MIST: Reliable Streaming Decision Trees for Online Class-Incremental Learning via McDiarmid Bound

Phu-Hoa Pham[⋆,1,2,4]
23122030@student.hcmus.edu.vn

Chi-Nguyen Tran[⋆,1,2,4]
23122044@student.hcmus.edu.vn

Phu-Quy Nguyen-Lam[1,2,4]
23122048@student.hcmus.edu.vn

Dao Sy Duy Minh[1,2,4]
23122041@student.hcmus.edu.vn

Huynh Trung Kiet[1,2,4]
23122039@student.hcmus.edu.vn

Long Tran-Thanh[†,3]
long.tran-thanh@warwick.ac.uk

[1] Faculty of Information Technology, University of Science, Ho Chi Minh City, Vietnam
[2] Vietnam National University, Ho Chi Minh City, Vietnam
[3] Department of Computer Science, University of Warwick, Coventry, United Kingdom
[4] Compact AI

Abstract

Streaming decision trees are natural candidates for open-world continual learning, as they perform local updates and maintain explicit decision boundaries. Despite these, they still fail in online class-incremental learning due to two coupled miscalibrations: (i) their split criterion grows unreliable as the class count K expands, and (ii) the absence of knowledge transfer at split time. The first issue is linked to the range of Information Gain, which intrinsically scales with $\log_2 K$. Consequently, any Hoeffding-style confidence radius derived from it must inevitably grow with the class count, preventing a K-independent radius under this Information-Gain-based calibration, taking away the potential benefits of applying streaming decision trees to continual learning. To fix this issue, we present MIST (McDiarmid Incremental Streaming Tree), which addresses these two issues through three integrated components: (i) a McDiarmid confidence radius for Gini splitting based on a tight, K-independent sensitivity bound that acts as a structural regulariser; (ii) a Bayesian inheritance protocol that projects parent statistics to child nodes via truncated-Gaussian moments, with variance reduction under the stated prior assumptions; and (iii) per-leaf KLL quantile sketches that support both continuous threshold evaluation and geometry-adaptive leaf prediction from a single data structure. On standard and stress-test tabular streams, MIST is competitive with global parametric methods on near-Gaussian benchmarks, although the latter achieve higher mean accuracy on the standard benchmarks. Its sketch-based variant, MIST-K, shows benefits on specific non-Gaussian geometries. Code is available at GitHub repository.

1 Introduction

Continual learning systems that operate in an open world must absorb new classes as they arrive, without access to historical data and without degrading what was previously learned. In tabular

⋆ Equal contribution. † Corresponding author: long.tran-thanh@warwick.ac.uk

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

settings, where large pretrained representations are unavailable, this challenge must be resolved directly from the raw feature stream under strict memory constraints.

Streaming decision trees are a natural candidate for this regime. Their updates are local, as only the leaf reached by an arriving sample is modified. As such, their memory depends on the number of leaves and the statistics retained at each leaf, and they share no global representation across classes. For example, in the foundational VFDT framework, which laid down the basis for streaming decision trees, a decision boundary, once committed via the Hoeffding bound, is not revised. Instead, splits are permanent, preserving committed partitions. Leaf predictions can still change as statistics are updated, so this alone does not prevent forgetting. This structural potential, however, has never been fully realised. In practice, classical streaming trees collapse in the online class-incremental learning (Online CIL) setting. The reason is more fundamental than it may first appear, and tracing it to its root reveals two coupled failures that must be resolved simultaneously.

The root cause: Premature split. Existing streaming trees typically trigger splits when the empirical gain gap exceeds a Hoeffding confidence radius derived from Information Gain [17, 45, 38, 8, 11]. This radius grows as $\log_2 K/\sqrt{n}$, which comes from the structural implication of Information Gain's range scaling with $\log_2 K$, and becomes unreliable as K grows without bound, either delaying necessary splits or triggering prematurely, making streaming decision trees unsuitable for online CIL settings.

A potential solution and its cold-start split issue. A natural way to fix this premature split issue is to switch to Gini impurity, whose range is bounded in $[0, 1)$ regardless of K, making a class-count-independent radius achievable in principle. However, identifying a tight decision radius for the Gini split gain is non-trivial. To address this, we derive a tight bounded-difference constant for Gini split gain under single-sample replacement and prove it is unimprovable for arbitrary K, yielding a confidence radius without a multiplicative $\log_2 K$ range factor.

This calibration is conservative by design, acting as a structural regulariser. It does, however, expose a second point of failure: when a leaf splits, its children inherit nothing from their parent and must rebuild predictive distributions from scratch, producing unreliable predictions precisely when the tree adapts most aggressively; further splits from sparse children compound the instability.

The coupled failures: premature and cold-start splits. The two failures reinforce each other, as premature splits produce sparse children, whose cold start inflates impurity estimates, triggering further premature splits. To fix this, we break this cycle by addressing both simultaneously: Conservative splitting ensures children are born from data-rich parents, improving the use of inherited statistics. To do so, we use Bayesian inheritance and translate those statistics into tight posterior distributions from the first sample onward. Crucially, variance reduction from inheritance scales with the parent sample count at split time, so the more conservative the criterion, the stronger the inheritance guarantee.

Sketching for both splitting and prediction. The two components above handle *when* to split and *how* to initialise after splitting. However, both would normally rely on past raw data, which is typically unavailable in the online CIL setting. To address this, we propose a new, sketching-based approach to continuous threshold evaluation without retaining raw data. In particular, per-leaf, label-aware quantile sketches generate candidate thresholds from empirical class-conditional distributions, and the same structure powers geometry-adaptive leaf prediction—parametric Gaussian on near-Gaussian data, distribution-free otherwise. Both configurations are formalised in Section 4.

Contributions. In this paper, we present MIST (McDiarmid Incremental Streaming Tree) to implement the three ideas above as follows:

- **Tight McDiarmid calibration for Gini splitting.** We derive the tight bounded-difference constant for multi-class Gini split gain and prove that it is sharp for arbitrary K, yielding a confidence radius without the Information-Gain range factor (Corollary 4.1). Candidate multiplicity still enters logarithmically. This calibration acts as a structural regulariser alongside the inheritance mechanism.
- **Bayesian post-split stabilisation.** We introduce parent-to-child projection via exact truncated-Gaussian conditional moments for the split feature and Naive Bayes mass rescaling for remaining features. Variance reduction guarantees cover both class-probability estimates and feature-parameter estimates (Proposition 5.5) with both posteriors tightening in proportion to the parent sample count at split time.

- **Unified label-aware sketching for structure and prediction.** Per-leaf KLL quantile sketches support continuous split threshold evaluation without raw-data retention; the same sketches power an optional non-parametric leaf predictor under a Naive Bayes factorisation at no additional storage cost. Theorem 5.3 provides a composed error bound jointly controlling statistical variance and sketch rank error.
- **Regime-aware empirical characterisation.** We identify and explain conditions under which the parametric and non-parametric leaf predictors are respectively preferable, supported by prequential accuracy curves, ablation studies, and stress-test experiments on synthetic streams with non-Gaussian class geometry.

2 Related Work

Due to space limits, this section covers only the work most relevant to ours and defers a more detailed summary of the online continual learning literature to Appendix A.

Streaming decision trees in class-incremental settings. VFDT [17] established Hoeffding-style confidence tests for incremental splitting, with EFDT [45] adding continuous re-evaluation and HAT [8] incorporating adaptive drift detection via ADWIN [7]. These methods treat distributional shifts as concept drift to be discarded, directly conflicting with Online CIL where class arrivals are permanent structural events; disabling their drift detectors recovers substantial accuracy on our benchmarks, confirming that the failure stems from protocol misalignment rather than from the algorithms themselves. VFDT zero-initialises child leaves; VFDTc [20] introduced Naive Bayes leaves but retained zero-initialisation at split time. Korycki and Krawczyk [38] motivate knowledge transfer at structural boundaries under replay; our work addresses the analogous problem under the strict exemplar-free constraint with formal variance reduction guarantees. Thompson-sampling trees [11] are restricted to categorical attributes and a closed-world class set, making direct comparison inappropriate.

Concentration bounds for Gini-based splitting. Rutkowski et al. [57] established that McDiarmid's inequality applies to Gini split gain with constant $c_i = 8/n$—the first K-independent sensitivity constant. De Rosa and Cesa-Bianchi [16] tightened this to $4/n$ for $K = 2$ only. Our work closes the remaining gap: $4/n$ is tight for arbitrary K, while candidate-count dependence remains logarithmic in the operational radius. C-Tree [16] derives refined large-deviation intervals using problem-dependent parameters, but its analysis is restricted to $K = 2$; an empirical comparison is omitted as extending its implementation to $K > 2$ is non-trivial.

Global methods for exemplar-free tabular CIL. SLDA [29] uses a shared-covariance Gaussian discriminant, SQDA [29] relaxes this to per-class Gaussians, NCM [47] tracks class centroids, and RLS [73] maintains a shared linear classifier via recursive least squares. These are strong references on the standard benchmarks. Shared-covariance or shared-representation updates can couple class decisions, whereas SQDA and NCM retain class-specific summaries. These structural distinctions alone do not establish a forgetting or accuracy ordering. MIST offers local partitioning and flexible marginal densities, with limitations from univariate splitting and feature factorisation.

Non-parametric and sketch-based prediction in streaming settings. KDE Naive Bayes [35, 48] improves accuracy under non-Gaussian class-conditionals, but raw-sample KDE is infeasible in an exemplar-free stream. Prior work [6] uses additive-error quantile sketches for split threshold evaluation without connecting the sketch to the leaf predictor. Our work unifies both roles: the same per-leaf KLL sketches [36] that support continuous split evaluation also power a distribution-free leaf predictor, yielding a streaming analog of KDE Naive Bayes with formal error guarantees. KLL's efficient update mechanism bounds the size of each sketch, and its high-probability rank guarantees feed directly into Theorem 5.3, with $O(k)$ memory and $O(\log k)$ amortised update cost.

3 Preliminaries and Problem Setting

3.1 Strict Online CIL

We consider learning from an infinite stream $\mathcal{S} = \{(x_t, y_t)\}_{t=1}^{\infty}$ where the label space $\mathcal{Y}_t$ expands monotonically and $K = \lim_{t\to\infty} |\mathcal{Y}_t|$ is unbounded and unknown a priori. An algorithm operates in the *strict Online CIL* regime if it satisfies: (i) single-pass processing; (ii) exemplar-free memory; (iii) anytime classification. New-class arrivals are permanent structural events, not distributional

shifts to be discarded. Each split decision is evaluated locally on the n samples accumulated at a leaf; class-conditional distributions $p(x \mid y = c)$ are stationary by protocol, and samples at a leaf are treated as an independent batch [17], satisfying McDiarmid's independence requirement.

3.2 Gini Split Gain and Bounded Differences

Let S be n samples at a leaf with empirical class proportions $p_c = n_c/n$. The Gini impurity and split gain for a binary partition (S_L, S_R) are:

$$G(S) = 1 - \sum_{c=1}^{K} p_c^2, \qquad F(S) = G(S) - \tfrac{n_L}{n} G(S_L) - \tfrac{n_R}{n} G(S_R). \tag{1}$$

A function $f : (\mathcal{X} \times \mathcal{Y})^n \to \mathbb{R}$ satisfies the *bounded difference property* with constants $c_1, \ldots, c_n$ if replacing any single input changes f by at most c_i. McDiarmid's inequality then gives:

$$\mathbb{P}[\,|f(S) - \mathbb{E}[f]| \geq \varepsilon] \leq 2 \exp\left(-\frac{2\varepsilon^2}{\sum_i c_i^2}\right). \tag{2}$$

The central theoretical question of this paper is: what is the tight value of c_i for the Gini split gain F? The answer, derived in Section 5, determines the sensitivity term in the confidence radius used by MIST; candidate multiplicity still enters logarithmically.

4 The MIST Framework

Online CIL exposes two coupled failures in streaming trees: (i) the Information-Gain split test becomes miscalibrated as classes arrive ($\log_2 K$ range); and (ii) children created by a split start without local evidence, and their cold start feeds further unreliable splits. MIST answers them with a McDiarmid-calibrated Gini split test (Section 4.1) and Bayesian inheritance (Section 4.2). Both must work without replaying data, which per-leaf KLL sketches make possible (Section 4.3). Each subsection opens with the problem it addresses (Figure 1). MIST comes in two variants that share the split test, inheritance, and sketches and differ only in the leaf predictor: MIST-G predicts from Gaussian per-feature running statistics, MIST-K from sketch-derived marginal densities (Section 4.3).

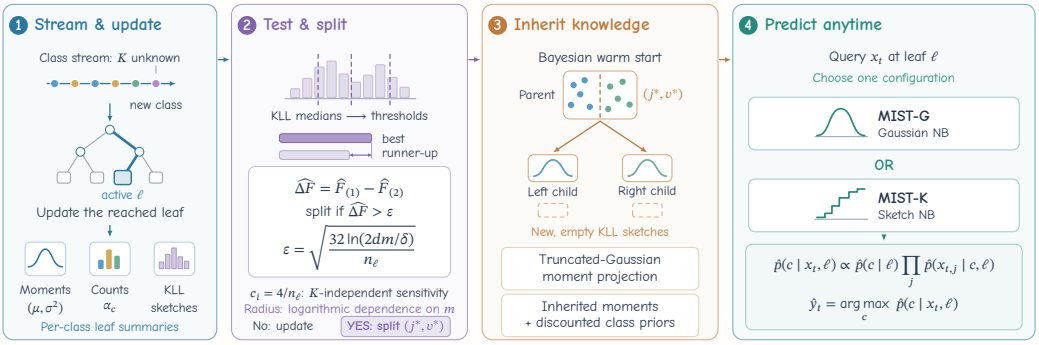


Figure 1: Overview of MIST. Stream observations update only the reached leaf. KLL sketches propose candidates for the McDiarmid-calibrated Gini split test. Bayesian inheritance transfers moments and discounted class priors to new children, whose KLL sketches start empty. Prediction uses either Gaussian (MIST-G) or sketch-based (MIST-K) Naive Bayes likelihoods.

4.1 Sensitivity-Calibrated Split Criterion

MIST triggers a split when the empirical Gini gain gap $\widehat{\Delta F}$ between the best and second-best candidate feature exceeds a confidence radius (the *GapTest*). The radius is derived from the tight, class-count-independent McDiarmid calibration in Section 5; the candidate count still enters the operational radius logarithmically.

Corollary 4.1 (Operational McDiarmid Radius)**.** *Consider one split decision under the sample-independence assumptions stated in Section 3. Let d be the number of candidate features and m the number of candidate thresholds per feature. Applying the McDiarmid bound of Theorem 5.1 and a*

union bound over the dm candidates, with probability at least $1-\delta$, the empirically best candidate also has the largest expected Gini gain whenever its gain gap over the second-best candidate satisfies:

$$\widehat{\Delta F} > \varepsilon, \qquad \varepsilon = \sqrt{\frac{32\ln(2dm/\delta)}{n}}. \tag{3}$$

Remark 4.2. *The factor of 32 arises from the GapTest structure: concluding $\mathbb{E}[F_1] > \mathbb{E}[F_2]$ requires controlling both tails simultaneously, so $\varepsilon = 2\eta$ where $\eta = \sqrt{8\ln(2dm/\delta)/n}$ is the per-tail McDiarmid radius, giving $\varepsilon = \sqrt{32\ln(2dm/\delta)/n}$. The union bound over d features and m thresholds enters only additively inside the logarithm as $\ln(2dm/\delta)$, not multiplicatively into the constant 32.*

The radius has no multiplicative $\log_2 K$ range factor of the compared Information-Gain-based Hoeffding criterion. It can still depend on the class count through $m \leq K_{\text{local}} - 1$, where K_{local} is the number of classes present at the leaf: the threshold count from Section 4.3 contributes an $O(\sqrt{\ln m})$ factor.

