AdaPrior: Bayesian-Inspired Adaptive Prior Correction for Long-Tailed Continual Learning

Supplementary Material

Method	Setting	Prior source	Dynamic prior	Train+Test	Theory
Menon *et al.* [32]	LT	counts	×	post-hoc only	×
Huang *et al.* [19]	balanced online CIL	counts (window)	limited	train only	×
AdaPrior	**LTCIL**	predictions	✓	✓	✓

Table 7. Conceptual comparison with prior-correction methods.

7. Conceptual comparison with prior correction methods.

Table 7 contrasts AdaPrior with representative prior-correction approaches. Static logit-adjustment methods for long-tailed recognition [32] use dataset-level class counts and apply a fixed post-hoc correction. More recent online variants [19] allow limited adaptation through sliding-window label frequencies, but remain count-based and are typically applied only during training. In contrast, AdaPrior is designed for LTCIL, where the effective class prior is induced by the evolving model itself and may drift over time due to replay imbalance and representation updates.

Rather than relying on dataset counts, AdaPrior estimates the prior directly from model predictions and tracks it online via an EMA. This enables continuous adaptation to non-stationary posterior drift, and the correction can be applied during both training and inference. Unlike prior count-based formulations, AdaPrior also admits a theoretical interpretation. Overall, the key distinction is that AdaPrior targets *model-induced prior drift*, a failure mode not addressed by static or frequency-based correction methods.

8. Theoretical Analysis

In this section we present rigorous justification of the proposed AdaPrior approach. We first provide full proofs for Theorem 1 (post-hoc correction) and Theorem 2 (loss-based correction) introduced in Sec. 3.3. We then establish convergence properties of the EMA-based prior estimator and derive an excess risk bound for AdaPrior.

8.1. Proof of Theorem 1 (Post-hoc AdaPrior)

[Post-hoc AdaPrior] Let $P_m(y|x)$ denote the posterior probabilities of a trained model after task τ, and $P_m^\tau(y)$ its model-induced prior. Then the bias-corrected posterior with correction coefficient $\alpha \in [0,1]$ is

$$P_t^{(\alpha)}(y|x) = P_m(y|x) \cdot \left(\frac{P_t(y)}{P_m^\tau(y)}\right)^\alpha \cdot \frac{P(x)}{P_t(x)}.$$

Equivalently, in logit space,

$$z^\tau(x,y) = \bar{z}^\tau(x,y) + \alpha(\log P_t(y) - \log P_m^\tau(y)).$$

Under class-balanced evaluation ($P_t(y)$ uniform), this reduces to $z_t^{(\alpha)}(x,y) = z(x,y) - \alpha \log P_m^\tau(y)$.

Proof. From Bayes' theorem under the training distribution $P(x,y)$ we have

$$P_m(y|x) = \frac{P_m(x|y)\, P_m(y)}{P(x)}. \tag{10}$$

Under the target distribution $P_t(x,y)$,

$$P_t(y|x) = \frac{P_t(x|y)\, P_t(y)}{P_t(x)}. \tag{11}$$

Assuming the model captures class conditionals well ($P_m(x|y) \approx P_t(x|y)$), dividing (11) by (10) yields

$$P_t(y|x) = P_m(y|x) \frac{P_t(y)}{P_m(y)} \frac{P(x)}{P_t(x)}. \tag{12}$$

Tempered correction. Because $P_m(y)$ is estimated from finite, evolving data, we temper the prior ratio by $\alpha \in [0,1]$ to control correction strength:

$$P_t^{(\alpha)}(y|x) \propto P_m(y|x) \left(\frac{P_t(y)}{P_m(y)}\right)^\alpha \frac{P(x)}{P_t(x)}.$$

Normalization over y gives Eq. 5. In logit space, this corresponds to the additive form in Eq. 6.

Marginal correction. Integrating both sides of (12) over $P_t(x)$ gives

$$\int P_t(y|x) P_t(x)\, dx = \frac{P_t(y)}{P_m(y)} \int P_m(y|x) P(x)\, dx. \tag{13}$$

By definition of the model prior (Eq. 4), $\int P_m(y|x)P(x)\,dx \approx P_m(y)$, so the marginal of the corrected model matches $P_t(y)$. When the correction is tempered by α, the marginal smoothly interpolates between $P_m(y)$ (for α=0) and $P_t(y)$ (for α=1), providing controlled adaptation.

