.86 versus 11.11), and its overall MAE of 7.40 is also worse than that of WSD alone (7.04, Table 28).

The same behavior is visible in the validation curves (Figure 8) and in checkpoint selection. Across all ten seeds, the best validation checkpoint occurs between epochs 63 and 70, shortly after the first coupled transition at epoch 60 and before the second one at epoch 80; the selected epochs are 70, 68, 68, 64, 67, 70, 69, 63, 70, and 70 for seeds 42 to 51. After the first transition, the clipped distribution reduces the number of batches per epoch from 190 to 39 at batch size 64. The latest selected checkpoint is therefore only 390 optimizer steps after the first distribution change, a little more than two epochs under the original distribution. After the second transition, the validation MAEs of all three regions nearly converge to a common value of about 8.3.

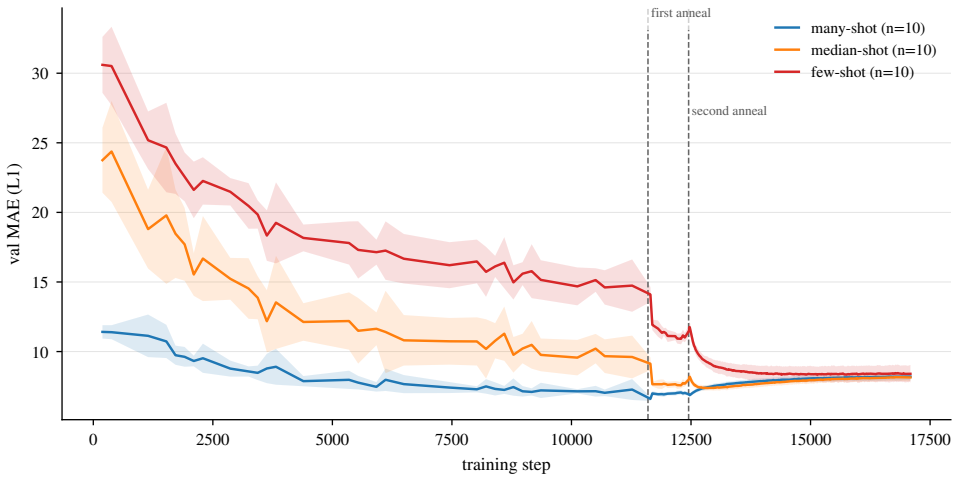


Figure 8: Validation MAE for the many, medium, and few-shot regions of V. + DEC + Step across ten seeds on AgeDB-DIR. The vertical dashed lines mark the two learning rate drops at epochs 60 and 80.

These results suggest that an abrupt change of exposure does not provide the same separation between consolidation and tail correction as the continuous schedule. The first transition shifts the data composition at once, and the best model is reached within a few hundred steps of it, with gains concentrated in the medium region, a smaller few-shot gain than either continuous variant, and a slight many-shot cost. ECA instead spreads the same distributional change across the entire decay phase, so that increasingly balanced exposure coincides with increasingly small updates. Deferred re-balancing through the established discrete schedule is therefore not sufficient to reproduce the result of ECA.

J.4 Breaking Synchronization

The last design decision is that DEC and WSD decay in sync. To test it, we desynchronize the two components on AgeDB-DIR by giving their decay phases different lengths. ECA is not intended to be used this way and the variants serve only as an ablation. In the standard configuration for AgeDB-DIR, both components decay over the final 20% of training (decay fraction 0.2). We compare it with two asymmetric variants. In the first, DEC decays over the final 50% of training while WSD decays only over the final 10%; in the second, the roles are swapped. The subscripts in Table 30 give the decay fraction of each component.

Table 30: Asynchronous decay phases of DEC and WSD on AgeDB-DIR (MAE, mean ± std over ten seeds). Subscripts give the decay fraction of each component; the synchronized configuration (row 4) is ECA. **Bold** marks the best and underline the second best value per column; V. abbreviates Vanilla.