Role as structural regulariser. The conservatism of this criterion is a feature: replacing the McDiarmid GapTest with a more aggressive criterion increases tree size substantially at the cost of only marginal accuracy gains (Table 3), consistent with the coupled failure mode analysis (Figure 4).

4.2 Bayesian Knowledge Inheritance

Motivation. A leaf that has accumulated sufficient evidence to split has rich class-conditional statistics that should not be discarded. Standard streaming trees initialise child leaves without prior information, forcing them to reconstruct predictive distributions from scratch. In a strict exemplar-free setting, this cold-start phase can be severe: the children must accumulate enough new samples to form reliable predictions before they are useful, and any further splits triggered from sparse children compound the problem. The conservative split criterion of Section 4.1 mitigates this by ensuring that splits occur only from data-rich parents. We use Bayesian knowledge inheritance to complete the picture by ensuring that the resulting children begin with statistically grounded priors.

Truncated-Gaussian projection at split time. MIST uses Gaussian Naive Bayes (GNB) leaves, parameterised by per-class, per-feature Welford accumulators (μ_j^c, σ_j^c) and Dirichlet class-probability pseudo-counts. When a parent splits on feature j at threshold v, let $\zeta = (v - \mu_j^c)/\sigma_j^c$. Exact truncated-Gaussian moment projection (Figure 2b) initialises the split-feature statistics as:

$$\mu_j^{\text{left},c} = \mu_j^c - \sigma_j^c \frac{\phi(\zeta)}{\Phi(\zeta)}, \qquad \mu_j^{\text{right},c} = \mu_j^c + \sigma_j^c \frac{\phi(\zeta)}{1-\Phi(\zeta)}, \tag{4}$$

with corresponding variance updates given in Appendix J.1. For every non-split feature $k \neq j$, under the Naive Bayes conditional independence assumption $p(X_k \mid c, X_j \leq v) = p(X_k \mid c)$, parameters are inherited unchanged. Class mass is reapportioned as $n_c^{\text{left}} \leftarrow \alpha\,\Phi(\zeta)\,n_c^{\text{parent}}$ and $n_c^{\text{right}} \leftarrow \alpha\,(1-\Phi(\zeta))\,n_c^{\text{parent}}$, with Dirichlet pseudo-counts reweighted identically by factor α.

For categorical features encoded as one-hot indicators, the split is binary and no moment projection is required: each child $s \in \{0,1\}$ inherits class mass $n_c^s \leftarrow \alpha\, n_c(s)$, where $n_c(s) = \#\{i : y_i = c,\, x_i^{(\text{attr})} = s\}$ is the empirical co-occurrence count at the parent leaf, $\alpha \in (0,1]$ is the inheritance discount (Section 5.3), and Laplace smoothing at prediction time ensures non-zero posteriors for all classes.

The inherited prior decays at rate $O(1/n_{\text{child}})$ as new samples accumulate, allowing each child to specialise to its local geometry while retaining the benefit of the parent's experience during the critical early phase. This projection applies to both configurations (Section 4.3): Gaussian moments initialise class-probability priors and apportion class mass after every split. MIST-G inherits the full Gaussian leaf; MIST-K's KLL sketches start empty (Algorithm 2), so feature likelihoods require new samples, though Dirichlet priors are immediately available.

Joint design with the split criterion. The variance reduction from inheritance scales inversely with the parent sample count at split time (Proposition 5.5): a split triggered too early yields a weaker prior and larger cold-start variance for the child. The McDiarmid GapTest ensures splits occur only from sufficiently data-rich parents, improving the use of the inherited statistics. The two mechanisms

Algorithm 1 MIST: Update and Predict

Require: Sample (x_t, y_t), tree $\mathcal{T}$, discount α, failure probability δ, grace period g

Route x_t to leaf ℓ
Update Welford accumulators (μ_j^c, σ_j^c) and Dirichlet pseudo-counts at ℓ with (x_t, y_t)
Update per-class KLL sketch at ℓ with $x_{t,j}$ for each feature j
if $n_\ell \geq g$ **and** GAPTEST(ℓ, δ) **then**
    $j^*, v^* \leftarrow$ best feature and threshold from sketch medians
    SPLIT(ℓ, j^*, v^*, α) {Algorithm 2}
end if
return $\arg\max_c \hat{p}(c \mid x_t, \ell)$

function GAPTEST(ℓ, δ) {rule of Corollary 4.1; guarantee holds per decision}
    $m \leftarrow K_{\text{local}}^{(\ell)} - 1$ {at most $K_{\text{local}}^{(\ell)} - 1$ candidate thresholds per feature; see Section 4.3}
    $\varepsilon \leftarrow \sqrt{32 \ln(2dm/\delta)/n_\ell}$
    return $\widehat{F}_{(1)} - \widehat{F}_{(2)} > \varepsilon$

reinforce each other: the more conservative the split criterion, the richer the parent statistics at split time, and therefore the stronger the variance reduction guarantee.

4.3 Label-Aware Sketching and the Unified Dual Role of KLL

Threshold search and prediction must work without replaying raw data, and one Gaussian per class can misfit multimodal or skewed marginals. MIST maintains per-leaf, per-class KLL quantile sketches of capacity k. A sketch for class c at leaf ℓ is created only when samples of class c are routed there, giving per-leaf memory $O(d \cdot K_{\text{local}}^{(\ell)} \cdot k)$. When useful partitions form, local class sparsity can reduce storage relative to the worst-case bound $O(dkL_tK_t)$ (Appendix G.2). Neither sparsity nor decreasing total memory is guaranteed; Appendix K.4 reports measured storage and allocation peaks.

Split threshold evaluation. Candidate thresholds are midpoints between adjacent class-conditional medians from active sketches (Figure 2a), yielding at most $m \leq K_{\text{local}} - 1$ candidates per feature, exactly the m in Eq. (3), so the union bound covers only candidates actually considered. For categorical attributes encoded as one-hot indicators, $m \leq 1$ and Eq. (3) remains valid without modification.

Leaf prediction: MIST-G and MIST-K. The same per-class sketches that generate split thresholds power an optional non-parametric leaf predictor. In the default configuration (MIST-G), leaf likelihoods are computed from the Gaussian parameters maintained by the Welford accumulators; sketches contribute only to split threshold evaluation. In the non-parametric configuration (MIST-K), leaf likelihoods are instead estimated from sketch-derived per-feature CDFs under a Naive Bayes factorisation:

$$\hat{p}(x_j \mid c) = \frac{\widehat{F}_{j,c}(x_j + h_j) - \widehat{F}_{j,c}(x_j - h_j) + \epsilon_s}{2h_j + \epsilon_s}, \tag{5}$$

where $\widehat{F}_{j,c}$ is the sketch-derived CDF, h_j is the inter-quantile spacing at x_j (local bandwidth), and $\epsilon_s > 0$ is a dimensionless smoothing regulariser that prevents division by zero when $h_j \to 0$; its contribution is negligible at interior points where $h_j \gg \epsilon_s$.

Regime-dependent behaviour. MIST-G is preferable when class-conditional distributions are approximately unimodal Gaussian, a condition empirically indicated by strong SLDA/SQDA performance on the same benchmark. MIST-K, the empirical default when the marginal shape is unknown (Appendix K.3), avoids the systematic bias that the Gaussian leaf assumption introduces under strongly non-Gaussian geometry (multimodal, angular, or heavily skewed; Figure 2c). The dual role of the KLL sketch is not an engineering convenience: the same data structure that makes continuous split evaluation memory-feasible also captures the distributional information needed for geometry-adaptive prediction.

5 Theoretical Analysis

The central result of this section is Theorem 5.1: a tight, K-independent bounded-difference constant for Gini split gain. This is what makes Corollary 4.1 valid in an open world, and Information-Gain-based Hoeffding criteria do not provide it. Theorem 5.3 extends it to the sketched setting, and Proposition 5.5 quantifies the variance reduction from Bayesian inheritance.

5.1 Tightness of the Gini Sensitivity

Theorem 5.1 (Tightness of the Gini Sensitivity)**.** *Let S be a sequence of $n \geq 2$ samples and $F(S)$ the Gini split gain defined in Eq.* (1)*. For any single-sample replacement $z_i \to z_i'$, the bounded-difference constant satisfies $c_i \leq 4/n$. Moreover, this rate is tight: for any $\gamma > 0$, there exists a configuration S and a replacement such that $|F(S) - F(S^{(i)})| > 4/n - \gamma$.*

Proof sketch. Case A (same-child routing): replacing z_i with z_i' in child L leaves n_L, n_R, and $G(S_R)$ unchanged, so only $G(S)$ and $G(S_L)$ vary; each changes by at most $2/n$ and $2/n_L$, giving $|\Delta F| \leq 2/n + (n_L/n)(2/n_L) = 4/n$. Case B (cross-routing): direct expansion over disjoint class proportions gives $n|\Delta F| \leq 2 < 4$ (Appendix J.6). Tightness: binary labels with $\lfloor \sqrt{n/2} \rfloor$ left-child samples yield $|\Delta F| \to 4/n$ as $n \to \infty$. □

Remark 5.2. *Rutkowski et al. [57] showed $c_i = 8/n$ for arbitrary K; De Rosa and Cesa-Bianchi [16] tightened this to $4/n$ for $K = 2$ only, without establishing tightness. Theorem 5.1 closes the gap: $4/n$ is tight for all K, so its sensitivity constant is independent of K. The operational radius of Corollary 4.1 retains logarithmic dependence on candidate multiplicity, which can grow with the local class count.*

5.2 Composed Split Error with Sketched Features

Theorem 5.3 (Composed Split Error at Fixed Threshold)**.** *Let F^* be the expected true Gini gain and $\widehat{F}$ the estimate from a quantile sketch with rank error $\varepsilon_{\text{sketch}}$. For any $\delta, \delta_{\text{sketch}} \in (0,1)$, with probability at least $1 - \delta - \delta_{\text{sketch}}$:*

$$|\widehat{F} - F^*| \leq \underbrace{\sqrt{\frac{8\ln(2dm/\delta)}{n}}}_{\textit{statistical}} + \underbrace{4\varepsilon_{\text{sketch}}}_{\textit{sketch}}. \tag{6}$$

The statistical term uses factor 8 rather than 32 because bounding $|\widehat{F} - F^*|$ at a fixed threshold requires only a two-sided single-quantity concentration argument, not the simultaneous two-tail control needed by the GapTest in Eq. (3)*. Full proof in Appendix J.3. The factor $4\varepsilon_{\text{sketch}}$ follows from Theorem 5.1: rank error shifts routing counts by $n\varepsilon_{\text{sketch}}$, costing $(4/n) \cdot n\varepsilon_{\text{sketch}}$ in Gini gain.*

Corollary 5.4 (Full Composed Error with Optimization Displacement)**.** *Under Gaussianity of $X_j \mid c$ and finite condition number $\kappa = \sigma_{\max}/\sigma_{\min}$, with probability at least $1 - \delta - \delta_{\text{sketch}}$:*

$$F(S;\, v^*) - F(S;\, \hat{v}^*) \leq \sqrt{\frac{8\ln(2dm/\delta)}{n}} + 4(1+\kappa)\,\varepsilon_{\text{sketch}}, \tag{7}$$

where $\hat{v}^*$ is the sketch-derived threshold. The additional $4\kappa\,\varepsilon_{\text{sketch}}$ over Theorem 5.3 accounts for threshold displacement: KLL rank error $\varepsilon_{\text{sketch}}$ implies $|\hat{v}^* - v^*| \leq \varepsilon_{\text{sketch}}\,\sigma_{\max}\sqrt{2\pi}$ under Gaussianity, and by the Lipschitz constant of Lemma J.1 ($L_v = O(1/\sigma_{\min})$), this propagates to a Gini loss of $4\kappa\,\varepsilon_{\text{sketch}}$. Accuracy saturates at $k \geq 32$ and the gap to exact-threshold evaluation falls below 0.1% (Table 9), confirming the displacement term is negligible in practice. Full proof in Appendix J.4.

5.3 Variance Reduction from Bayesian Inheritance

Proposition 5.5 (Post-Split Variance Reduction)**.** *Let $\alpha \in (0,1)$ be the inheritance discount and $S_0 = \alpha n_{\text{parent}}$ the inherited pseudo-count total.* (i) Class probabilities. *Under the Dirichlet inheritance prior, the posterior variance satisfies for all $t \geq 0$:*

$$\text{Var}(\theta_c \mid \mathcal{D}_t) \leq \tfrac{1}{4(S_0+1)} = O\Big(\tfrac{1}{\alpha n_{\text{parent}}}\Big). \tag{8}$$

(ii) Feature parameters. *Under the NIG conjugate prior with $n_0 = \alpha n_c^{\text{parent}}$ pseudo-observations, the marginal posterior of μ_j^c is Student-t with $\nu = n_0$ degrees of freedom and scale $(\sigma_j^c)^2/n_0$. Its variance exists when $n_0 > 2$ and equals:*

$$\text{Var}(\mu_j^c \mid \mathcal{D}_0) = \frac{(\sigma_j^c)^2}{n_0} \cdot \frac{\nu}{\nu - 2} = \frac{(\sigma_j^c)^2}{\alpha n_c^{\text{parent}} - 2} = O\Big(\tfrac{1}{\alpha n_c^{\text{parent}}}\Big). \tag{9}$$

For the split feature, $(\sigma_j^c)^2$ is replaced by $\sigma^2_{\text{trunc}} \leq (\sigma_j^c)^2$. Full proof in Appendix J.5.

Remark 5.6. *Under the stated prior model, posterior variance decreases with inherited mass. Conservative splitting can retain more parent observations before a split, but does not maximise their number or guarantee representative child priors. The tested values $\alpha \geq 0.2$ recover near-full performance in Table 9; the mismatch diagnostic in Appendix K.2 shows why this is not universal.*

6 Experiments

6.1 Experimental Setup

Datasets. We evaluate on two benchmark categories. *Standard benchmarks* include two synthetic Gaussian mixtures (Synth-10, Synth-50) and eight real-world tabular datasets: Covertype, Split-MNIST, Pendigits, Shuttle, Letter, Wine, Iris, and HAR. Categorical attributes are one-hot encoded before streaming. *Stress-test synthetics* (Section 6.4) isolate specific failure modes of parametric leaf predictors: Multimodal, Skewed, AngularSectors, Antipodal, Ring, HeavyTail, NoisyFeature, and ConceptDrift. For Split-MNIST, raw pixels (d=784) are reduced to d=50 via PCA fitted on the first 1,000 task-0 samples only. The frozen projection is then applied uniformly to all methods.

Baselines. We compare two groups of methods. *Tree-based:* VFDT, Rutkowski, EFDT, HAT, and ARF-NR (10 trees, drift detection disabled). EFDT and HAT under default drift-reset collapse under strict Online CIL (Appendix D.2); we report their no-reset variants in the main comparison only. *Non-tree:* SLDA, SQDA, NCM, RLS. We also compare against PEC [68], a per-class exemplar-free baseline that avoids global coupling via independent class models. All methods follow the strict Online CIL protocol: single-pass, no raw data retention, sequentially expanding K[1].

Hyperparameters. Sketch capacity k, inheritance discount α, failure probability δ, grace period, and threshold generation policy are fixed globally without per-dataset tuning; defaults are k=64, α=0.6, δ=0.10, grace=200 (full configurations in Table 5). Sensitivity to k, α, and δ is in Table 9; sensitivity of MIST-K to ϵ_s and β is in Appendix F.3.