Fixed-point property. At equilibrium, when $P_m(y) = P_t(y)$, the correction term equals 1 for any α, implying $P_t^{(\alpha)}(y|x) = P_m(y|x)$. Thus, the unbiased alignment $P_m(y) = P_t(y)$ remains the fixed point of the transformation.

Balanced-test simplification. In typical evaluation setups with uniform class priors, $P_t(y)$ cancels out, yielding

$$P_t^{(\alpha)}(y|x) \propto P_m(y|x)\, P_m(y)^{-\alpha},$$

equivalent to the practical update $z_t^{(\alpha)}(x,y) = z(x,y) - \alpha \log P_m^{\tau}(y)$ used in our experiments.

Interpretation. The coefficient α controls the rate, not the target, of alignment. Larger α enforces stronger Bayes correction, while smaller α moderates updates when $P_m^{\tau}(y)$ is noisy or drifting. This tempering stabilizes adaptation across sequential tasks without altering the unbiased fixed point where $P_m(y) = P_t(y)$. □

We summarize the steps involved in post-hoc AdaPrior as shown in Algorithm 1.

8.2. Proof of Theorem 2 (AdaPrior Loss)

[AdaPrior Loss] If $\mathcal{L}_{\text{AP}}$ is optimized with accurate $P_m(y)$, then the resulting posterior is aligned with $P_t(y|x)$ up to a scale factor and has marginal bias $P_t(y)$.

Proof. From Theorem 1 we know

$$P_t(y|x) = P_m(y|x) \cdot \frac{P_t(y)}{P_m(y)} \cdot \frac{P(x)}{P_t(x)}.$$

Let $P_{\bar{m}}(y|x)$ denote the unadjusted model outputs before correction. By definition of logit adjustment,

$$P_m(y|x) = P_{\bar{m}}(y|x) \cdot \frac{P_m(y)}{P_t(y)}.$$

Substituting back:

$$P_t(y|x) = P_{\bar{m}}(y|x) \cdot \frac{P(x)}{P_t(x)}.$$

Thus, $P_{\bar{m}}(y|x)$ differs from $P_t(y|x)$ only by a normalizing scale factor $\frac{P(x)}{P_t(x)}$, independent of y. Therefore, its marginal bias is

$$\int P_{\bar{m}}(y|x) P_t(x) dx = \frac{P_t(y)}{P_m(y)} \int P_m(y|x) P(x) dx = P_t(y).$$

Optimizing $\mathcal{L}_{\text{PA}}$ enforces this alignment, completing the proof. □

Algorithm 2 summarizes the steps involved in AdaPrior Loss approach.

8.3. Excess Risk Bound

Finally, we analyze the generalization impact of correcting priors.

Theorem 4 (Excess Risk Bound)**.** *Let $\hat{P}_m(y)$ be the estimated prior and $P_m(y)$ the true model prior. Let $\mathcal{R}(f)$ denote the expected risk under balanced test distribution. Then the excess risk of AdaPrior satisfies*

$$\mathcal{R}(\hat{f}) - \mathcal{R}(f^*) \leq C \cdot \|\hat{P}_m - P_m\|_1,$$

for some constant C depending on the loss Lipschitz constant.

Proof. Let f^* denote the Bayes-optimal classifier under perfect prior correction. When using $\hat{P}_m(y)$ instead of $P_m(y)$, the logit adjustment deviates by

$$\Delta(y) = \log \frac{P_t(y)}{\hat{P}_m(y)} - \log \frac{P_t(y)}{P_m(y)}.$$

By Lipschitz continuity of the loss in logits, the difference in risk is bounded by $C\|\Delta\|_1$. Using a Taylor expansion around $P_m(y)$ shows $\|\Delta\|_1 = O(\|\hat{P}_m - P_m\|_1)$, yielding the stated bound. □

9. EMA Convergence Analysis

This section complements Theorem 3 from the main paper by providing a full derivation with the same notation as Eq. (9).