#	Method	All	Many	Med	Few
1	Vanilla + Step (Orig.)	$7.68_{\pm 0.15}$	$\underline{6.67}_{\pm 0.14}$	$8.99_{\pm 0.18}$	$13.52_{\pm 0.62}$
2	V. + $\text{DEC}_{0.5}$ + $\text{WSD}_{0.1}$	$\underline{7.33}_{\pm 0.08}$	$6.89_{\pm 0.12}$	$\mathbf{7.71}_{\pm 0.24}$	$\underline{10.44}_{\pm 0.54}$
3	V. + $\text{DEC}_{0.1}$ + $\text{WSD}_{0.5}$	$7.39_{\pm 0.07}$	$6.69_{\pm 0.13}$	$8.18_{\pm 0.22}$	$11.93_{\pm 0.44}$
4	V. + $\text{DEC}_{0.2}$ + $\text{WSD}_{0.2}$ (ECA)	$\mathbf{6.91}_{\pm 0.08}$	$\mathbf{6.25}_{\pm 0.13}$	$\underline{7.90}_{\pm 0.28}$	$\mathbf{10.35}_{\pm 0.21}$

The synchronized configuration is best in three of four regions (Table 30). More revealing is how the asynchronous variants fail: each reverts toward one of the isolated components of subsection J.1. With $\text{DEC}_{0.5}$ + $\text{WSD}_{0.1}$, rebalancing starts while the learning rate is still at its peak. Like DEC alone, this variant gains in the sparse regions, reaching the best medium-shot MAE in the table (7.71) and the second best few-shot MAE (10.44), but it pays in the many-shot region, where its error of 6.89 is almost identical to that of DEC alone (6.86) and worse than Vanilla. With $\text{DEC}_{0.1}$ + $\text{WSD}_{0.5}$, rebalancing starts only after the learning rate has already largely decayed. Like WSD alone, this variant gains little in the sparse regions (8.18 and 11.93, close to the 8.22 and 11.67 of WSD alone), without matching its many-shot error (6.69 versus 6.20).

Figure 9 shows the same pattern over the course of training. Compared with Figure 7, $\text{DEC}_{0.5}$ + $\text{WSD}_{0.1}$ reproduces the curves of DEC alone, with improving medium and few-shot validation MAE at the expense of the many-shot region, while $\text{DEC}_{0.1}$ + $\text{WSD}_{0.5}$ reproduces the stable but largely flat tail curves of WSD alone.

Only the synchronized configuration combines both benefits: the head is consolidated at full step size, and the tail is corrected while the step size vanishes. Rebalancing too early costs the head; rebalancing too late leaves too few and too small updates to write the tail correction (following our intuition of subsection 3.2).

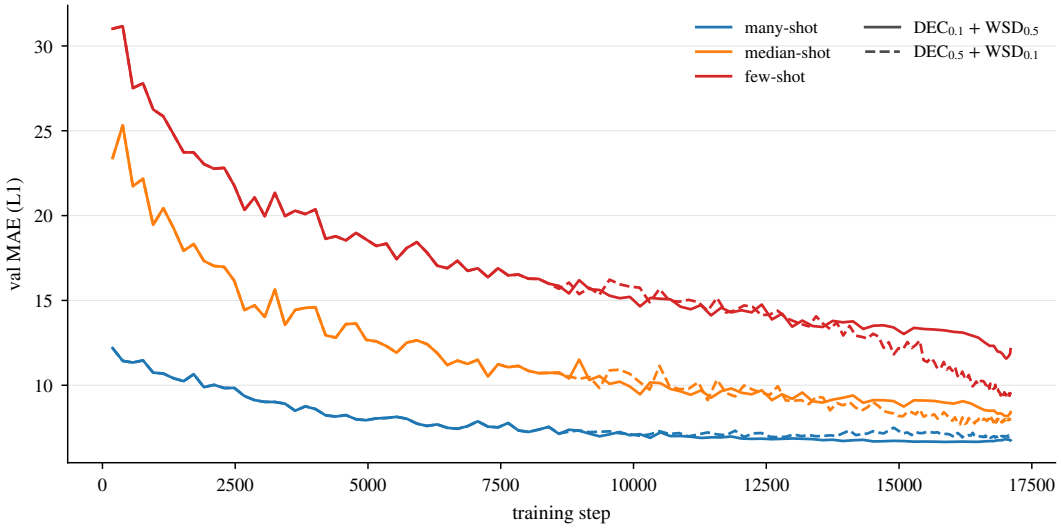


Figure 9: Validation MAE for the many, medium, and few-shot regions of the two asynchronous configurations, $\text{DEC}_{0.5}$ + $\text{WSD}_{0.1}$ and $\text{DEC}_{0.1}$ + $\text{WSD}_{0.5}$, across ten seeds on AgeDB-DIR. For readability, only the means are shown.