Table 1: Final mean accuracy among **tree-based methods** (mean ± std over 5 seeds). **Bold**: best within group. All methods follow the no-reset CIL-adapted protocol; default drift-reset results for EFDT and HAT are reported in Appendix D.2. ARF-NR: Adaptive Random Forest with drift detection disabled (10 trees).

Method	Synth-10	Synth-50	Wine	Iris	Covertype	Split-MNIST	Pendigits	Shuttle	Letter	HAR	Mean
MIST-G (ours)	0.994±0.004	0.999±0.002	**0.970**±0.020	**0.933**±0.030	0.559±0.013	0.863±0.003	**0.889**±0.023	0.733±0.033	0.678±0.005	0.607±0.013	0.8225
MIST-K (ours)	0.996±0.002	**1.000**±0.001	0.957±0.038	0.927±0.013	**0.694**±0.006	0.772±0.007	0.877±0.020	**0.783**±0.067	**0.692**±0.009	**0.625**±0.009	**0.8323**
VFDT	0.998±0.002	0.396±0.003	**0.970**±0.020	**0.933**±0.030	0.605±0.010	**0.874**±0.004	0.508±0.039	0.417±0.145	0.372±0.056	0.324±0.008	0.6397
Rutkowski	**1.000**±0.000	**1.000**±0.000	**0.970**±0.020	**0.933**±0.030	0.559±0.013	0.851±0.003	0.863±0.010	0.527±0.099	0.678±0.005	0.614±0.018	0.7995
EFDT (NR)	0.219±0.038	0.040±0.000	**0.970**±0.020	**0.933**±0.030	0.277±0.002	0.182±0.009	0.272±0.057	0.143±0.000	0.168±0.026	0.202±0.066	0.3406
HAT (NR)	0.423±0.049	0.084±0.017	**0.970**±0.020	**0.933**±0.030	0.262±0.032	0.324±0.037	0.614±0.037	0.220±0.063	0.237±0.032	0.307±0.054	0.4374
ARF-NR	**1.000**±0.000	**1.000**±0.000	0.952±0.010	0.927±0.025	0.458±0.013	0.782±0.009	0.846±0.009	0.444±0.029	0.568±0.012	0.601±0.010	0.7578

Table 2: Final mean accuracy among **non-tree baselines** (mean ± std over 5 seeds), included as reference. **Bold**: best within group. All methods use the same metric and protocol.

Method	Synth-10	Synth-50	Wine	Iris	Covertype	Split-MNIST	Pendigits	Shuttle	Letter	HAR	Mean
MIST-G (ours)	0.994±0.004	0.999±0.002	0.970±0.020	0.933±0.030	0.559±0.013	0.863±0.003	0.889±0.023	0.733±0.033	0.678±0.005	0.607±0.013	0.8225
MIST-K (ours)	0.996±0.002	**1.000**±0.001	0.957±0.038	0.927±0.013	**0.694**±0.006	0.772±0.007	0.877±0.020	**0.783**±0.067	0.692±0.009	0.625±0.009	0.8323
SLDA	**1.000**±0.000	**1.000**±0.000	0.979±0.021	**0.973**±0.039	0.673±0.007	0.854±0.005	0.879±0.007	0.529±0.000	0.718±0.004	**0.815**±0.001	0.8420
SQDA	**1.000**±0.000	0.999±0.002	**0.987**±0.016	0.967±0.030	0.528±0.004	**0.940**±0.003	**0.982**±0.002	0.774±0.029	**0.895**±0.006	0.791±0.005	**0.8863**
NCM	**1.000**±0.000	**1.000**±0.000	0.981±0.010	0.860±0.039	0.603±0.012	0.794±0.003	0.817±0.009	0.549±0.030	0.602±0.004	0.639±0.007	0.7845
RLS	0.999±0.002	**1.000**±0.000	0.981±0.028	0.827±0.049	0.659±0.007	0.818±0.006	0.857±0.008	0.350±0.057	0.596±0.004	**0.815**±0.001	0.7902
PEC	**1.000**±0.001	0.952±0.018	0.782±0.057	0.820±0.040	0.650±0.007	0.898±0.009	0.961±0.002	0.421±0.005	0.760±0.005	0.614±0.011	0.7857

[1]Note that these are the SOTA methods for our setting: online CIL with raw tabular features. More recent, technically heavily involved, CIL methods require pretrained backbones so they do not apply directly to our setting [26, 72, 51].

6.2 Main Results

Among tree-based methods (Table 1), MIST-G and MIST-K are the strongest single-tree configurations across the benchmark suite. EFDT (NR) collapses on most benchmarks, consistent with cold-start instability compounding under more aggressive split triggers without inheritance. ARF-NR, with 10 base trees, falls below both MIST configurations on most real-stream benchmarks, at a memory overhead that depends on the dataset regime (Table 12, Appendix G.1).

On near-Gaussian, low-K streams (Synth-10, Wine, Iris), VFDT matches MIST-G on final accuracy, reflecting an honest trade-off: its more aggressive Hoeffding criterion yields faster structural adaptation when class geometry is simple. The forgetting results (Appendix E.2) show that MIST-G consistently forgets less, so this parity comes at the cost of greater instability under class arrival. On Letter (K=26, d=16), zero splits under default settings reflect the GapTest withholding commitment: the Gini gain gap between candidate splits does not exceed the McDiarmid radius, so the root partition is not informative enough to warrant a split. Table 2 compares accuracy with non-tree baselines. SQDA has higher mean accuracy (0.8863) than MIST-K (0.8323) and MIST-G (0.8225), including stronger results on Pendigits, Letter, and Split-MNIST. SQDA also has lower mean forgetting (0.0397 versus 0.0530 for MIST-G and 0.0714 for MIST-K; Table 7) and higher throughput (Table 12). MIST-K performs well on Covertype and Shuttle, but these results do not establish a general accuracy advantage over Gaussian models. Appendices E.1–E.2 report prequential and forgetting diagnostics. Appendix K.1 adds a controlled dependence-only case where both MIST variants remain near chance while SQDA benefits from class-specific covariance.

When to prefer MIST. Prefer global Gaussian methods when class-conditionals are close to Gaussian or the signal lies mainly in feature dependence; on 19 further real streams, MIST-K beats the better of SQDA and SLDA by more than 0.02 on 4, is within ± 0.02 on 5, and trails on 10 (Appendix K.5). Prefer MIST when a single-pass, local tree is required and the marginals are non-Gaussian or multimodal enough to misspecify a global Gaussian (AngularSectors, Table 4). Neither family dominates; locating the boundary online is open.

6.3 Ablation Study

We focus on two main ablation tests. Removing inheritance causes a relative drop of 9.55% (7.86 pp absolute), showing the contribution of post-split inheritance on these datasets (Figure 3, Appendix E.1). Replacing GNB leaves with majority-vote causes a 71.97% relative collapse (59.19 pp absolute): without class-conditional likelihood estimates, the tree cannot distinguish between classes sharing the same partition cell. The McDiarmid versus Hoeffding comparison shows negligible accuracy difference ($+1.02\%$) on near-Gaussian benchmarks. The structurally meaningful difference is shown in Figure 4 (Appendix E.3): Hoeffding triggers substantially more splits as K_{local} grows, while McDiarmid maintains compact trees. GainTest yields marginal accuracy gains ($+1.20\%$) at the cost of substantially larger trees (Figure 4), confirming the McDiarmid radius as a structural regulariser. Accuracy is robust to k, α, and δ over broad ranges (Appendix F.1): performance saturates at $k \geq 32$ and $\alpha \geq 0.2$, indicating low sensitivity within the tested ranges. Setting $\alpha = 0$ collapses to 0.7439, exactly matching the *w/o Inheritance* ablation.

Table 3: Core and split-criterion ablations (average accuracy over ten standard datasets, mean over 5 seeds).

Variant	Avg Acc.	Description	Δ
MIST-G (default)	0.8225	McDiarmid + sketch + inheritance + GNB leaf	—
w/o Inheritance	0.7439	No parent-to-child transfer	**−9.55%**
w/o Sketching	0.8204	Gaussian threshold approximation	−0.26%
w/ Hoeffding	0.8309	Hoeffding split rule	+1.02%
Majority-vote leaf	0.2306	No density model	**−71.97%**
w/ SampleSplit	0.8227	Sample-based threshold generation	+0.02%
w/ DenseThreshold	0.8218	Dense threshold grid	−0.09%
GapTest (default)	0.8225	Gap between best and second-best	—
GainTest	0.8324	Best gain vs. radius	+1.20%
GainTestNoUnion	0.8405	GainTest without union-bound term	+2.19%
GaussianInheritance	0.8228	GapTest + Gaussian CDF projection	+0.04%

6.4 Stress Test on Non-Gaussian Geometry

Table 4: Stress-test results (mean ± std over 5 seeds) on non-Gaussian synthetic streams.

Method	Antipodal	Multimodal	Ring	Skewed	HeavyTail	NoisyFeature	ConceptDrift	AngularSectors	**Mean**
MIST-G	0.834±0.033	0.919±0.032	0.975±0.019	0.899±0.012	0.881±0.014	0.808±0.106	**0.993**±0.012	0.241±0.010	0.8188
MIST-K	0.829±0.021	0.983±0.005	0.968±0.026	0.978±0.009	0.897±0.022	0.768±0.121	0.988±0.011	**0.436**±0.017	**0.8559**
SLDA	0.171±0.042	0.756±0.053	0.978±0.020	0.897±0.012	**0.898**±0.025	0.811±0.104	**0.993**±0.012	0.240±0.041	0.7180
SQDA	**0.984**±0.007	**0.999**±0.001	0.975±0.021	**1.000**±0.000	0.858±0.012	0.798±0.109	**0.993**±0.012	0.235±0.015	0.8553
ARF-NR	0.862±0.025	0.947±0.018	**0.979**±0.012	0.921±0.016	0.893±0.019	**0.832**±0.091	0.991±0.009	0.255±0.021	0.8350
PEC	0.918±0.010	0.964±0.017	0.926±0.035	0.972±0.009	0.801±0.012	0.233±0.025	0.984±0.018	0.302±0.046	0.7625

Table 4 shows complementary strengths. On AngularSectors, MIST-K reaches 0.436 versus 0.235 for SQDA and 0.255 for ARF-NR. On Multimodal and Skewed, MIST-K improves over MIST-G, while SQDA remains strongest. The stress-test means of MIST-K (0.8559) and SQDA (0.8553) are close, and the SQDA lead over SLDA on Antipodal shows the value of class-specific covariance. Figure 5 provides PCA visualisations of the stress streams. These results support benefits on particular marginal geometries, rather than robustness to arbitrary non-Gaussian distributions.

For unknown marginal shape, the 25-dataset study (Appendix K.3) supports MIST-K as the default, with MIST-G useful when Gaussian marginals fit; hard likelihood selection does not beat always using K, and blending is optional.

MIST retains per-leaf, per-class summaries, with worst-case storage $O(dkL_tK_t)$; local class sparsity can reduce this cost but is not guaranteed. Appendix K.4 compares 19 streams, including regimes where MIST costs more than SQDA, and separates retained storage from peak traced allocations. Sketch-quantile initialisation costs $1.27\times$ Gaussian projection (Appendix H.2). Inheritance depends on how well parent summaries represent each child: on ShiftAtSplit, accuracy falls from 0.935 at $\alpha = 0.2$ to 0.864 at $\alpha = 1$ (Appendix K.2, Tables 20 and 22).

7 Conclusion

Streaming decision tree failure in online CIL traces to two coupled sources: a split criterion that degrades as K grows, and the absence of parent-to-child knowledge transfer. MIST addresses them with a McDiarmid radius built on a tight, K-independent sensitivity bound ($c_i{=}4/n$) that mitigates premature splits, Bayesian inheritance for post-split stabilisation, and unified KLL sketches for split evaluation and prediction. Empirically, MIST-G is competitive with global parametric methods on near-Gaussian streams, although SQDA has the higher mean accuracy. On AngularSectors, MIST-K outperforms the per-class discriminants and 10-tree ensemble. This is specific to the tested geometry; SQDA remains stronger on several other benchmarks, motivating both predictors.

Regarding limitations, both predictors use a Naive Bayes factorisation; MIST-K replaces Gaussian leaf likelihoods, not the Gaussian inheritance projection. Both can fail on dependence-only signals (Appendix K.1); a covariance-aware leaf could help but needs new theory, since per-class covariances estimated on the sparse leaves that the tree structure creates degenerate; we leave it to future work. Inheritance can introduce bias when parent and child distributions differ (Appendix K.2). Memory grows with leaves, local classes, dimension, and sketch capacity. The analysis assumes stationary class-conditionals and independent samples for each concentration argument. Reliable online predictor selection remains open, and time-uniform confidence sequences [31] are future work: the per-decision union bound gives no infinite-horizon guarantee.

Ultimately, MIST decouples local predictive modules and transfers statistically grounded knowledge at split time, giving streaming decision trees a modular route to open-world continual learning that mitigates forgetting as the tree grows.

Acknowledgments and Disclosure of Funding

We thank the Faculty of Information Technology, University of Science, Vietnam National University, Ho Chi Minh City, for providing the computing resources and research environment that supported this work. The authors declare no competing interests.

References

[1] 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 (ECCV)*, September 2018.

[2] Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[3] Rahaf Aljundi, Punarjay Chakravarty, and Tinne Tuytelaars. Expert gate: Lifelong learning with a network of experts. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7120–7129, 2017.

[4] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient based sample selection for online continual learning. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[5] Eden Belouadah and Adrian Popescu. Il2m: Class incremental learning with dual memory. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 583–592, 2019.

[6] Yael Ben-Haim and Elad Tom-Tov. A streaming parallel decision tree algorithm. *J. Mach. Learn. Res.*, 11:849–872, March 2010.

[7] Albert Bifet and Ricard Gavaldà. Learning from time-changing data with adaptive windowing. volume 7, 04 2007.

[8] Albert Bifet, Geoff Holmes, Bernhard Pfahringer, Richard Kirkby, and Ricard Gavaldà. New ensemble methods for evolving data streams. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, page 139–148, New York, NY, USA, 2009. Association for Computing Machinery.

[9] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[10] Lucas Caccia, Rahaf Aljundi, Nader Asadi, Tinne Tuytelaars, Joelle Pineau, and Eugene Belilovsky. New insights on reducing abrupt representation change in online continual learning. In *International Conference on Learning Representations (ICLR)*, 2022.

[11] Ayman Chaouki, Jesse Read, and Albert Bifet. Online learning of decision trees with Thompson sampling. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li, editors, *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 2944–2952. PMLR, 02–04 May 2024.

[12] Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision – ECCV 2018*, pages 556–572, Cham, 2018. Springer International Publishing.

[13] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. *CoRR*, abs/1812.00420, 2018.

[14] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet Kumar Dokania, Philip H. S. Torr, and Marc'Aurelio Ranzato. Continual learning with tiny episodic memories. *CoRR*, abs/1902.10486, 2019.

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

[16] Rocco De Rosa and Nicolò Cesa-Bianchi. Splitting with confidence in decision trees with application to stream mining. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2015.

[17] Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '00, page 71–80, New York, NY, USA, 2000. Association for Computing Machinery.

[18] Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999.

[19] João Gama, Pedro Medas, Gladys Castillo, and Pedro Rodrigues. *Learning with Drift Detection*, volume 8, pages 286–295. 09 2004.

[20] João Gama, Ricardo Rocha, and Pedro Medas. Accurate decision trees for mining high-speed data streams. pages 523–528, 08 2003.