Setup. Recall the EMA update from Eq. (9):

$$P_m^{i+1}(y) \;=\; (1-\gamma)\, P_m^{i}(y) + \gamma\, \frac{1}{|\mathcal{B}^i|} \sum_{x \in \mathcal{B}^i} P_m(y \mid x), \qquad \gamma \in (0,1). \tag{14}$$

Let $\mathcal{F}_i$ be the filtration up to iteration i and define the conditional batch mean

$$M^{\,i}(y) \;\triangleq\; \mathbb{E}\left[\frac{1}{|\mathcal{B}^i|} \sum_{x \in \mathcal{B}^i} P_m(y \mid x) \;\middle|\; \mathcal{F}_{i-1} \right]. \tag{15}$$

We make the standard assumptions used in stochastic approximation: (i) bounded variance $\mathrm{Var}\big[\frac{1}{|\mathcal{B}^i|} \sum_{x \in \mathcal{B}^i} P_m(y \mid x)\big] \leq \sigma^2$, and (ii) either stationarity $M^{\,i} \equiv M$ or bounded drift $\|M^{\,i+1} - M^{\,i}\| \leq \bar{d}$.

Theorem 5 (Restatement of Theorem 3)**.** *Under (14)–(15) with the assumptions above:*
(a) Stationary case. *If $M^{\,i} \equiv M$ (no drift), then $P_m^{\,i} \to M$ almost surely.*
(b) Drifting case. *If $\|M^{\,i+1} - M^{\,i}\| \leq \bar{d}$ for all i, then*

$$\limsup_{i \to \infty} \mathbb{E}\big[\, \|P_m^{\,i} - M^{\,i}\| \big] \;=\; \mathcal{O}(\gamma) \;+\; \mathcal{O}\left(\frac{\bar{d}}{\gamma}\right). \tag{16}$$

Proof sketch (complete details). Let $e_i = P_m^i - M^i$ and write the batch-average as $M^i + N_i$, where $\mathbb{E}[N_i \mid \mathcal{F}_{i-1}] = 0$ and $\mathbb{E}\|N_i\|^2 \le C\sigma^2$. From (14):

$$e_{i+1} = (1-\gamma)\, e_i + \gamma N_i - (1-\gamma)\Delta_i, \quad \Delta_i := M^{i+1} - M^i.$$

Define $V_i = \|e_i\|^2$. Using $\mathbb{E}[\langle e_i, N_i\rangle \mid \mathcal{F}_{i-1}] = 0$ and $\|a+b+c\|^2 \le (1+\eta)\|a\|^2 + (1+\frac{1}{\eta})\|b\|^2 + C\|c\|^2$, one obtains

$$\mathbb{E}[V_{i+1} \mid \mathcal{F}_{i-1}] \;\le\; (1-c\gamma)\, V_i + C_1\gamma^2 + C_2\, \frac{\|\Delta_i\|^2}{\gamma}.$$

When $\Delta_i = 0$, Robbins–Siegmund yields $V_i \to 0$ almost surely. For bounded drift, taking expectations and unrolling the recursion with constant γ gives $\mathbb{E}V_{i+1} \le (1-c\gamma)\mathbb{E}V_i + C_1\gamma^2 + C_2\bar{d}^2/\gamma$, implying the steady-state bound in (16). □

Variable step sizes. For diminishing steps $\lambda_i \in (0,1)$ with $\sum_i \lambda_i = \infty$ and $\sum_i \lambda_i^2 < \infty$, the same proof applies with γ replaced by λ_i, yielding convergence in the stationary case and a tracking bound with λ_i in place of γ.

Practical guidance for γ. Eq. (16) shows the classic trade-off: larger γ adapts faster but increases noise ($\mathcal{O}(\gamma)$); smaller γ reduces noise but tracks drift more slowly ($\mathcal{O}(\bar{d}/\gamma)$). In practice we fix $\gamma \in [0.03, 0.07]$ across datasets (chosen on CIFAR100-LT once), consistent with our stability/accuracy trends.

Initialization. The result is agnostic to $P_m^0(y)$; initializing with class frequencies $P(y)$ or uniform is valid and only affects the short transient.

10. EMA Tracking Sanity Check

We validate Theorem 3 and Theorem 5 in a controlled toy setup: a categorical prior over $K = 3$ classes undergoes piecewise drifts of small magnitude every 200 iterations. At each iteration, we sample a batch of labels from the current prior and convert them to soft posteriors to emulate $P_m(y \mid x)$; we then update the EMA according to Eq. (14). We report (i) true vs. EMA trajectories, (ii) L1 tracking error over time for different γ, and (iii) steady-state error vs. drift magnitude, which empirically follows the predicted $\mathcal{O}(\gamma) + \mathcal{O}(\bar{d}/\gamma)$ trend.