K Quantization

In parallel to the noise probe of subsection 4.4, we test the same prediction through deterministic rounding instead of stochastic perturbation: we apply post-training uniform quantization to the

model weights at decreasing bit widths and monitor test MAE per shot region, both for the final annealed model and for the pre-annealing checkpoint taken at the start of the decay phase (Figure 10). For the annealed model, many-shot performance is unchanged between 32 and 4 bits, while below 6 bits the medium region degrades slightly and the few-shot region degrades substantially; at 3 bits the model collapses across all regions (excluded for readability). The pre-annealing checkpoint behaves differently: it shows no significant degradation at 5 bits in any region and smaller deviations at 4 bits in the medium and few-shot regions than the annealed model.These results support our hypothesis: the annealing gains in the sparse regions are the first casualty of reduced weight precision, while the majority mapping, learned during the stable phase, resides in structure that survives coarse quantization. The main limitation of this probe is its coarse resolution: bit widths are discrete, differences only become visible between 5 and 4 bits, and the model collapses at 3 bits. We therefore use the noise probe, whose perturbation strength can be varied continuously, as the primary test in the main paper.

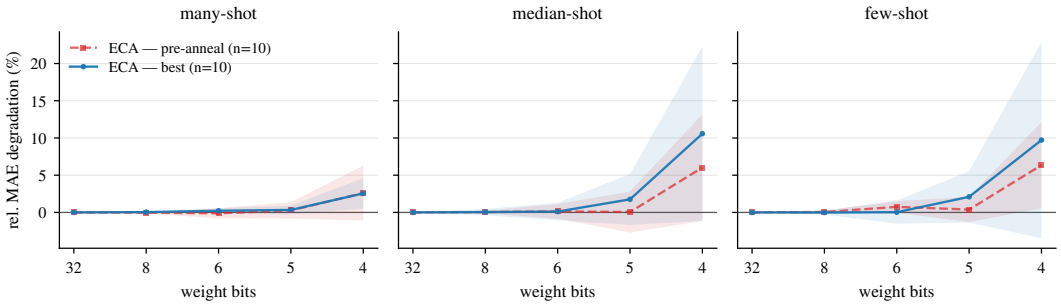


Figure 10: Quantization probe on AgeDB-DIR: test MAE for the many (left), medium (center), and few-shot (right) regions under post-training uniform quantization at decreasing bit widths, for the pre-annealing checkpoint and the annealed model. The dashed line marks the unquantized reference.

L Noise Probe Controls with Static Rebalancing

The noise probe of subsection 4.4 admits an alternative explanation: predictions in sparse target regions might be generically more sensitive to weight perturbations, regardless of when the corresponding structure was written. To separate the two explanations, we repeat the probe on models trained with LDS and SQInv, two static rebalancing methods that acquire their tail gains from the first step of training onward, each once with a constant learning rate and once with the WSD schedule used by ECA. All four configurations are repeated across four seeds; the protocol is otherwise identical to subsection 4.4.

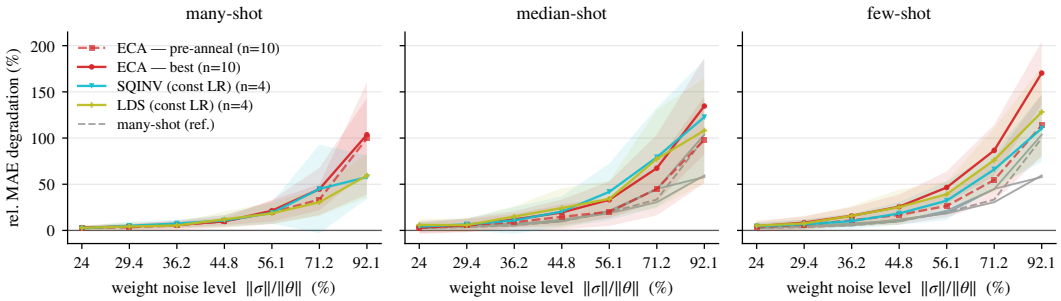


Figure 11: Noise probe for LDS and SQInv trained with a constant learning rate on AgeDB-DIR: relative test MAE degradation for the many, medium, and few-shot regions under increasing Gaussian weight noise (mean over four seeds).