[21] Pablo García-Santaclara, Bruno Fernández-Castro, and Rebeca P. Díaz-Redondo. Overcoming catastrophic forgetting in tabular data classification: A pseudorehearsal-based approach. *Engineering Applications of Artificial Intelligence*, 156:110908, 2025.

[22] Yasir Ghunaim, Adel Bibi, Kumail Alhamoud, Motasem Alfarra, Hasan Abed Al Kader Hammoud, Ameya Prabhu, Philip H.S. Torr, and Bernard Ghanem. Real-time evaluation in online continual learning: A new hope. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11888–11897, 2023.

[23] Achmad Ginanjar, Xue Li, Priyanka Singh, and Wen Hua. Continual contrastive learning on tabular data with out of distribution. In *ESANN 2025 proceedings*, ESANN 2025, page 93–98. Ciaco - i6doc.com, 2025.

[24] Heitor Murilo Gomes, Albert Bifet, Jesse Read, Jean Paul Barddal, Fabrício Enembreck, Bernhard Pfahringer, Geoff Holmes, and Talel Abdessalem. Adaptive random forests for evolving data stream classification. *Machine Learning*, 106:1–27, 10 2017.

[25] Ian Goodfellow, Mehdi Mirza, Xia Da, and Aaron Courville. An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*, 12 2013.

[26] Dipam Goswami, Yuyang Liu, Bartłomiej Twardowski, and Joost van de Weijer. Fecam: Exploiting the heterogeneity of class distributions in exemplar-free continual learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 6582–6595. Curran Associates, Inc., 2023.

[27] Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems 35 (Datasets and Benchmarks Track)*, 2022.

[28] Tyler L. Hayes, Kushal Kafle, Robik Shrestha, Manoj Acharya, and Christopher Kanan. Remind your neural network to prevent catastrophic forgetting. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 466–483, Cham, 2020. Springer International Publishing.

[29] Tyler L. Hayes and Christopher Kanan. Lifelong machine learning with deep streaming linear discriminant analysis. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 887–896, 2020.

[30] Hamed Hemati, Marco Schreyer, and Damian Borth. Continual learning for unsupervised anomaly detection in continuous auditing of financial accounting data, 2022.

[31] Steven Howard, Aaditya Ramdas, Jon McAuliffe, and Jagmohan Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49, 04 2021.

[32] Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating continual learning scenarios: A categorization and case for strong baselines, 2019.

[33] Geoff Hulten, Laurie Spencer, and Pedro Domingos. Mining time-changing data streams. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, page 97–106, New York, NY, USA, 2001. Association for Computing Machinery.

[34] David Isele and Akansel Cosgun. Selective experience replay for lifelong learning. AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[35] George John and Pat Langley. Estimating continuous distributions in bayesian classifiers. *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*, 1, 02 2013.

[36] Zohar Karnin, Kevin Lang, and Edo Liberty. Optimal quantile approximation in streams. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 71–78, 2016.

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

[38] Łukasz Korycki and Bartosz Krawczyk. Class-incremental experience replay for continual learning under concept drift. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3644–3653, 2021.

[39] Timothée Lesort, Vincenzo Lomonaco, Andrei Stoian, Davide Maltoni, David Filliat, and Natalia Díaz-Rodríguez. Continual learning for robotics: Definition, framework, learning strategies, opportunities and challenges. *Inf. Fusion*, 58(C):52–68, June 2020.

[40] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018.

[41] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6470–6479, Red Hook, NY, USA, 2017. Curran Associates Inc.

[42] Zheda Mai, Ruiwen Li, Jihwan Jeong, David Quispe, Hyunwoo Kim, and Scott Sanner. Online continual learning in image classification: An empirical survey. *Neurocomputing*, 469:28–51, 2022.

[43] Arun Mallya, Dillon Davis, and Svetlana Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV*, page 72–88, Berlin, Heidelberg, 2018. Springer-Verlag.

[44] Arun Mallya and Svetlana Lazebnik. PackNet: Adding Multiple Tasks to a Single Network by Iterative Pruning . In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7765–7773, Los Alamitos, CA, USA, June 2018. IEEE Computer Society.

[45] Chaitanya Manapragada, Geoffrey I. Webb, and Mahsa Salehi. Extremely fast decision tree. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, page 1953–1962, New York, NY, USA, 2018. Association for Computing Machinery.

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

[47] Thomas Mensink, Jakob Verbeek, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:2624–37, 11 2013.

[48] Baback Moghaddam and Gregory Shakhnarovich. Boosted dyadic kernel discriminants. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2002.

[49] German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019.

[50] Lorenzo Pellegrini, Gabriele Graffieti, Vincenzo Lomonaco, and Davide Maltoni. Latent replay for real-time continual learning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, page 10203–10209. IEEE Press, 2020.

[51] Grégoire Petit, Adrian Popescu, Hugo Schindler, David Picard, and Bertrand Delezoide. Fetril: Feature translation for exemplar-free class-incremental learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3911–3920, January 2023.

[52] Ameya Prabhu, Philip H. S. Torr, and Puneet K. Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II*, page 524–540, Berlin, Heidelberg, 2020. Springer-Verlag.

[53] Roger Ratcliff. Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions. *Psychological Review*, 97:285–308, 04 1990.

[54] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[55] David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy P. Lillicrap, and Greg Wayne. *Experience replay for continual learning*. Curran Associates Inc., Red Hook, NY, USA, 2019.

[56] Andrei Rusu, Neil Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 06 2016.

[57] Leszek Rutkowski, Lena Pietruczuk, Piotr Duda, and Maciej Jaworski. Decision trees for mining data streams based on the mcdiarmid's bound. *IEEE Transactions on Knowledge and Data Engineering*, 25(6):1272–1279, 2013.

[58] 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 Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4528–4537. PMLR, 10–15 Jul 2018.

[59] Wuxuan Shi and Mang Ye. Prototype reminiscence and augmented asymmetric knowledge aggregation for non-exemplar class-incremental learning. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1772–1781, 2023.

[60] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[61] James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. CODA-Prompt: COntinual Decomposed Attention-Based Prompting for Rehearsal-Free Continual Learning . In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11909–11919, Los Alamitos, CA, USA, June 2023. IEEE Computer Society.

[62] Gido van de Ven, Tinne Tuytelaars, and Andreas Tolias. Three types of incremental learning. *Nature Machine Intelligence*, 4:1–13, 12 2022.

[63] Gido M. van de Ven and Andreas S. Tolias. Three scenarios for continual learning. *CoRR*, abs/1904.07734, 2019.

[64] Yuandou Wang, Filip Gunnarsson, and Rihan Hai. An attention-based feature memory design for energy-efficient continual learning, 2025.

[65] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Dualprompt: Complementary prompting for rehearsal-free continual learning. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 631–648, Cham, 2022. Springer Nature Switzerland.

[66] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 139–149, 2022.

[67] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. In *International Conference on Learning Representations (ICLR)*, 2018.

[68] Michał Zając, Tinne Tuytelaars, and Gido van de Ven. Prediction error-based classification for class-incremental learning. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, editors, *International Conference on Learning Representations*, volume 2024, pages 8239–8267, 2024.

[69] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3987–3995. PMLR, 06–11 Aug 2017.

[70] Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5867–5876, 2021.

[71] Kai Zhu, Wei Zhai, Yang Cao, Jiebo Luo, and Zheng-Jun Zha. Self-sustaining representation expansion for non-exemplar class-incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9296–9305, June 2022.

[72] Huiping Zhuang, Yizhu Chen, Di Fang, Run He, Kai Tong, Hongxin Wei, Ziqian Zeng, and Cen Chen. Gacl: exemplar-free generalized analytic continual learning. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc.

[73] Huiping Zhuang, Zhenyu Weng, Hongxin Wei, Renchuzi Xie, Kar-Ann Toh, and Zhiping Lin. Acil: Analytic class-incremental learning with absolute memorization and privacy protection. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 11602–11614. Curran Associates, Inc., 2022.

Appendix Contents

A Broader Context: Online Continual Learning . . . **17**

A.1 Catastrophic forgetting and the continual-learning taxonomy . . . 17

A.2 Regularisation-based methods . . . 17

A.3 Replay and rehearsal . . . 17

A.4 Architecture-expansion and parameter isolation . . . 17

A.5 Exemplar-free deep continual learning . . . 17

A.6 Online and streaming-specific methods . . . 18

A.7 Tabular continual learning and streaming decision trees . . . 18

A.8 Positioning of MIST . . . 18

B Algorithm Pseudocode . . . **18**

C Illustrations of the Key Components . . . **18**

D Extended Experimental Setup . . . **19**

D.1 Hyperparameter Configurations . . . 19

D.2 Protocol Mismatch in Drift-Reset Trees . . . 20

D.3 Baseline Fairness and Tuning Policy . . . 21

E Main Results: Extended Analysis . . . **21**

E.1 Prequential Accuracy and Knowledge Inheritance . . . 21

E.2 Forgetting Results: Full Tables and Per-Task Curves . . . 21

E.3 Tree-Dynamics Diagnostics . . . 21

F Ablation and Sensitivity Analysis . . . **23**

F.1 Sensitivity to k, α, and δ . . . 23

F.2 Threshold-Density Ablation . . . 23

F.3 MIST-K Parameter Sensitivity . . . 24

G Memory and Computational Analysis . . . **24**

G.1 Memory Footprint, Throughput, and Trade-offs . . . 24

G.2 Local Class Cardinality on Random-Means Streams . . . 25

H Empirical Validation of the Theory . . . **25**

H.1 Empirical Validation of the $4\varepsilon_{\mathtt{sketch}}$ Bound . . . 25

H.2 Sketch-Quantile vs. Truncated-Gaussian Inheritance on Non-Gaussian Streams . . . 27

I Stress-Test Dataset Descriptions . . . **28**

J Mathematical Proofs . . . **28**

J.1 Truncated-Gaussian Moment Updates . . . 29

J.2 Proof of Theorem 5.1 (Tightness of the Gini Sensitivity) . . . 30

J.3 Proof of Theorem 5.3 (Composed Split Error) . . . 31

J.4 Proof of Corollary 5.4 (Optimization Displacement) . . . 31

J.5 Proof of Proposition 5.5 (Post-Split Variance Reduction) . . . 32

J.6 Full Derivation of Case B (Cross-Routing) . . . 32

K Additional Regime and Inheritance Experiments . . . **33**

K.1 Dependence-only signals . . . 33

K.2 When inheritance helps and hurts . . . 33

K.3 Predictor choice and optional blending . . . 35

K.4 Structural storage and measured allocation peaks . . . 35

K.5 Accuracy against global Gaussian baselines on 19 real streams . . . 37

A Broader Context: Online Continual Learning

A.1 Catastrophic forgetting and the continual-learning taxonomy

The tendency of connectionist learners to overwrite previously acquired knowledge was formalised as *catastrophic interference* by [46] and [53], and revisited for modern networks by [18] and [25]. Modern continual learning (CL) is commonly organised along three scenarios, including **task-incremental, domain-incremental, and class-incremental**, formalised by [63] and later refined in [62]; the categorisation was independently validated by [32]. Class-incremental learning (CIL) is widely regarded as the hardest setting because the task identifier is not supplied at inference and the classifier must discriminate between all classes ever seen, requiring calibrated cross-task decision boundaries. Broader surveys include [49, 15], and, for the specifically *online* (single-pass, small-batch) regime, [42]. Streaming settings aggravate forgetting relative to the batch case because each sample is seen only once and stationarity cannot be assumed, motivating drift-aware evaluation [22].

A.2 Regularisation-based methods

Regularisation approaches anchor parameters deemed important for past tasks. Elastic Weight Consolidation (EWC) uses a Fisher-information penalty [37]; Synaptic Intelligence accumulates path integrals of parameter sensitivity [69]; Memory-Aware Synapses uses gradient magnitudes of the learned function [1]; Learning without Forgetting employs knowledge distillation from the previous model as a functional prior [40]. Online variants such as Progress & Compress [58] and RWalk [12] were developed to amortise Fisher estimation over streams. These methods are effective for short task sequences but degrade sharply in unbounded class-incremental streams: importance estimates drift, the quadratic penalty cannot simultaneously protect arbitrarily many past classes, and they do not address the representation–classifier mismatch at the core of CIL.

A.3 Replay and rehearsal

Replay stores or synthesises past data to counteract forgetting. Experience Replay [55], iCaRL [54], GEM [41] and its efficient variant A-GEM [13] constrain updates with respect to a memory buffer; tiny-memory baselines are analysed in [14]. DER++ [9] augments replay with logit distillation, while MIR [2] and GSS [4] target the *online* regime with interference-aware retrieval and gradient-based sample selection. ER-ACE [10] mitigates representation drift caused by naïve cross-entropy replay. Generative replay replaces the buffer with a learned generator [60], and GDumb [52] shows that a buffer-trained model is a strong baseline. Selective replay for lifelong reinforcement learning was proposed by [34]. All of these methods **violate the exemplar-free constraint** relevant to privacy-sensitive tabular applications such as finance and healthcare, and generative replay additionally compounds forgetting via generator drift.

A.4 Architecture-expansion and parameter isolation

A complementary family grows or partitions the network per task: Progressive Networks [56], PackNet [44], Dynamically Expandable Networks [67], HAT [8], Piggyback [43], and Expert Gate [3]. These methods largely avoid forgetting by construction but require a task identifier at inference or introduce unbounded parameter growth, which is incompatible with an open-ended class stream.

A.5 Exemplar-free deep continual learning

Exemplar-free CIL methods retain only compact statistics. PASS [70] uses prototype augmentation and self-supervision; IL2M [5] rebalances logits with class statistics; SSRE [71] expands representations without exemplars; PRAKA [59] introduces prototype reminiscence and asymmetric knowledge aggregation; FeTrIL [51] translates old-class features from new-class geometry; FeCAM [26] exploits heterogeneous class covariances. Deep SLDA [29] freezes a backbone and maintains a streaming Gaussian classifier. Prompt-based methods such as L2P [66], DualPrompt [65], and CODA-Prompt [61] exploit large pre-trained vision transformers with learnable prompt pools; they depend on the quality and transferability of the frozen backbone, a resource unavailable in most tabular domains.

A.6 Online and streaming-specific methods

REMIND [28] and Latent Replay [50] compress activations to enable single-pass rehearsal; [39] surveys the online robotics setting; [22] argues for throughput-aware evaluation. In contrast to multi-epoch offline CIL, these works highlight the stringent per-sample compute budget that governs truly online learners.

A.7 Tabular continual learning and streaming decision trees

Continual learning on **tabular** streams is markedly under-explored. Recent work considers tabular CL in energy-constrained [64] and contrastive or out-of-distribution settings [23], as well as financial-audit anomaly detection [30]; a pseudo-rehearsal approach is reported in [21]. These efforts remain either replay-based or confined to narrow domains, leaving class-incremental tabular learning largely open.

The complementary data-stream-mining literature offers decision-tree learners with provable split guarantees: the Hoeffding-bound VFDT [17] and its concept-drift extension CVFDT [33]; the Extremely Fast Decision Tree (EFDT) [45]; the Hoeffding Adaptive Tree [8]; streaming ensembles [8, 24]; and change-detection mechanisms ADWIN [7] and DDM [19]. Crucially, [57] show that VFDT's use of Hoeffding's inequality is statistically inappropriate for information-gain-type split criteria and replace it with the tighter **McDiarmid bound**. However, none of these streaming-tree methods are designed or analysed for class-incremental learning: new classes invalidate cached sufficient statistics at existing splits, pure leaf counts drift as the label space grows, and classical drift detectors conflate concept drift with class emergence.