Protocol. $T = 2000$ iterations, $K = 3$, batch size 256, label smoothing toward the true prior (soft=0.2). We use $\gamma \in \{0.02, 0.05, 0.10\}$ and drift magnitude max_step $\in \{0.00, 0.01, 0.02, 0.04, 0.06\}$.

Transient spikes and adaptation. The small spikes visible in Fig. 6 coincide with moments when the underlying true prior drifts to a new value in the toy stream. Because the EMA update in Eq. (14) averages over past estimates, it cannot instantaneously follow these abrupt changes. The temporary deviation and its exponential recovery are the expected transient response of the recursion analyzed in Theorem 5: after a drift step of size Δ_i, the error decays roughly as $(1-\gamma)^k\|\Delta_i\|$ for k subsequent iterations. These short-lived spikes therefore confirm the finite adaptation speed and bounded tracking error $\mathcal{O}(\bar{d}/\gamma)$ predicted by the theory.

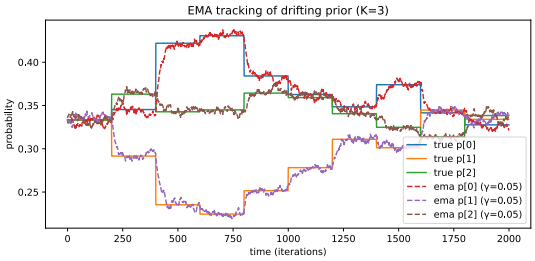

Figure 6. **True vs. EMA trajectories** (K=3, $\gamma = 0.05$). The EMA closely tracks the drifting prior; brief spikes appear when the true prior changes, reflecting the transient adaptation predicted by Theorem 5.

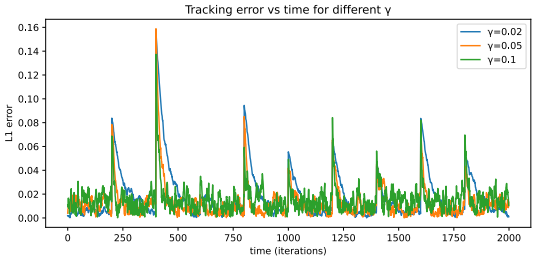

Figure 7. **L1 tracking error vs. time** for different γ. Larger γ adapts faster but with higher steady-state variance; smaller γ is smoother but slower, matching the bound.

Takeaway. The toy results mirror the theory: EMA tracks the underlying model-induced prior with a noise–adaptation trade-off governed by γ. This supports the practical choice of a small, fixed γ across datasets.

11. Generalization scope.

Although the primary experiments employ the LUCIR backbone for fair comparison with prior LTCIL work [21, 29], the design of AdaPrior is inherently modular. Since it requires only the model's predicted posteriors to estimate priors, no architectural coupling is assumed. This allows

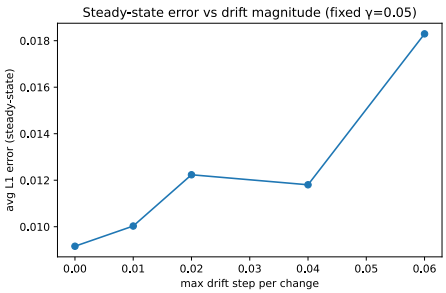

Figure 8. **Steady-state error vs. drift magnitude** (fixed $\gamma = 0.05$). Error increases approximately linearly with drift, consistent with $\mathcal{O}(\bar{d}/\gamma)$.

Methods	CIFAR100-LT		ImageNet-subset-LT	
	5 Tasks	10 Tasks	5 Tasks	10 Tasks
LUCIR+CE	37.07	37.45	47.90	48.98
+AdaPrior*	**42.08**	**41.91**	**54.98**	**56.27**
PODNET	37.84	37.91	55.11	54.18
+AdaPrior*	**38.93**	**38.35**	**58.95**	**54.79**

Table 8. AdaPrior over LUCIR and PODNET on Shuffled LT setting. AdaPrior* denotes the simple post-hoc only variant.

seamless integration with other continual-learning frameworks (e.g., PODNET, DER, or transformer-based incremental models) without re-training structural components. Table 8 confirms similar gains when plugged into PODNET, reinforcing architecture-agnostic applicability.