Figure 11 shows the results for a constant learning rate. Without WSD, both methods display a moderate increase in degradation from the many to the medium-shot region but no significant further

increase from medium to few; for SQInv, the few-shot region is even less affected than the medium-shot region. Sparse regions are thus mildly more perturbation-sensitive in general, but the signature of ECA, a degradation that escalates monotonically with region rarity and peaks in the few-shot region, does not appear. With WSD, the two methods diverge (Figure 12): LDS degrades more strongly in the many-shot region than in the medium-shot region and more strongly again in the few-shot region, a V-shaped pattern rather than a monotone escalation, whereas SQInv reproduces the ECA-like monotone escalation, although with a smaller tail gain to lose in the first place.

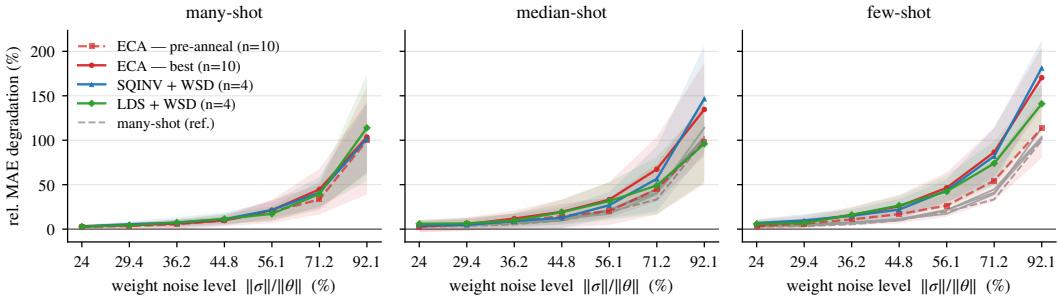


Figure 12: Noise probe for LDS and SQInv trained with the WSD schedule on AgeDB-DIR: relative test MAE degradation for the many, medium, and few-shot regions under increasing Gaussian weight noise (mean over four seeds).

The behavior of SQInv under WSD is consistent with the mechanism proposed in subsection 3.2. As training converges, residuals in the many-shot region shrink while inverse-frequency weighting keeps amplifying the sparse regions, so the effective gradient composition drifts toward the tail late in training. Combined with a decaying step size, SQInv therefore writes tail structure under conditions similar to ECA, small tail-directed updates at the end of training, and inherits the same noise sensitivity. Two conclusions follow. First, the escalating fragility tracks when and how tail structure is written, not merely where predictions are sparse, which is exactly the prediction of the weight precision view. Second, SQInv arrives at this regime with smaller gains, whereas ECA constructs it deliberately; we read this as evidence for designing the late-training exposure explicitly rather than relying on loss weighting to approximate it.

M Statistical Comparison Across Datasets

The standard deviations in the main tables describe the variation of each method across seeds on a single dataset. They do not show whether a method is consistently better across datasets. To answer this question, we compare the methods over all datasets with the procedure recommended by Demšar (2006), implemented in the `autorank` package (Herbold, 2020).

Procedure. Each dataset contributes one value per method, namely the mean over ten seeds. For every dataset, the methods are ranked from best (rank 1) to worst, with ties receiving the average rank. A Friedman test then checks whether the mean ranks differ across methods at all. If it rejects the null hypothesis at $\alpha = 0.05$, the Nemenyi post-hoc test identifies which pairs of methods differ: two methods are significantly different if their mean ranks differ by more than the critical difference (CD). We use these rank-based tests rather than a parametric alternative such as a repeated-measures ANOVA, because the error scales differ by orders of magnitude between datasets (for example, bMASE ranges from 0.02 on Battery SoC to 0.69 on Chem. Conc.), so raw values are not comparable across datasets. Ranks remove this dependence on scale.