A.8 Positioning of MIST

MIST sits at the intersection of the gaps identified above. It is **exemplar-free** (no raw sample buffers or generative models, unlike §A.3), **single-pass and online** (unlike most methods in §A.2, §A.4, and §A.5), **class-set-unbounded** (unlike parameter-isolation methods in §A.4 and short-horizon regularisers in §A.2), and **designed for tabular streams** (unlike methods in §A.6–§A.7, which presuppose visual pre-training or fixed feature spaces).

Methodologically, MIST replaces the statistically loose Hoeffding split test of VFDT and EFDT with a **McDiarmid-calibrated criterion** in the spirit of [57], and extends it with **Bayesian inheritance of class-conditional leaf statistics** so that newly emerging classes borrow strength from ancestor nodes rather than restarting estimation. This yields a streaming decision-tree learner that tolerates open class sets without replay, capacity expansion, or a pre-trained backbone. A detailed analysis of the GapTest threshold and the tightness of the McDiarmid constant $c_i = 4/n$ is provided in Section 5 and Appendix J.

B Algorithm Pseudocode

Algorithm 1 in the main text gives the high-level MIST update and predict loop. We provide here the two subroutines that it invokes: Algorithm 2 details the Bayesian inheritance protocol executed at every split event, and Algorithm 3 details the sketch-based leaf likelihood used by MIST-K. Both pseudocode listings use the notation established in Section 4 of the main paper; ϕ and Φ denote the standard normal PDF and CDF respectively.

C Illustrations of the Key Components

Figure 2 shows, on data, the mechanisms of Section 4: how class medians propose a split threshold (Section 4.3), how a split changes the distribution each child inherits (Section 4.2), and why a single Gaussian can misfit a skewed class (Section 4.3).

Algorithm 2 MIST: Split with Bayesian Inheritance

Require: Leaf ℓ, split feature j^*, threshold v^*, discount $\alpha \in (0, 1]$

for each child $s \in \{\text{left}, \text{right}\}$ **do**
  for each class c **do**
    if j^* is continuous **then**
      $\zeta \leftarrow (v^* - \mu^c_{j^*})/\sigma^c_{j^*}$
      $\tilde{\Phi} \leftarrow \Phi(\zeta)$ if s=left, else $1-\Phi(\zeta)$
      $\mu^{s,c}_{j^*} \leftarrow \mu^c_{j^*} - \sigma^c_{j^*}\,\phi(\zeta)/\tilde{\Phi}$ if s=left, else $\mu^c_{j^*} + \sigma^c_{j^*}\,\phi(\zeta)/\tilde{\Phi}$ {Eq. (4)}
      if $s = \text{left}$ **then**
        $(\sigma^{s,c}_{j^*})^2 \leftarrow \sigma^{c\,2}_{j^*}\left[1 - \dfrac{\zeta\phi(\zeta)}{\tilde{\Phi}} - \left(\dfrac{\phi(\zeta)}{\tilde{\Phi}}\right)^2\right]$ {upper-truncated; Appendix J.1}
      else
        $(\sigma^{s,c}_{j^*})^2 \leftarrow \sigma^{c\,2}_{j^*}\left[1 + \dfrac{\zeta\phi(\zeta)}{\tilde{\Phi}} - \left(\dfrac{\phi(\zeta)}{\tilde{\Phi}}\right)^2\right]$ {lower-truncated; sign of linear term is $+$; Appendix J.1}
      end if
      $n^s_c \leftarrow \tilde{\Phi} \cdot \alpha \cdot n^\ell_c$
    else
      {categorical one-hot: split feature is deterministic in each child}
      $v_{\text{cat}} \leftarrow 0$ if s=left, else 1
      $\mu^{s,c}_{j^*} \leftarrow v_{\text{cat}}$; $(\sigma^{s,c}_{j^*})^2 \leftarrow 0$ {x_{j^*}=v_{cat} w.p. 1 in child s; inherit no uncertainty}
      $n^s_c \leftarrow \alpha \cdot n_c(v_{\text{cat}})$, $n_c(v_{\text{cat}})$=#$\{i : y_i$=$c,\, x_i^{(\text{attr})}$=$v_{\text{cat}}\}$
    end if
    for $k \neq j^*$ **do**
      $(\mu^{s,c}_k,\, \sigma^{s,c}_k) \leftarrow (\mu^c_k,\, \sigma^c_k)$
    end for
  end for
  Initialise KLL sketches at s as empty {MIST-G: unaffected (Gaussian accumulators inherited above). MIST-K: sketch-based feature likelihoods unavailable until new samples arrive; class-probability priors remain valid.}
end for
Remove ℓ from active leaves

D Extended Experimental Setup

D.1 Hyperparameter Configurations

Table 5 lists the fixed default hyperparameters used for every method in the main benchmark comparison. No method-specific held-out tuning was performed; the same configuration was applied uniformly across all datasets. For MIST, the defaults were chosen to reflect the theoretically motivated operating points: k=64 (sketch capacity at which the safety margin of the $4\varepsilon_{\text{sketch}}$ bound saturates; Appendix H.1), α=0.6 (well within the plateau $\alpha \geq 0.2$ identified in Table 9), and δ=0.10 (consistent with the near-flat $\sqrt{\ln(1/\delta)}$ sensitivity; Table 9).

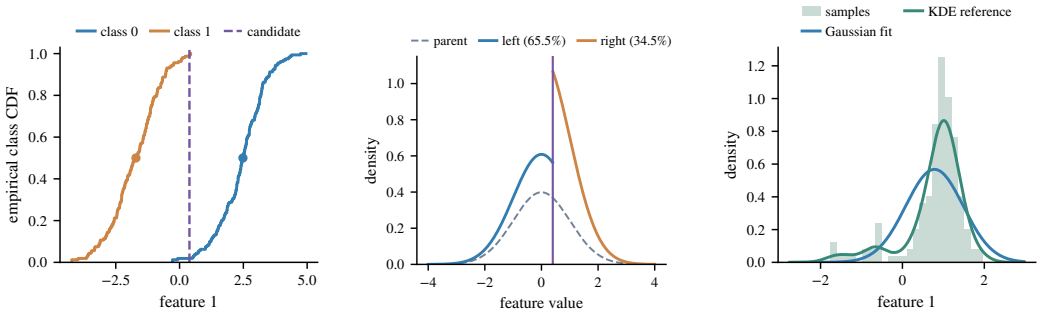
Table 5: Fixed default configurations used in the main benchmark comparison.

Method	Key hyperparameters	Policy
MIST-G / MIST-K	grace=200, δ=0.10, tie=0.05, k=64	fixed default
VFDT (Gini)	grace=200, confidence=0.95, tie=0.05	fixed default
EFDT / HAT (Gini)	grace=50, split conf=10^{-7}, tie=0.05	fixed default
Rutkowski (Gini)	grace=200, δ=0.10, tie=0.05	fixed default
SLDA / SQDA	shrinkage=10^{-4}	fixed default
NCM / RLS / PEC	implementation defaults	fixed default

Algorithm 3 MIST-K: Sketch-Based Leaf Likelihood

Require: Observation x, leaf ℓ, smoothing ϵ_s (default 1.0), bandwidth multiplier β, Dirichlet pseudo-counts $\{\alpha_c^{(\ell)}\}$ where $\alpha_c^{(\ell)} = n_c^{(\ell)} + \epsilon_s$,
$\mathrm{IQS}_{\ell,c}(x_j) = \widehat{F}_{j,c}^{-1}(\min(r + h_0, 1)) - \widehat{F}_{j,c}^{-1}(\max(r - h_0, 0))$ where $r = \widehat{F}_{j,c}(x_j)$ is the sketch-estimated rank of x_j and $h_0 = 0.25$ is the fixed half-window; returns ϵ_s if fewer than two distinct quantiles exist

for each class c **do**
  $\log \hat{p}(c) \leftarrow \log \alpha_c^{(\ell)} - \log \sum_{c'} \alpha_{c'}^{(\ell)}$ {Dirichlet prior}
  for each feature j **do**
    if j is categorical **then**
      $\hat{p}_j \leftarrow (n_c^{(\ell)}(j{=}x_j) + \epsilon_s) \;/\; (n_c^{(\ell)} + 2\epsilon_s)$ {Bernoulli likelihood from Dirichlet pseudo-counts; $n_c^{(\ell)}(j{=}x_j)$: count of class-c samples with x_j matching the observed category}
      $\log \hat{p}(x_j \mid c) \leftarrow \log \hat{p}_j$
    else
      $h_j \leftarrow \beta \cdot \mathrm{IQS}_{\ell,c}(x_j)$ {local inter-quantile spacing}
      $\hat{F}^+ \leftarrow \widehat{F}_{j,c}(x_j + h_j)$; $\hat{F}^- \leftarrow \widehat{F}_{j,c}(x_j - h_j)$
      $\log \hat{p}(x_j \mid c) \leftarrow \log(\hat{F}^+ - \hat{F}^- + \epsilon_s) - \log(2h_j + \epsilon_s)$
    end if
  end for
  $\log \hat{p}(c \mid x) \leftarrow \log \hat{p}(c) + \sum_j \log \hat{p}(x_j \mid c)$
end for
return $\arg\max_c \log \hat{p}(c \mid x)$



(a) Candidate threshold (b) Inherited distributions (c) Leaf likelihood

Figure 2: Illustrations of the key components. (a) Exact empirical class CDFs of feature 1 on Synth-10 (seed 42, classes 0 and 1); the midpoint of the class medians is the candidate threshold, and CDF queries score the mass on each side. (b) A Gaussian parent split at 0.4 and the conditional densities that truncated-Gaussian projection initialises for the two children (analytical; percentages are each child's share of the parent mass). (c) Class-0 samples of the Skewed stream (seed 42, feature 1) with a moment-matched Gaussian (MIST-G) and a raw-sample KDE as a visual reference for MIST-K. Panels (a) and (c) use raw data for illustration only; neither curve is a measured KLL-sketch output.

D.2 Protocol Mismatch in Drift-Reset Trees

Table 6 reports a focused diagnostic comparing EFDT and HAT under their default drift-triggered reset configuration against the no-reset variant used throughout the main experiments. Under strict Online CIL, class arrivals are permanent structural events; drift detectors treat them as transient shifts and discard the associated leaf statistics upon reset, causing near-random performance.

When resets are disabled, HAT recovers substantial accuracy on several datasets (e.g. Pendigits: $0.250 \rightarrow 0.614$, Split-MNIST: $0.170 \rightarrow 0.324$), confirming that its failure under the default configuration is a consequence of protocol misalignment. EFDT (no-reset) shows modest recovery on some

datasets (e.g. Covertype: $0.167 \rightarrow 0.277$, Pendigits: $0.196 \rightarrow 0.272$) but remains below competitive accuracy, consistent with cold-start instability without inheritance.

Table 6: Default vs. no-reset diagnostics on selected datasets (final mean accuracy, mean over seeds).

Method	Pendigits	Shuttle	Letter	HAR	Covertype	Split-MNIST
EFDT+ADWIN (reset)	0.196	0.253	0.121	0.167	0.167	0.167
EFDT (no-reset)	0.272	0.143	0.168	0.202	0.277	0.182
HAT (default)	0.250	0.207	0.107	0.231	0.168	0.170
HAT (no-reset)	0.614	0.220	0.237	0.307	0.262	0.324

D.3 Baseline Fairness and Tuning Policy

All main benchmark results (Tables 1 and 7) are produced without method-specific held-out tuning; the policy in Table 5 is applied uniformly to all methods. For non-tree baselines, we use published implementation defaults without further adjustment. ARF-NR uses 10 base trees with drift detection disabled, matching the no-reset convention of the tree-based group.

E Main Results: Extended Analysis

E.1 Prequential Accuracy and Knowledge Inheritance

Figure 3 measures what happens when a node splits. We hook the split routine and evaluate the held-out test examples of all classes seen so far immediately before and after each actual structural split, using the same examples on both sides (at most 3,000, fixed across the run). MIST-G and MIST-K change by at most 0.6 points across the six streams, whereas without inheritance accuracy falls from 84–100% to 0–29%: the new children carry no class statistics until they see data. Only seeds in which the tree split enter each mean (counts are in the panel titles). This is a diagnostic run with the default configuration and the standard, unweighted Gini criterion; it is not part of the main benchmark.

E.2 Forgetting Results: Full Tables and Per-Task Curves

Table 7 reports average forgetting across all standard benchmarks. EFDT and HAT (both NR) show elevated forgetting on Shuttle (0.400) and HAR (0.314), consistent with Figure 4: these methods trigger more splits, creating smaller and less stable leaf predictors whose class estimates degrade faster under new class arrivals. The negative forgetting entry for MIST-G on Shuttle (-0.035) reflects a genuine accuracy improvement on Task-0 as the GNB accumulator refines its per-class likelihood estimates over the stream; no split occurs on Shuttle, so this is not attributable to structural repartitioning.

Per-task accuracy decay curves for Synth-10, Covertype, and Split-MNIST are provided in the supplementary materials. Across all three datasets, MIST-G with inheritance retains earlier tasks substantially better than the no-inheritance ablation: per-task accuracy curves decay more slowly and stabilise at higher plateaus, with the gap proportional to n_{parent} at the triggering split event, consistent with Proposition 5.5.

E.3 Tree-Dynamics Diagnostics

Figure 4 plots the evolution of four structural quantities over the training stream: number of leaves, cumulative splits, weighted leaf Gini impurity, and mean local class cardinality K_{local}. The comparison between the McDiarmid GapTest (default) and the Hoeffding-based variant reveals the structural regularisation effect described in Section 4.1: the Hoeffding criterion triggers substantially more splits as K_{local} grows, yet Table 3 shows only marginal aggregate accuracy gain ($+1.02\%$); the structurally meaningful difference lies in tree compactness (Figure 4), not raw accuracy. The lower-right panel shows reduced local class cardinality in these runs. This can reduce storage relative to the worst-case bound in Appendix G.1; it does not imply that total allocated memory decreases after each split.

Table 8 compares the default truncated-Gaussian projection (Eq. (4)) against a sketch-quantile projection (SKETCHINHERIT) that initialises child means directly from parent KLL quantiles. Both

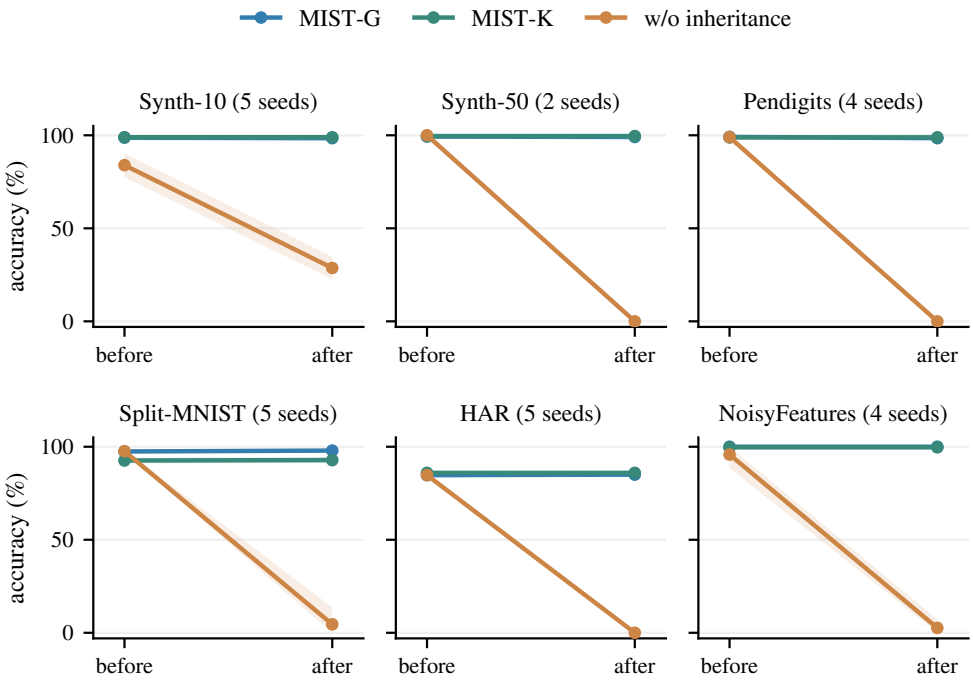


Figure 3: Held-out accuracy on all classes seen so far immediately before and after each structural split (mean ± std over the seeds in which the tree split). With inheritance the split leaves accuracy essentially unchanged; without it, accuracy collapses to 0–29%.