12. iNaturalist18-subset dataset

In this paper we propose using a realistic high imbalance dataset to evaluate the performance of LTCIL frameworks. We create this dataset by sampling a subset of 100 classes from the iNaturalist18 [41] dataset widely used in the long-tailed learning literature. Unlike other datasets iNaturalist18 naturally posses a high imbalance factor of the order of 435 and consists of 89% of classes with number of samples less than 100. We create the iNaturalist18-subset dataset by maintaining this high imbalance factor. But, inspired from the findings in [15] that the forgetting is positively correlated with the number of samples per class, we conclude that directly maintaining the same proportion of tail classes in the subset will reduce the impact the imbalance have on the extent of forgetting. Therefore we make the subset sufficiently challenging for LTCIL tasks by adequately sampling classes from the head and tail categories. The proposed iNaturalist18-subset dataset consists of 100 classes with an imbalance factor of ≈ 435 and the proportion of many (> 800), medium(< 800 and > 100) and few (< 100) classes as 15%, 35% and 50% respectively.

13. Additional Experiments

Different imbalance factors on ImageNet-subset-LT: We also substantiate the effect of different imbalance factors on ImageNet-subset-LT dataset in Table. 9. We show results for imbalance factors 200 and 10. One may note that simple baseline LUCIR+CE has poor performance and is inferior to many baselines in the table. However, when using proposed AdaPrior Loss, the performance improves by a significant margin and outperforms all the baselines. In particular, for a high imbalance factor of 200, the performance boost is 7.5% and 5.67% for 5 and 10 tasks settings, respectively. Further, when using Full AdaPrior, there is additional performance boost in all cases.

Full AdaPrior on large scale data: In order to depict the versatality of our method and how well it can also help perform well on large scale dataset, we show our result on the 1000 class ImageNet-LT dataset under the LFH setting. Figure 9 shows the performance of Full AdaPrior compared to a few prominent methods on the large scale ImageNet-LT dataset with 1000 classes closely mimicking the continual learning on a large scale imbalanced data. It can be clearly seen that our approach easily outperforms other traditional methods.

Food-101-LT We show comparative analysis in Table. 10 on Food-101-LT [7] dataset. We evaluate the performance using different numbers of tasks for both LFS and LFH settings. One may note that proposed approach outperforms existing approaches and establishes new SOTA on this dataset. For the experiments on the Food-101-LT dataset we follow the same protocol as in [15].

Conventional CIL We now show results on conventional CIL setting for CIFAR100 and ImageNet-subset datasets where the datasets are well balanced. It should be noted

Method	IF=200		IF=10	
	5 Tasks	10 Tasks	5 Tasks	10 Tasks
iCaRL [34]	43.35	45.56	63.15	63.35
BiC [48]	38.68	36.47	60.40	48.05
IL2M [4]	42.01	41.67	54.43	52.46
SSIL [2]	43.33	40.18	60.45	56.39
EEIL-2stage [9]	45.30	44.00	57.28	55.04
LUCIR+GValign [21]	49.84	49.44	65.68	64.37
LUCIR+CE*	44.69	45.27	58.38	57.04
Bayesian Alignment				
AdaPrior Loss	52.19	50.94	65.30	65.51
Full AdaPrior	**53.87**	**52.26**	**66.57**	**66.78**

Table 9. Results on ImageNet-subset-LT dataset for different imbalance factors across multiple task settings. * denotes the reproduced results for LUCIR using linear classifier.

Method	LFS		LFH	
	10 Tasks	20 Tasks	5 Tasks	10 Tasks
iCaRL [34]	18.13	12.5	21.83	21.31
IL2M [4]	16.11	16.27	23.93	22.48
BiC [48]	16.94	16.81	22.8	20.75
SSIL [2]	16.86	15.65	21.65	19.03
EEIL-2stage [9]	19.75	20.02	22.65	22.83
GradReweighting [15]	29.05	26.42	36.84	36.19
LUCIR2stage [18]	27.65	24.68	36.05	35.06
LUCIR+CE*	33.14	37.01	48.56	43.21
Bayesian Alignment				
AdaPrior Loss	40.24	40.3	49.63	50.77
Full AdaPrior	**44.33**	**42.61**	**50.24**	**52.76**

Table 10. Results on the Food-101-LT [7] dataset in the shuffled LTCIL setting. We show the results on both LFS and LFH scenarios across different number of tasks. Baseline results used are from [15] while * denotes the reproduced results for LUCIR using linear classifier.