Methods and datasets. The Friedman test requires a complete comparison: every method must be evaluated on every dataset. We therefore restrict the analysis to the eight methods that are available with ten-seed results on all MuViS-DIR datasets: Vanilla, V. + ConR, V. + Focal-L1, V. + LDS, V. + SQInv, V. + ECA, RnC, and UVote. The same eight methods are also available on AgeDB-DIR. IMDB-WIKI-DIR is excluded, because several of these methods (ConR, Focal-L1, RnC, and

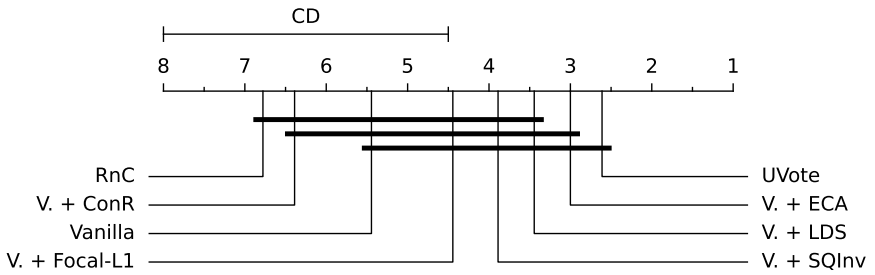


Figure 13: Critical difference diagram for bMASE on the nine MuViS-DIR datasets (Friedman $p = 0.0006$, Nemenyi CD $= 3.50$ at $\alpha = 0.05$). Methods are placed at their mean rank; lower is better. Methods connected by a bar do not differ significantly. V. abbreviates Vanilla.

Table 31: Friedman and Nemenyi results per shot region, based on the shot-stratified MAE of AgeDB-DIR and MuViS-DIR (eight methods, $\alpha = 0.05$). N is the number of datasets with a non-empty shot region.

Region	N	Friedman p	CD	Rank of ECA	Position	Best method (rank)
Overall	10	0.015	3.32	2.50	1/8	V. + ECA (2.50)
Many	9	0.033	3.50	2.67	1/8	V. + ECA (2.67)
Medium	9	0.061	3.50	3.00	2/8	UVote (2.83)
Few	9	0.002	3.50	3.67	3/8	UVote (2.33)

LDS without further components) are not reported there, and its baselines are single runs rather than ten-seed means (Appendix H).

We run two analyses. The first uses bMASE, the primary metric of MuViS-DIR, on its nine datasets. The second is carried out separately for each shot region and uses the shot-stratified MAE of AgeDB-DIR (Table 14) and of the nine MuViS-DIR datasets (Appendix I). Shot regions that are empty for a dataset are left out of the corresponding test: Battery SoC has no medium and few-shot samples, and Tyre Temp. has no many-shot samples. The overall comparison therefore covers ten datasets, and each shot region nine.

Results on bMASE. The Friedman test rejects equal mean ranks ($p = 0.0006$, CD $= 3.50$). Figure 13 shows the result. UVote attains the best mean rank (2.61), followed by ECA (3.00), which is the best orthogonal method. The ranks of the remaining orthogonal methods range from 3.44 (LDS) to 6.39 (ConR). With nine datasets, only the difference between ECA and RnC exceeds the critical difference.

Results per shot region. Table 31 and Figure 14 summarize the per-region analysis. ECA attains the best mean rank of all eight methods in the overall and in the many-shot comparison. In the many-shot region, it is significantly better than LDS. In the medium-shot region, ECA ranks second behind UVote, but the Friedman test is not significant ($p = 0.061$), so no pairwise conclusions can be drawn for this region. In the few-shot region, ECA ranks third behind UVote and LDS, without a significant difference to either.