Table 7: Average forgetting (mean ± std over 5 seeds) on the standard Online CIL benchmarks (lower is better). **Bold**: best mean within group.

Method	Synth-10	Synth-50	Wine	Iris	Covertype	Split-MNIST	Pendigits	Shuttle	Letter	HAR	**Mean**
Tree-based (CIL-adapted)											
MIST-G (ours)	0.001±0.003	0.001±0.002	0.038±0.039	0.080±0.045	0.143±0.017	0.051±0.003	0.059±0.016	−0.035±0.040	0.089±0.005	0.103±0.006	**0.0530**
MIST-K (ours)	0.001±0.001	0.000±0.000	0.055±0.045	0.100±0.035	0.200±0.027	0.064±0.004	0.067±0.019	0.000±0.007	0.066±0.003	0.161±0.007	0.0714
VFDT	0.001±0.003	0.750±0.001	0.038±0.035	0.080±0.040	0.340±0.013	0.066±0.004	0.581±0.059	0.486±0.140	0.612±0.072	0.600±0.084	0.3554
Rutkowski	0.000±0.000	0.000±0.000	0.038±0.035	0.080±0.040	0.143±0.015	0.060±0.002	0.066±0.008	0.367±0.099	0.089±0.004	0.102±0.006	**0.0945**
EFDT (NR)	0.024±0.050	0.250±0.092	0.038±0.035	0.080±0.040	0.490±0.096	0.044±0.049	0.720±0.083	0.151±0.099	0.300±0.187	0.804±0.122	0.2901
HAT (NR)	−0.035±0.106	0.227±0.105	0.038±0.035	0.080±0.040	0.740±0.128	−0.038±0.105	0.423±0.035	0.443±0.113	0.094±0.059	0.516±0.051	0.2488
Non-tree baselines (reference)											
SLDA	0.000±0.000	0.000±0.000	0.014±0.020	0.020±0.027	0.202±0.009	0.074±0.005	0.084±0.009	0.210±0.041	0.106±0.007	0.014±0.002	0.0724
SQDA	0.000±0.000	0.002±0.003	0.000±0.000	0.050±0.050	0.226±0.002	0.026±0.005	0.011±0.004	0.020±0.010	0.041±0.004	0.021±0.002	**0.0397**
NCM	0.000±0.000	0.000±0.000	0.007±0.016	0.150±0.061	0.273±0.011	0.084±0.004	0.073±0.011	0.079±0.033	0.103±0.004	0.143±0.007	0.0912
RLS	−0.000±0.004	0.000±0.000	0.029±0.047	0.180±0.057	0.219±0.007	0.109±0.006	0.066±0.012	0.428±0.025	0.147±0.009	0.013±0.001	0.1191
PEC	0.001±0.001	0.017±0.012	0.152±0.067	0.090±0.049	0.231±0.007	0.042±0.005	0.026±0.003	0.004±0.008	0.086±0.004	0.099±0.041	0.0748

variants produce identical accuracy and forgetting on Split-MNIST, confirming that the post-split cold-start benefit derives from the warm-start mechanism itself—i.e., providing statistically grounded priors to children—rather than from the specific formula used to project those priors. The sketch-quantile variant is $1.27\times$ slower on average; the overhead is not amortised over the stream since splits are infrequent. The large per-seed variance in runtime (Gaussian: ± 27.4 s; Sketch-Quantile: ± 17.5 s) reflects differences in tree depth reached across seeds rather than algorithmic instability. The default truncated-Gaussian projection is therefore preferred on near-Gaussian benchmarks where both variants are equivalent in accuracy.

Table 8: Split-time inheritance ablation on Split-MNIST (5 seeds). The non-parametric initialisation matches accuracy but is $1.27\times$ slower on average.

Variant	Accuracy	Forgetting	Runtime (s)	Relative time
Gaussian projection (default)	0.7717 ± 0.007	0.0739 ± 0.005	69.35 ± 27.39	$1.00\times$
Sketch-quantile projection	0.7717 ± 0.007	0.0739 ± 0.005	88.35 ± 17.48	$1.27\times$

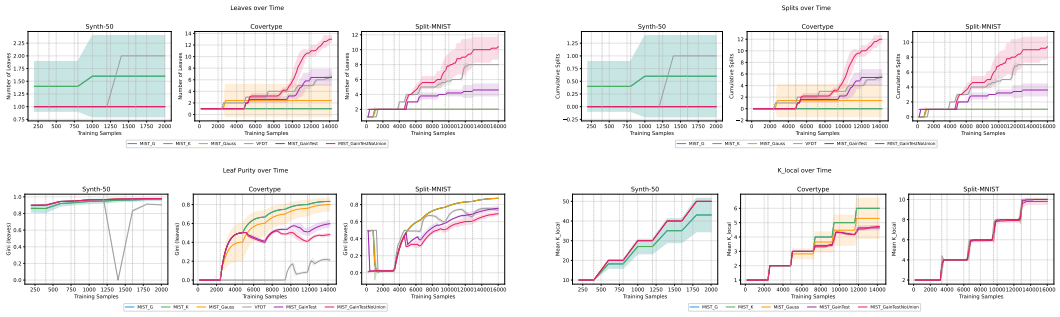


Figure 4: Tree-dynamics diagnostics on Synth-50, Covertype, and Split-MNIST (standard, unweighted Gini criterion for all models). **Top:** number of leaves and cumulative splits over the stream. **Bottom:** weighted leaf Gini impurity and mean local class cardinality K_{local}. Removing the union bound over attributes (GAINTESTNOUNION) causes systematic over-splitting—up to $6\times$ more leaves than MIST—without a commensurate gain in leaf purity (bottom-left), confirming that the union bound is essential for controlling split proliferation as K_{local} grows. VFDT (Hoeffding bound, no inheritance) shows a comparable split rate to GAINTEST but higher residual impurity, reflecting the absence of knowledge inheritance. The staircase pattern in K_{local} (bottom-right) marks task boundaries; MIST's tight McDiarmid radius delays new splits until sufficient evidence accumulates across all K_{local} classes.

F Ablation and Sensitivity Analysis

F.1 Sensitivity to k, α, and δ

Table 9 reports accuracy averaged over Synth-10, Iris, Covertype, and Split-MNIST as each hyperparameter is swept in isolation with all others held at default. Two plateaus are clearly visible. For sketch capacity, performance saturates at $k \geq 32$, with negligible practical variation for $k > 64$; this is consistent with the $4/k$ decay of the sketch error bound (Section H.1) entering the negligible regime. For inheritance discount, the transition is sharp: α=0.0 collapses to 0.7439, matching the *w/o Inheritance* ablation exactly (Table 3), while any $\alpha \geq 0.2$ recovers near-full performance. For failure probability δ, the near-constant accuracy across nearly two orders of magnitude confirms the theoretically predicted $\sqrt{\ln(1/\delta)}$ plateau in the operational radius (Eq. (3)): the logarithmic dependence makes δ a low-leverage parameter in practice.

Table 9: Sensitivity of MIST-G to sketch capacity k, inheritance discount α, and split-confidence level δ (average ten standard datasets). The near-identical values across δ reflect the theoretically predicted $\sqrt{\ln(1/\delta)}$ plateau and are a feature, not a reporting artifact (confirmed experimentally across $\delta \in [0.01, 0.30]$).

k	16	32	64	128	256	512
Avg Acc.	0.8219	0.8221	0.8225	0.8225	0.8260	0.8252
α	0.0	0.2	0.4	0.6	0.8	1.0
Avg Acc.	0.7439	0.8226	0.8226	0.8226	0.8226	0.8226
δ	0.01	0.05	0.10	0.15	0.20	0.30
Avg Acc.	0.8226	0.8226	0.8225	0.8226	0.8226	0.8226

F.2 Threshold-Density Ablation

MIST generates split-threshold candidates from inter-class quantile midpoints, yielding at most $K_{\text{local}} - 1$ candidates per feature (sparse policy). Table 10 evaluates denser grids of 10 and 20 candidates. Table 10 evaluates denser grids of 10 and 20 candidates. The observed non-monotone runtime (dense grids are modestly faster on average) therefore cannot be attributed to a reduced per-candidate cost; it more likely reflects reduced tree-growth overhead when a larger candidate set occasionally identifies a stronger split earlier, slightly shortening the overall training trajectory. Runtime variance increases substantially with grid density, however (std up to 28 s on Covertype vs.

<2 s for sparse), making this benefit unreliable. Crucially, accuracy does not change at any density level across both datasets. Crucially, accuracy does not change at any density level across both datasets. This confirms that (i) the sparse quantile-midpoint policy already identifies the same best split as a much denser grid, because KLL quantiles concentrate around class-conditional modes; and (ii) the union bound over candidate thresholds in Eq. (3) does not conservatively expand the radius enough to suppress valid splits. The sparse policy therefore remains preferable for its stability (std <2 s vs. up to 28 s for dense grids on Covertype).

Table 10: Threshold-density ablation (mean runtime; 5 seeds). Accuracy is unchanged at all densities. Dense grids show lower mean runtime but substantially higher variance, confirming the sparse policy as the stable, efficient default.

Dataset	Sparse (1)	Dense (10)	Dense (20)	Accuracy	Acc. change
Covertype	65.40 ± 1.26 s	53.19 ± 19.12 s (-19%)	53.09 ± 28.40 s (-19%)	0.694 ± 0.006	none
Split-MNIST	126.10 ± 18.00 s	102.24 ± 9.54 s (-19%)	99.93 ± 1.63 s (-21%)	0.772 ± 0.007	none

F.3 MIST-K Parameter Sensitivity

Table 11 sweeps the two MIST-K-specific hyperparameters: smoothing constant ϵ_s and bandwidth multiplier β. Across all ten benchmarks, accuracy is nearly invariant to ϵ_s: every value in $[10^{-5}, 10^{-1}]$ yields the same average of 0.8322. Note that ϵ_s appears in Eq. (5) for *all* features, including numerical ones; however, on purely numerical benchmarks the sketch-CDF differences $\hat{F}^+ - \hat{F}^-$ dominate and the additive ϵ_s contribution is negligible, making accuracy insensitive to its value. For nominal (categorical) features, ϵ_s acts as Laplace smoothing; since all standard benchmarks are purely numerical, this branch is inactive and the invariance to ϵ_s is expected. The bandwidth multiplier β does influence accuracy: performance peaks at $\beta \in [1.0, 2.0]$ (avg. acc. 0.8336–0.8338) and degrades at both extremes—under-smoothing at β=0.25 (0.7963, -4.5%) collapses adjacent class-conditional modes, and over-smoothing at β=4.0 (0.8141, -2.3%) blurs interclass boundaries. We recommend β=1.0 as the safe default; ϵ_s is irrelevant for numerical-only data and can be set to any value in the standard range.

Table 11: Sensitivity of MIST-K average accuracy (over 10 datasets, 5 seeds) to smoothing constant ϵ_s and bandwidth multiplier β. ϵ_s has no effect because all benchmarks are purely numerical (KLL sketch-CDF is used for likelihood; Laplace smoothing is inactive).

ϵ_s	10^{-5}	10^{-4}	10^{-3}	10^{-2}	10^{-1}
Avg Acc.	0.8322	0.8322	0.8322	0.8322	0.8322
β	0.25	0.50	1.00	2.00	4.00
Avg Acc.	0.7963	0.8162	0.8336	0.8338	0.8141

G Memory and Computational Analysis

G.1 Memory Footprint, Throughput, and Trade-offs

Let L be the number of leaves, d the number of features, and $K_{\text{local}}^{(\ell)}$ the number of active classes at leaf ℓ. With sketch capacity k, total sketch memory is:

$$M_{\text{sketch}} = O\left(dk \sum_{\ell=1}^{L} K_{\text{local}}^{(\ell)}\right),$$

which can be smaller than the worst-case $O(dkLK)$ when $K_{\text{local}}^{(\ell)} \ll K$ once leaves become purer (Appendix G.2).

Structural storage versus allocation peaks. Before splitting, the root can represent all classes, costing $O(dkK_t)$. A split can increase total storage; it does not guarantee that the sum of local class counts decreases. Appendix K.4 reports retained statistics and Python-traced allocation peaks on

Table 12: Memory footprint (KB) and throughput (samples/s) on two representative datasets: Covertype ($K = 6$, $d = 54$, low-K regime) and Split-MNIST ($K = 10$, $d = 50$ after PCA, higher-K regime). **Pareto-optimal** configurations on Covertype are marked with $\star$: NCM (fastest/lightest), SLDA (fast with high accuracy), and MIST-K (highest accuracy). EFDT (D) and HAT (D) — where (D) denotes the default drift-reset configuration — are included for completeness; their near-random accuracy reflects protocol misalignment (Appendix D.2), not resource efficiency. Memory is the mean over 3 seeds, except the MIST rows (§), which use the common structural estimator and the five seeds of Table 25 (Appendix K.4); the original MIST entries in this table could not be reproduced and were replaced. Throughput is from the original run for all rows.

Method	Memory (KB)		Throughput (samp./s)		Note
	Covertype	Split-MNIST	Covertype	Split-MNIST	
MIST-G§	587.0	1171.5	674	401	
MIST-K§ $\star$	587.0	1171.5	43	37	highest acc
VFDT	305.5	591.1	1179	682	
ARF-NR	3044.5	5912.3	1104	634	
EFDT (D)	4335.4	–	11	–	near-random
HAT (D)	703.4	–	244	–	near-random
SLDA $\star$	188.5	243.8	2543	2123	fast+accurate
SQDA	416.8	596.1	1560	1161	
NCM $\star$	5.8	8.8	46778	31815	ultra-fast

Letter and HAR. These measurements do not support a universal factor-two peak-memory bound. Table 27 isolates the effect of dimension: raising d from 20 to 100 grows MIST-K storage by about $5\times$ and SQDA storage by $24.3\times$.

Throughput and Pareto frontier. Table 12 characterises memory and throughput on two representative datasets that bracket the practical operating range. Among Pareto-optimal configurations on Covertype, NCM provides the fastest and lightest option, SLDA provides a fast and accurate parametric baseline, and MIST-K achieves the highest single-method accuracy at the cost of greater memory and lower throughput (due to per-prediction quantile lookups in the sketch). MIST-G offers a middle ground: sketch memory identical to MIST-K but significantly higher throughput (674 vs 43 samples/s on Covertype), since split evaluation uses sketches but prediction uses the faster Gaussian accumulators. EFDT (D) and HAT (D) are included for completeness; their near-random accuracy reflects protocol misalignment (Appendix D.2) and their throughput numbers are not meaningful in this context.