Method	CIFAR100	ImageNet-subset
iCaRL [34]	47.69	60.85
EEIL [9]	51.65	53.05
IL2M [4]	51.49	51.47
BiC [48]	33.56	58.57
WA [55]	35.62	56.95
SSIL [2]	43.52	59.30
LUCIR [18]	59.25	62.56
PODNet [12]	60.50	61.41
FOSTER [42]	59.54	63.82
Gradreweighting [15]	59.31	67.32
LUCIR+LWS* [29]	60.57	66.47
LUCIR+GVAlign* [21]	61.23	66.60
Bayesian Alignment		
AdaPrior Loss	61.57	67.39
Full AdaPrior	**63.00**	**69.90**

Table 11. Result of our approach on conventional CIL setting using LFH setting for 10 tasks settings. * denotes the results from [21] while the remaining baselines are from [15].

here that, in case of conventional CIL there still exists an imbalance between new task samples and old task samples. In Table. 11 we show results of AdaPrior approach with other baselines. Proposed AdaPrior loss already outperforms all the baselines and the final Full AdaPrior outperforms the closest baseline by 1.77% and 2.58% for CIFAR100 and ImageNet-subset datasets,respectively.

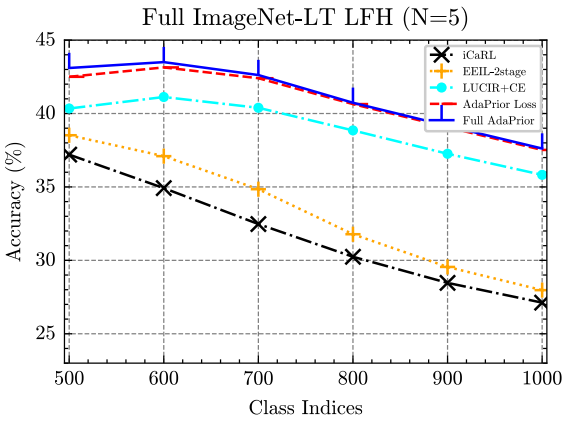

Figure 9. Results on large scale ImageNet-LT dataset with 1000 classes LFH setting.

Algorithm 1 AdaPrior Post-hoc Algorithm

Input:
Model Θ_τ after task τ
Validation data $\mathcal{D}^\tau_{\text{val}} = \left(x_i^\tau, y_i^\tau\right)_{i=1}^{n_\tau}$
Model prior on Training data $\mathcal{D}^\tau$
$P_m(y) = \frac{1}{|\mathcal{D}^\tau|} \sum_{x \in \mathcal{D}^\tau} \Theta_\tau(x).$
Output:
Post-hoc aligned model Θ^a_τ

Start:
Compute $P_t(y)$

$P_t(y) = \frac{1}{n_\tau} \sum_i^{n_\tau} 1_{hot}(y_i)$

Align the logits

for $(\alpha = 0, \alpha < 1.0, \alpha += 0.01)$ **do**
$y_i^{pred} = ArgMax\left(\Theta_\tau(x_i).\left(\frac{P_t(y)}{P_m(y)}\right)^\alpha\right)$
A = Get_Average_Acc$\left(y_i^\tau, y_i^{pred}\right)_{i=1}^{n_\tau}$

if A is the best
α^* = α
end for
$\Theta^a_\tau = \left(\frac{P_t(y)}{P_m(y)}\right)^{\alpha^*}.\Theta_\tau$
Return Θ^a_τ
End

14. Discussion

It should be noted that, under ideal conditions, AdaPrior loss should be enough to remove the effect of data imbal-

Algorithm 2 AdaPrior Loss Algorithm

Input:
Model $\Theta_{\tau-1}$ at task $\tau-1$
Validation data $\mathcal{D}_{\text{val}}^{n_\tau} = \left(x_i^\tau, y_i^\tau\right)_{i=1}^{\tau}$
Output:
Model Θ_τ trained using AdaPrior Loss on τ tasks

Start:
Initialize Model, $\Theta_\tau = \Theta_{\tau-1}$

Initialize $P_m^0(y) = P(y)$

Train using AdaPrior Loss

for $(i = 1, i < n_{iter}, i+ = 1)$ **do**
Compute $P_m^i(y)$
$P_m^i(y) = (1-\gamma)P_m^{i-1}(y) + \gamma \frac{1}{|\mathcal{B}^i|} \sum_{x \in \mathcal{B}^i} P_m(y|x)$