Interpretation. The analysis supports three statements. First, the advantage of ECA is consistent across datasets rather than driven by a few favorable benchmarks: on bMASE, ECA is the best orthogonal method and is only outranked by the non-orthogonal UVote. Second, ECA does not pay the majority tax of static rebalancing: it holds the best many-shot rank of all eight methods and is significantly better than LDS in this region, which matches the per-benchmark results of section 4. Third, the few-shot ranking reflects the expected trade-off: methods that rebalance throughout training, such as LDS, or that allocate dedicated capacity, such as UVote, rank higher in the tail, at a cost in the head.

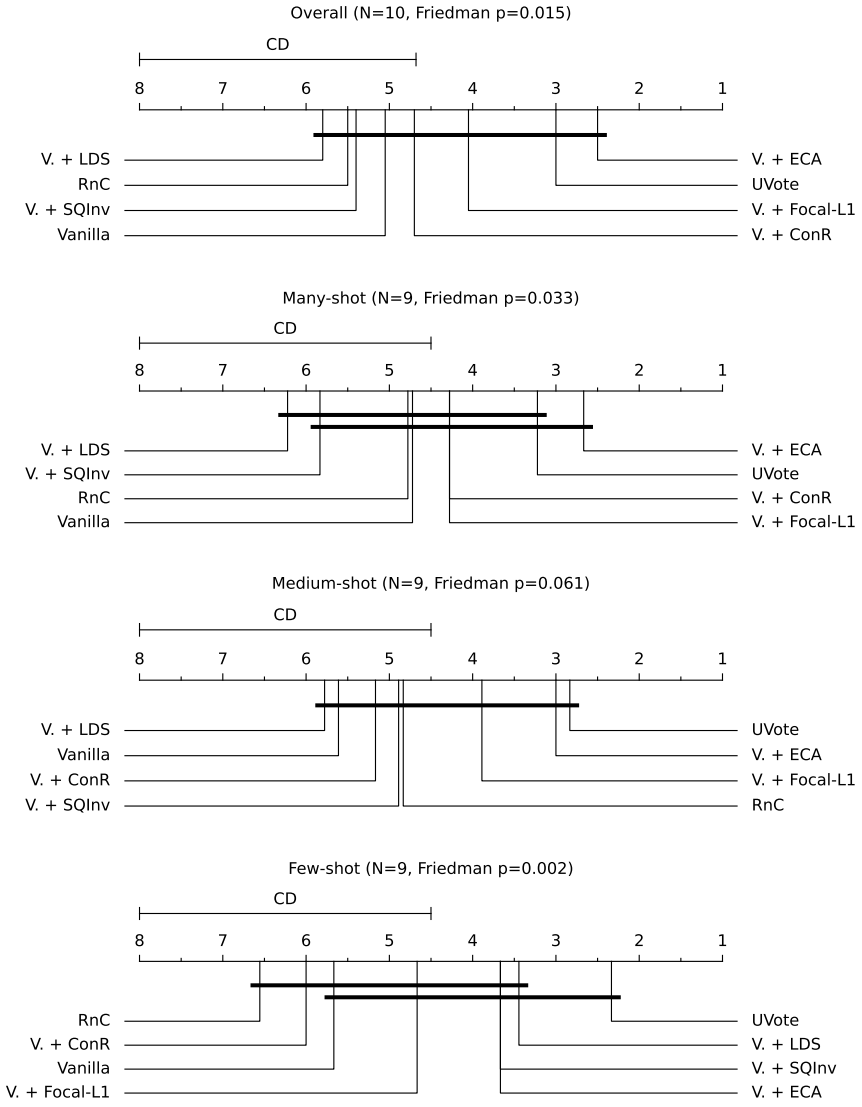


Figure 14: Critical difference diagrams per shot region (top to bottom: overall, many-shot, medium-shot, few-shot), based on the shot-stratified MAE of AgeDB-DIR and the nine MuViS-DIR datasets. Methods are placed at their mean rank; lower is better. Methods connected by a bar do not differ significantly. The medium-shot Friedman test is not significant ($p = 0.061$). V. abbreviates Vanilla.

The analysis also has a clear limit. With nine to ten datasets, the Nemenyi test can only detect large rank differences, so most pairwise differences, including those between ECA and Vanilla, LDS, SQInv, and Focal-L1 on bMASE, remain below the critical difference. The results therefore establish a consistent ranking, not a significant superiority over every competing method.