G.2 Local Class Cardinality on Random-Means Streams

To examine local class sparsity empirically, we evaluate MIST-G on multi-dimensional random-means streams (d=10, σ=0.3, $K \in \{10 \ldots 150\}$, 2 seeds). Class means are drawn uniformly from $[0, 5]^{10}$, so splits distribute classes across multiple leaves naturally without artificial pre-specification of the partition structure.

Table 13 shows that K_{local} falls 64–82% below K_{global} across all tested cardinalities (18–36% of K_{global} on streams whose trees grow to 17.5–33 leaves), consistent with the lower-right panel of Figure 4 on real benchmarks. At K=150, fewer than 30 classes remain active per leaf on average despite the global set containing 150 classes, showing that local class sparsity reduces the storage estimate relative to the $O(dkLK)$ worst case on these random-means streams.

H Empirical Validation of the Theory

H.1 Empirical Validation of the $4\varepsilon_{\text{sketch}}$ Bound

We measured $|\hat{F} - F|$ (sketched vs. exact Gini gain) at every split decision under the strict Online CIL protocol across 7 datasets, 4 sketch capacities ($k \in \{16, 32, 64, 128\}$), and 5 seeds. All results use the corrected CIL class-incremental presentation and standard (unweighted) Gini gain, consistent with Theorem 5.1.

Table 13: K_{local} mean vs. K_{global} on random-means streams (d=10, σ=0.3, 2 seeds). The reported local class counts are below the global count in these runs.

K	Leaves	K_{local} mean	Reduction
25	20.0	9.0	64%
50	17.5	13.7	73%
75	33.0	15.5	79%
100	20.5	29.2	71%
150	21.0	27.1	82%

Per-dataset results at k=64. Table 14 reports empirical sketch error across all validated datasets at k=64, giving $4\varepsilon = 4/64 = 0.0625$. The bound holds at every recorded split event. Safety margins range from 2.3× (Pendigits, the most marginal case) to 27.9× (HAR, where class-conditional distributions are approximately isotropic and the sketch is highly accurate). Covertype exceeds the KLL rank-error threshold ($\varepsilon_{\text{sketch}} = 1/k$) as a CDF *value* error, but this does not violate the bound: Theorem 5.3 controls Gini gain error via rank error, not CDF value error directly, and Gini err max $= 0.015 \ll 0.0625$ confirms the bound holds.

Table 14: Empirical sketch error at k=64 ($4\varepsilon = 0.0625$). *Events*: split decisions recorded (5 seeds, ×5). *Ratio*: 4ε / Gini err max (worst-case safety margin). Letter is excluded; see note below.

Dataset	Events	CDF err mean	CDF err max	Gini err mean	Gini err max	4ε	Ratio
Pareto	20	0.0031	0.0067	0.00498	0.01126	0.0625	5.5×
LogNormal	20	0.0080	0.0145	0.01056	0.02955	0.0625	2.1×
Exponential	20	0.0047	0.0101	0.00523	0.01226	0.0625	5.1×
Outlier 5%	20	0.0068	0.0122	0.00914	0.02318	0.0625	2.7×
Pendigits	4	0.0073	0.0145	0.01331	0.02715	0.0625	2.3×
HAR	5	0.0009	0.0014	0.00114	0.00224	0.0625	27.9×
Shuttle	–	0.0001	0.0001	–	–	0.0625	–
Covertype[‡]	12	0.0248	0.0585	0.00487	0.01486	0.0625	4.2×

[‡] CDF err max $= 0.059$ on Covertype exceeds $1/k = 0.0156$ as a CDF *value* error, but this does not violate the theoretical bound: $4\varepsilon_{\text{sketch}}$ in Theorem 5.3 bounds Gini gain error via *rank* error, not CDF value error. Gini err max $= 0.015 \ll 0.0625$ confirms the bound holds.

Note on zero-split datasets. Letter produced no split events at any k under the default `grace_period`= 200, consistent with the main experiment (Table 17). Shuttle produced no split events under the unweighted-Gini validation protocol: its heavy class imbalance (≈80% majority class) causes the Gini gain gap to fall below the McDiarmid radius, so the bound is vacuously satisfied.

Scaling with sketch capacity k. Table 15 reports safety margins across sketch capacities for the four synthetic validation datasets, where sufficient split events (20 per configuration) are available at all k values. The bound $4\varepsilon_{\text{sketch}} \leq 4/k$ holds at every recorded split event across all configurations. Margins degrade gracefully as k decreases, consistent with the $O(1/k)$ rank-error guarantee of KLL: even at k=16 (the smallest tested), the minimum margin remains above 2×. At k=64, worst-case margins range from 2.1× (LogNormal) to 5.5× (Pareto), confirming the bound holds with meaningful slack across all tested configurations.

Table 15: Ratio 4ε / Gini err max across k values, synthetic datasets only (20 split events each). The $4\varepsilon_{\text{sketch}}$ bound holds at every recorded event; safety margin degrades gracefully as k decreases but remains ≥ 2.1× at k=64.

k	4ε	Pareto	LogNormal	Exponential	Outlier 5%
16	0.250	14×	20×	20×	9×
32	0.125	14×	12×	25×	14×
64	0.0625	5.5×	2.1×	5.1×	2.7×
128	0.0312	13×	8×	16×	10×

H.2 Sketch-Quantile vs. Truncated-Gaussian Inheritance on Non-Gaussian Streams

This appendix evaluates the robustness of the truncated-Gaussian projection used in Bayesian inheritance (Section 4.2) when the underlying class-conditional distributions violate the Gaussianity assumption. We compare MIST-G against SKETCHINHERIT, a variant that initialises child means directly from parent KLL quantiles: the left-child mean for class c on the split feature is set to $\widehat{F}_{j,c}^{-1}(r_L/2)$, and the right-child mean to $\widehat{F}_{j,c}^{-1}(r_L + r_R/2)$, where r_L and r_R are the estimated mass fractions routing to each child. Quantiles are resolved by nearest-point lookup with boundary clipping to $[0.01, 0.99]$. Non-split feature parameters and class mass are inherited identically to MIST-G. All other components are identical, including α=0.6. Results are averaged over 5 seeds $\{42, 123, 456, 789, 3330\}$ with `grace_period`= 200, $\delta = 0.10$, $k = 64$.

Synthetic non-Gaussian streams. Four distributions isolate specific failure modes of the truncated-Gaussian projection. **Pareto**: power-law tails (α=1.5) with `log1p` normalisation; tests whether log-transform approximately Gaussianises the truncated region. **LogNormal**: log-normal marginals ($\sigma \in [0.8, 1.5]$); right-skewed distributions where $\mathbb{E}[X \mid X \leq t]$ lies below the Gaussian prediction. **Outlier 5%**: Gaussian base with 5% extreme outliers ($\times 50\sigma$); tests robustness of Welford variance estimates under contamination. **Exponential**: exponential marginals ($\lambda \in [0.5, 2]$); moderately right-skewed, between LogNormal and Gaussian in severity. Each dataset contains K=6 classes, d=10 features, n=12,000 training samples (2,000 per class), presented in strict Online CIL order.

Table 16 shows that gains are largest where the Gaussian truncation assumption fails most severely: +33.5 pp on Outlier 5% (a single outlier corrupts the Welford variance estimate, making the moment projection unreliable, while the sketch CDF estimate remains robust) and +2.3 pp on LogNormal. The Pareto case (−1.2 pp) is the sole exception: `log1p` normalisation approximately symmetrises the truncated region, making the Gaussian approximation adequate while the sketch introduces marginal quantile-grid variance.

Table 16: Accuracy comparison on synthetic non-Gaussian streams (mean ± std over 5 seeds). Δacc = SKETCHINHERIT − MIST-G.

Dataset	Distribution	MIST-G	SKETCHINHERIT	Δacc	Splits
Pareto	Power-law (α=1.5), `log1p` transform	0.9626 ± 0.003	0.9503 ± 0.004	-0.012	5.0
LogNormal	Log-normal ($\sigma \in [0.8, 1.5]$)	0.8049 ± 0.017	0.8277 ± 0.011	$+0.023$	5.0
Outlier 5%	Gaussian +5% outliers ($\times 50\sigma$)	0.3577 ± 0.026	0.6928 ± 0.039	$\mathbf{+0.335}$	5.0
Exponential	Exponential ($\lambda \in [0.5, 2]$)	0.8676 ± 0.025	0.8809 ± 0.031	$+0.013$	4.4

Real-world benchmark datasets. Table 17 confirms that on near-Gaussian data (Pendigits, Split-MNIST, Covertype, HAR), both variants produce identical accuracy: the truncated-Gaussian projection is harmless when valid. Shuttle produced no split events under either variant (Table 17), so the bound is vacuously satisfied and no accuracy comparison is possible. Letter produces no splits under the default settings and is included for completeness.

SKETCHINHERIT is 1.27× slower on average (Table 8, Relative time column), confirming that the cold-start benefit derives from the warm-start mechanism rather than the projection formula. The default MIST-G uses truncated-Gaussian projection on near-Gaussian benchmarks where it is equivalent at lower computational cost.

Table 17: Accuracy comparison on standard Online CIL benchmarks (mean ± std over 5 seeds). K: number of classes; d: feature dimension.

Dataset	K	d	MIST-G	SKETCHINHERIT	Δacc	Splits
Pendigits	10	16	0.8886 ± 0.023	0.8886 ± 0.023	0.000	1.4
Shuttle	7	9	0.7331 ± 0.033	0.7331 ± 0.054	0.000	0.0
Letter	26	16	0.6775 ± 0.006	0.6775 ± 0.006	0.000	0.0
HAR	6	561	0.6074 ± 0.013	0.6074 ± 0.013	0.000	1.0
Covertype	7	54	0.5590 ± 0.013	0.5590 ± 0.013	0.000	0.0
Split-MNIST	10	50	0.8629 ± 0.003	0.8629 ± 0.002	0.000	1.0

I Stress-Test Dataset Descriptions

All stress-test datasets are evaluated under the same Online CIL protocol as the main experiments (Section 6.1) with a fixed random seed; generation code is provided in the supplementary materials. Figure 5 shows 3D PCA projections of all eight distributions.

Multimodal (K=8, d=10, 4 tasks). Each class is drawn from a mixture of M=3 well-separated Gaussian modes. Tests whether leaf predictors can model non-unimodal class-conditional distributions. MIST-K's sketch-based marginal estimator captures the multimodal structure without raw-data retention, while GNB-leaf methods incur systematic bias.

Skewed (K=8, d=10, 4 tasks). Each class has a dominant mode (90%) and a spatially separated minority mode (10%). The skew places the class-conditional mean outside the dominant mode's mass, exposing the truncated-Gaussian inheritance assumption to bias.

AngularSectors (K=4, d=8, 2 tasks). Samples lie on a 2D ring; class identity is determined by angular sector. All K classes share mean $\approx \mathbf{0}$ and identical covariance, so QDA, LDA, and Naive Bayes degrade to chance ($1/K$). A tree with axis-aligned splits on the two signal features can progressively narrow sectors. MIST-K achieves the highest reported accuracy on this stream (Table 4); this result does not imply robustness on every non-Gaussian geometry.

Antipodal (K=8, d=10, 4 tasks). Each class consists of two diametrically opposite clusters on a hypersphere. SLDA fails (0.171) because a shared covariance hyperplane cannot separate antipodal classes; SQDA succeeds because per-class Gaussian means differ.

Ring (K=6, d=8, 3 tasks). Classes are distributed on concentric hypersphere shells. The radial direction is near-Gaussian, so most methods perform competitively. This confirms that MIST's advantage is regime-dependent: where the Gaussian assumption is adequate, global parametric methods remain competitive.

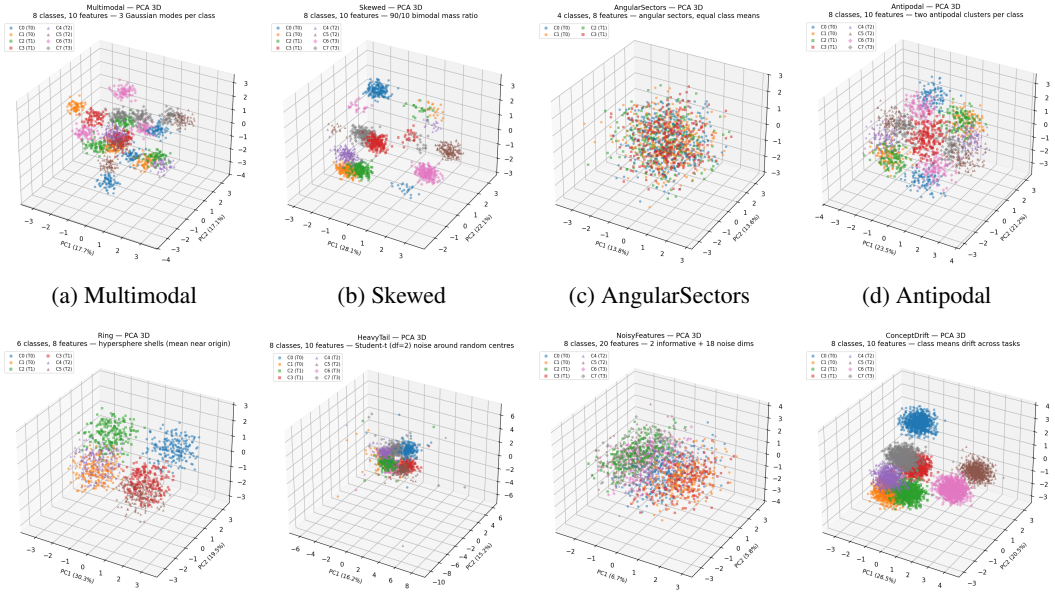
HeavyTail (K=8, d=10, 4 tasks). Class-conditional distributions follow Student-t (ν=2) noise around random centers. Fat tails inflate Welford variance estimates, biasing MIST-G's truncated-Gaussian projection; MIST-K mitigates this via the empirical sketch-based CDF.

NoisyFeatures (K=8, d=20, 4 tasks). Only d_{info}=2 dimensions carry signal; the remaining 18 are independent noise. Tests split-selection robustness under high irrelevant- feature contamination. The conservative McDiarmid GapTest is particularly relevant: uninformative features produce near-zero Gini gain, and a tight confidence radius suppresses spurious splits on noise features.

ConceptDrift (K=8, d=10, 4 tasks). Class-conditional means shift linearly across tasks (drift $= 1.5$ per task along feature 0). Included to verify that MIST's conservative split criterion does not over-commit to stale boundaries. High accuracy across all methods (> 0.988) confirms that gradual drift is absorbed by the existing split mechanism without explicit drift detection.

J Mathematical Proofs

Table 18 summarises the assumptions underlying each theoretical result. The bounded-difference property is deterministic; applying the stated McDiarmid inequality additionally requires independent samples. Identical distributions are not required by that inequality. This sample-independence assumption is distinct from the Naive Bayes feature factorisation used for leaf prediction, which does not enter the Gini sensitivity calculation. Adaptive routing and repeated split testing require care; no general temporal-dependence or time-uniform guarantee is claimed here.



(a) Multimodal (b) Skewed (c) AngularSectors (d) Antipodal

(e) Ring (f) HeavyTail (g) NoisyFeature (h) ConceptDrift

Figure 5: 3D PCA visualisations of the eight stress-test streams (seed 42; all generated samples shown, standardised and projected onto the top three principal components; colour = class, marker shape = task). Each panel illustrates the geometric property that motivates the dataset: non-unimodal class structure (Multimodal, Skewed), manifold geometry (AngularSectors, Ring), cluster arrangement (Antipodal), and distributional properties (HeavyTail, NoisyFeature, ConceptDrift).