$\bar{z}(x, y) = \Theta_\tau(x)$

Compute Loss

$$\mathcal{L}_{AdaPrior} = -\log\left(\frac{e^{\bar{z}(x,y)+\log\frac{P_m(y)}{P_t(y)}}}{\sum_k e^{\bar{z}(x_k,k)+\log\frac{P_m(k)}{P_t(k)}}}\right)$$

Compute gradients $\nabla(\mathcal{L}_{AdaPrior})$
Update parameters $\Theta_\tau \longleftarrow \nabla(\mathcal{L}_{AdaPrior})$
end for
Return Θ_τ
End

ance and Full AdaPrior should not result in any significant improvements. However, due to practical considerations, AdaPrior loss, although achieving superior performance, is not able to fully remove the model bias and when post-hoc AdaPrior is applied, the residual bias is removed resulting in improvements. As mentioned in the paper this usage of post-hoc AdaPrior applied on a model trained using AdaPrior loss is called the Full AdaPrior.

We summarize steps involved in AdaPrior Loss and post-hoc AdaPrior in the form of an algorithm as shown in Algorithms 1 and 2.

References

[1] Hongjoon Ahn, Donggyu Kwak, Hae Beom Lim, Hyeong Kyu Bang, and Taesup Moon. Ss-il: Separated softmax for incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 844–853, 2021. 3

[2] Hongjoon Ahn, Jihwan Kwak, Subin Lim, Hyeonsu Bang, Hyojun Kim, and Taesup Moon. Ss-il: Separated softmax for incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 844–853, 2021. 3, 5, 6, 7, 4

[3] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European Conference on Computer Vision*, pages 139–154, 2018. 3

[4] Eden Belouadah and Adrian Popescu. Il2m: Class incremental learning with dual memory. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 583–592, 2019. 3, 5, 6, 7, 4

[5] S Divakar Bhat, Biplab Banerjee, Subhasis Chaudhuri, and Avik Bhattacharya. Cilea-net: Curriculum-based incremental learning framework for remote sensing image classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:5879–5890, 2021. 3

[6] S Divakar Bhat, Amit More, Mudit Soni, and Surbhi Agrawal. Prior2posterior: Model prior correction for long-tailed learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1289–1298. IEEE, 2025. 3, 4

[7] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *Proceedings of the European Conference on Computer Vision*, pages 446–461. Springer, 2014. 7, 4, 5

[8] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[9] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision*, pages 233–248, 2018. 1, 3, 4, 5, 6

[10] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002. 1, 3

[11] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019. 1, 3

[12] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *Proceedings of European Conference on Computer Vision*, pages 86–102. Springer, 2020. 3, 8, 5

[13] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878–887. Springer, 2005. 3

[14] Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *IEEE international Joint Conference on Neural Networks (IEEE world congress on computational intelligence)*, pages 1322–1328. Ieee, 2008. 1, 3

[15] Jiangpeng He. Gradient reweighting: Towards imbalanced class-incremental learning. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 16668–16677, 2024. 2, 3, 5, 7, 4

[16] Geoffrey Hinton. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3

[17] Youngkyu Hong, Seungju Han, Kwanghee Choi, Seokjun Seo, Beomsu Kim, and Buru Chang. Disentangling label distribution for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6626–6636, 2021. 3

[18] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 831–839, 2019. 3, 4, 6, 5

[19] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5375–5384, 2016. 1

[20] Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Deep imbalanced learning for face recognition and attribute prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(11):2781–2794, 2019. 1

[21] Jayateja Kalla and Soma Biswas. Robust feature learning and global variance-driven classifier alignment for long-tail class incremental learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 32–41, 2024. 1, 2, 3, 5, 6, 7, 8, 4

[22] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019. 2, 3, 4

[23] Donggyu Kim, Jaehong Kim, and Sungroh Yoon. Calibrated decoupled distillation for long-tailed class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 3

[24] Jaehong Kim, Donggyu Kim, Jongseong Jeong, and Sungroh Yoon. Imbalanced continual learning with partitioning reservoir sampling. In *Advances in Neural Information Processing Systems*, 2023. 3

[25] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017. 3

[26] Mengke Li, Yiu-ming Cheung, and Yang Lu. Long-tailed visual recognition via gaussian clouded logit adjustment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6929–6938, 2022. 5

[27] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017. 1, 3

[28] TY Lin, P Goyal, R Girshick, K He, and P Dollár. Focal loss for dense object detection. arxiv 2017. *arXiv preprint arXiv:1708.02002*, 2002. 1, 3

[29] Xialei Liu, Yu-Song Hu, Xu-Sheng Cao, Andrew D Bagdanov, Ke Li, and Ming-Ming Cheng. Long-tailed class incremental learning. In *Proceedings of the European Conference on Computer Vision*, pages 495–512. Springer, 2022. 2, 3, 5, 7

[30] Yaoyao Liu, Bernt Schiele, and Qianru Sun. Adaptive aggregation networks for class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2544–2553, 2021. 3

[31] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-scale long-tailed recognition in an open world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2537–2546, 2019. 1, 3

[32] Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. Long-tail learning via logit adjustment. *arXiv preprint arXiv:2007.07314*, 2020. 2, 3, 4, 8, 1

[33] Jathushan Rajasegaran, Munawar Hayat, Salman H Khan, Fahad Shahbaz Khan, and Ling Shao. Random path selection for continual learning. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[34] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1, 3, 4, 5, 6, 7

[35] Jiawei Ren, Cunjun Yu, Xiao Ma, Haiyu Zhao, Shuai Yi, et al. Balanced meta-softmax for long-tailed visual recognition. *Advances in Neural Information Processing Systems*, 33:4175–4186, 2020. 3, 4, 8

[36] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951. 6

[37] Herbert Robbins and David Siegmund. A convergence theorem for non negative almost supermartingales and some applications. In *Optimizing methods in statistics*, pages 233–257. Elsevier, 1971. 6

[38] Jonathan Schwarz, Wojciech Czarnecki, Jelena Luketina, Agnieszka Grabska-Barwinska, Yee Whye Teh, Razvan Pascanu, and Raia Hadsell. Progress & compress: A scalable framework for continual learning. In *Proceedings of the International Conference on Machine Learning*, pages 4528–4537. PMLR, 2018. 3

[39] Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. *Advances in Neural Information Processing Systems*, 32, 2019. 3

[40] Jingru Tan, Changbao Wang, Buyu Li, Quanquan Li, Wanli Ouyang, Changqing Yin, and Junjie Yan. Equalization loss for long-tailed object recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11662–11671, 2020. 3

[41] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Proceedings of the IEEE/CVF Conference*

on Computer Vision and Pattern Recognition, pages 8769–8778, 2018. 7, 4

[42] Fu-Yun Wang, Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Foster: Feature boosting and compression for class-incremental learning. In *Proceedings of the European Conference on Computer Vision*, pages 398–414. Springer, 2022. 5, 7

[43] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X Yu. Long-tailed recognition by routing diverse distribution-aware experts. *arXiv preprint arXiv:2010.01809*, 2020. 3

[44] Xi Wang, Xu Yang, Jie Yin, Kun Wei, and Cheng Deng. Long-tail class incremental learning via independent sub-prototype construction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28598–28607, 2024. 2, 3, 7

[45] Yu-Xiong Wang, Deva Ramanan, and Martial Hebert. Learning to model the tail. *Advances in Neural Information Processing Systems*, 30, 2017. 1

[46] Max Welling. Herding dynamical weights to learn. In *Proceedings of the International Conference on Machine Learning*, pages 1121–1128, 2009. 3

[47] Chenshen Wu, Ching-Yao Liu, Zheng Kuang, Subhransu Maji, and Larry S. Davis. Class-balanced incremental learning for long-tailed recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16236–16245, 2022. 3

[48] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 374–382, 2019. 3, 5, 6, 7, 4

[49] Liuyu Xiang, Guiguang Ding, and Jungong Han. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *Proceedings of the European Conference on Computer Vision*, pages 247–263. Springer, 2020. 3

[50] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3014–3023, 2021. 3

[51] Xi Yin, Xiang Yu, Kihyuk Sohn, Xiaoming Liu, and Manmohan Chandraker. Feature transfer learning for face recognition with under-represented data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5704–5713, 2019. 3

[52] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. *arXiv preprint arXiv:1708.01547*, 2017. 3

[53] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *Proceedings of the International Conference on Machine Learning*, pages 3987–3995. PMLR, 2017. 3

[54] Hongyi Zhang. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 7, 8

[55] Bowen Zhao, Xi Xiao, Guojun Gan, Bin Zhang, and Shu-Tao Xia. Maintaining discrimination and fairness in class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13208–13217, 2020. 5, 7

[56] Zhisheng Zhong, Jiequan Cui, Shu Liu, and Jiaya Jia. Improving calibration for long-tailed recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16489–16498, 2021. 3