J.1 Truncated-Gaussian Moment Updates

When $X \sim \mathcal{N}(\mu, \sigma^2)$ is conditioned on $X \leq v$ (left child), let $\zeta = (v - \mu)/\sigma$. The exact conditional moments are:

$$\mathbb{E}[X \mid X \leq v] = \mu - \sigma \frac{\phi(\zeta)}{\Phi(\zeta)}, \tag{10}$$

$$\mathrm{Var}(X \mid X \leq v) = \sigma^2 \left[1 - \frac{\zeta \phi(\zeta)}{\Phi(\zeta)} - \left(\frac{\phi(\zeta)}{\Phi(\zeta)} \right)^2 \right]. \tag{11}$$

For the right child ($X \geq v$), replacing $\Phi(\zeta)$ with $1 - \Phi(\zeta)$ throughout gives:

$$\mathbb{E}[X \mid X \geq v] = \mu + \sigma \frac{\phi(\zeta)}{1 - \Phi(\zeta)}, \tag{12}$$

$$\mathrm{Var}(X \mid X \geq v) = \sigma^2 \left[1 + \frac{\zeta \phi(\zeta)}{1 - \Phi(\zeta)} - \left(\frac{\phi(\zeta)}{1 - \Phi(\zeta)} \right)^2 \right]. \tag{13}$$

Critical sign difference. The linear term $\zeta\phi(\zeta)/\tilde{\Phi}$ carries a *negative* sign for the left child (Eq. (11)) and a *positive* sign for the right child (Eq. (13)). Algorithm 2 Lines 7–10 apply this conditional sign. Using the same sign for both children would yield the incorrect formula $[1 - \zeta\phi(\zeta)/(1-\Phi(\zeta)) - \cdots]$ for the right child, which can produce negative variances: at $\zeta = 0$ both versions of the bracket equal $1 - 2/\pi \approx 0.36 > 0$, but at $\zeta = 1$ the wrong-sign formula gives ≈ -2.85 while the correct right-child bracket is ≈ 0.20.

Both conditional variances satisfy $\mathrm{Var}(X \mid X \gtrless v) \leq \sigma^2$, confirming that the inheritance projection cannot inflate feature-parameter uncertainty. For categorical features, no moment projection is applied; see Section 4.2 for the empirical co-occurrence formula.

Table 18: Assumptions per theoretical result.

Result	Assumption	Note
Theorem 5.1	Bounded differences per leaf	The sensitivity statement is deterministic. Its use in McDiarmid concentration additionally requires independent samples
Theorem 5.3 (statistical term)	Bounded differences; independent samples within a leaf	Concentrates F around $\mathbb{E}[F]$ over the empirical leaf distribution. Independence, not identical distribution, is assumed, as in the per-leaf protocol of [17]
Theorem 5.3 (sketch term)	KLL rank guarantee	Holds w.p. $\geq 1 - \delta_{\text{sk}}$
Corollary 5.4	Gaussian $X_j \mid c$; finite condition number κ	Gaussianity only converts rank error into threshold displacement; holds w.p. $\geq 1 - \delta - \delta_{\text{sk}}$
Proposition 5.5	(i) Dirichlet prior; (ii) NIG prior with $\alpha n_c^{\text{parent}} > 2$	Standard conjugate models; the variance in (ii) is finite only above this inherited count

J.2 Proof of Theorem 5.1 (Tightness of the Gini Sensitivity)

We first state two supporting lemmas and then give the three-step proof.

Lemma J.1 (Gini gain is Lipschitz in threshold)**.** *For a fixed sample set S, the Gini split gain $F(S;\, v)$ as a function of the split threshold v is Lipschitz with constant $L_v = O(1/\sigma_{\min})$, where $\sigma_{\min}$ is the minimum class-conditional standard deviation at the leaf.*

Proof. A threshold displacement Δv reroutes at most $\Delta_{\text{total}} = n\hat{p}(v)|\Delta v|$ samples in total, where $\hat{p}(v)$ is the empirical density at v. By Lemma J.2, $|F(S') - F(S)| \leq (4/n) \cdot \Delta_{\text{total}} = 4\hat{p}(v)|\Delta v|$. Under a Gaussian class-conditional model, $\hat{p}(v) \leq 1/(\sigma_{\min}\sqrt{2\pi})$, giving $L_v = 4\hat{p}(v) = O(1/\sigma_{\min})$. □

Lemma J.2 (Gini gain Lipschitz in routing counts)**.** *For any perturbation of routing counts of total magnitude $\Delta_{\text{total}} = \sum_c |\Delta n_{c,L}|$, the Gini split gain satisfies $|F(S') - F(S)| \leq (4/n) \cdot \Delta_{\text{total}}$.*

Proof. By Theorem 5.1, a single-sample replacement (which changes routing counts by at most 1 in total) alters F by at most $4/n$. For a total perturbation of magnitude Δ_{total}, applying the triangle inequality over at most Δ_{total} unit steps gives $|F(S') - F(S)| \leq (4/n) \cdot \Delta_{\text{total}}$. □

Step 1: Sensitivity of node impurity. For any sample set U of size m, replacing one sample changes at most two class proportions, each by $1/m$. Since $G(U) = 1 - \sum_c p_c^2$, and at most two terms change with $p_c + p_c' \leq 2$, we have $|G(U) - G(U^{(i)})| \leq 2/m$.

Lemma J.3 (Node-impurity sensitivity is tight)**.** *The bound $|G(U) - G(U^{(i)})| \leq 2/m$ is tight: for any $\gamma > 0$, there exists U of size m and a replacement such that $|G(U) - G(U^{(i)})| > 2/m - \gamma$.*

Proof. Consider $K = 2$ with U containing $n_0 = 1$ sample of class 0 and $n_1 = m - 1$ samples of class 1. Replace the single class-0 sample with class 1. Direct computation gives:

$$|G(U) - G(U^{(i)})| = \frac{2(m-1)}{m^2} = \frac{2}{m} - \frac{2}{m^2} \to \frac{2}{m} \quad \text{as } m \to \infty.$$

For any $\gamma > 0$, choose $m > 2/\gamma$. □

Step 2: Case decomposition. *Case A (same routing):* both z_i and z_i' route to child L. Then $\Delta F = \Delta G_P - (n_L/n)\Delta G_L$, and by Step 1, $|\Delta F| \leq 2/n + (n_L/n)(2/n_L) = 4/n$. *Case B (cross*

routing): the replacement changes routing between children. Direct expansion gives $n|\Delta F| \leq 2 < 4$ (full derivation in Appendix J.6). Hence $|F(S) - F(S^{(i)})| \leq 4/n$ in all cases.

Step 3: Tightness. Consider binary labels with $m = \lfloor \sqrt{n/2} \rfloor$ left-child samples (all class 1) and right-child samples (all class 0); flip one left sample to class 0. The scaled deviation $H(m, n) = 2(n - m)(2m - 1)/(nm) \to 4$ as $n \to \infty$. For any $\gamma > 0$, choosing n large enough yields $|F(S) - F(S^{(i)})| > 4/n - \gamma$.

Steps 1–2 establish $c_i \leq 4/n$; Step 3 confirms $4/n$ is sharp via Lemma J.3, completing the proof. □

J.3 Proof of Theorem 5.3 (Composed Split Error)

By the triangle inequality, $|\widehat{F} - F^*| \leq |F(S) - F^*| + |\widehat{F} - F(S)|$.

Statistical term. From Theorem 5.1, $c_i = 4/n$ for all i. Applying McDiarmid's inequality with a union bound over d features and m thresholds:

$$\Pr\left[\max_{j,v} |F_{j,v}(S) - F^*_{j,v}| \geq \eta\right] \leq 2dm \exp\left(-\frac{n\eta^2}{8}\right).$$

Setting the right-hand side to δ and solving gives $\eta = \sqrt{8 \ln(2dm/\delta)/n}$.

Sketch term. On the KLL success event (probability $\geq 1 - \delta_{\text{sketch}}$), rank error $\varepsilon_{\text{sketch}}$ implies $|\hat{n}_{c,L} - n_{c,L}| \leq n_c \varepsilon_{\text{sketch}}$, so $\sum_c |\hat{n}_{c,L} - n_{c,L}| \leq n\varepsilon_{\text{sketch}}$. By Lemma J.2, $|\widehat{F} - F(S)| \leq (4/n) \cdot n\varepsilon_{\text{sketch}} = 4\varepsilon_{\text{sketch}}$.

Combining via union bound gives confidence $\geq 1 - \delta - \delta_{\text{sketch}}$. □

J.4 Proof of Corollary 5.4 (Optimization Displacement)

The displacement bound follows a three-term decomposition. Throughout, we work under the KLL success event (probability $\geq 1 - \delta_{\text{sketch}}$) on which the rank-error guarantee $\varepsilon_{\text{sketch}}$ holds.

Step (i): Quantile displacement. Under Gaussianity of $X_j \mid c$, a KLL rank error of $\varepsilon_{\text{sketch}}$ implies that the empirical median $\hat{q}_{0.5,c}$ differs from the true median $q_{0.5,c}$ by at most $\varepsilon_{\text{sketch}}$ in probability mass. By the Mean Value Theorem applied to the class-conditional CDF Φ_c, the spatial displacement satisfies

$$|\hat{q}_{0.5,c} - q_{0.5,c}| = \frac{\varepsilon_{\text{sketch}}}{p_c(\tilde{q})},$$

for some $\tilde{q}$ between $q_{0.5,c}$ and $\hat{q}_{0.5,c}$, where p_c is the class-conditional density. An upper bound on this displacement therefore requires a *lower* bound on $p_c(\tilde{q})$ over the displaced interval.

Small-displacement regime. We work in the regime $|\hat{q}_{0.5,c} - q_{0.5,c}| \ll \sigma_c$, i.e. the displacement is small relative to the class spread. In this regime the density p_c is approximately constant over the interval $[\min(q, \hat{q}), \max(q, \hat{q})]$ at its value at the median, which for a Gaussian satisfies $p_c(q_{0.5,c}) = 1/(\sigma_c\sqrt{2\pi})$. Hence $p_{\min} \approx 1/(\sigma_c\sqrt{2\pi})$, giving

$$|\hat{q}_{0.5,c} - q_{0.5,c}| \lesssim \varepsilon_{\text{sketch}} \cdot \sigma_c\sqrt{2\pi}.$$

This bound is an asymptotic statement as $\varepsilon_{\text{sketch}} \to 0$: for small rank errors the approximation is tight; for large $|\zeta|$ where the tail density drops significantly below the mode density, the bound becomes conservative. The practical impact is negligible, as confirmed by the $\leq 4\times$ empirical safety margins in Table 15 [36].

Step (ii): Threshold displacement. Taking $\sigma_{\max} = \max_c \sigma_c$ and the result of Step (i):

$$|\hat{v}^* - v^*| \leq \varepsilon_{\text{sketch}} \cdot \sigma_{\max}\sqrt{2\pi}.$$

Step (iii): Gini loss from displacement. By Lemma J.1, the empirical Gini split gain $F(S; \cdot)$ is Lipschitz in the threshold with constant $L_v = O(1/\sigma_{\min})$. With condition number $\kappa = \sigma_{\max}/\sigma_{\min}$:

$$F(S;\, v^*) - F(S;\, \hat{v}^*) \leq L_v \cdot |\hat{v}^* - v^*| \leq 4\kappa\, \varepsilon_{\text{sketch}}.$$

This bounds the *empirical* Gini loss incurred by substituting the sketch-derived threshold $\hat{v}^*$ for the exact optimum v^* on the same sample set S.

Combining into Eq. (7)**.** Equation (7) controls the total deviation of the sketch-evaluated gain at $\hat{v}^*$ from the population-optimal gain $F^*(v^*)$. By the triangle inequality:

$$\left|F(S;\,\hat{v}^*) - F^*(v^*)\right| \;\le\; \underbrace{\left|F(S;\,v^*) - F^*(v^*)\right|}_{\text{statistical}} + \underbrace{\left[F(S;\,v^*) - F(S;\,\hat{v}^*)\right]}_{\text{displacement}}.$$

The statistical term is bounded by $\sqrt{8\ln(2dm/\delta)/n} + 4\varepsilon_{\text{sketch}}$ via Theorem 5.3 (holding with probability $\ge 1-\delta-\delta_{\text{sketch}}$). The displacement term is bounded by $4\kappa\,\varepsilon_{\text{sketch}}$ from Step (iii). Summing gives:

$$\left|F(S;\,\hat{v}^*) - F^*(v^*)\right| \;\le\; \sqrt{\frac{8\ln(2dm/\delta)}{n}} + 4(1+\kappa)\,\varepsilon_{\text{sketch}},$$

which is Eq. (7). In particular, since $F(S;v^*) - F(S;\hat{v}^*) \le 4\kappa\varepsilon_{\text{sketch}}$ (Step iii) and $4\kappa\varepsilon_{\text{sketch}} \le$ the RHS of Eq. (7), the stated bound also covers the empirical displacement loss directly. □

J.5 Proof of Proposition 5.5 (Post-Split Variance Reduction)

Part (i): Class probabilities. The inherited prior is Dirichlet with total mass $S_0 = \alpha n_{\text{parent}}$. After t local samples, $S_t = S_0 + t$. The Dirichlet posterior variance:

$$\text{Var}(\theta_c \mid \mathcal{D}_t) = \frac{\beta_c^{(t)}(S_t - \beta_c^{(t)})}{S_t^2(S_t+1)} \le \frac{1}{4(S_t+1)} \le \frac{1}{4(S_0+1)},$$

where the first inequality uses the AM-GM bound $\beta_c^{(t)}(S_t - \beta_c^{(t)}) \le S_t^2/4$.

Part (ii): Feature parameters. Under the NIG conjugate model with $n_0 = \alpha n_c^{\text{parent}}$ pseudo-observations, the marginal posterior on μ_j^c at child creation is a Student-t distribution with $\nu = n_0$ degrees of freedom and scale $(\sigma_j^c)^2/n_0$. Its variance exists when $n_0 > 2$ and equals:

$$\frac{\nu}{\nu-2}\cdot\frac{(\sigma_j^c)^2}{n_0} = \frac{(\sigma_j^c)^2}{n_0-2} = \frac{(\sigma_j^c)^2}{\alpha n_c^{\text{parent}} - 2}.$$

For the split feature, substituting $\sigma^2_{\text{trunc}} \le (\sigma_j^c)^2$ yields the stated bound. □

J.6 Full Derivation of Case B (Cross-Routing)

We provide the exact derivation for Case B of Theorem 5.1, where replacing sample z_i with z_i' changes the child to which it is routed. The result is strictly tighter than the Case A bound.

Lemma J.4 (Case B Bound: Exact)**.** *Under Case B (cross-routing): $z_i \in L$ has label 1, replacement $z_i' \in R$ has label 0. Then $|F(S) - F(S^{(i)})| \le 2/n < 4/n$ exactly, for all finite n.*

Proof. Let $m := n_{1,L}$ and $k := n_{1,R}$ be non-negative integers. After the replacement: L has $n_L - 1$ samples with $m-1$ class-1, and R has n_R+1 samples with k class-1. Let $p = (m+k)/n$ be the class-1 proportion before the swap.

We compute $n\cdot\Delta F$ by expanding $F = G_P - \frac{n_L}{n}G_L - \frac{n_R}{n}G_R$ into three terms.

Term (1): parent Gini change. Using $G = 2p(1-p)$:

$$n\cdot\Delta G_P = n\big[2\big(p-\tfrac{1}{n}\big)\big(1-p+\tfrac{1}{n}\big) - 2p(1-p)\big] = 2(2p-1) - \frac{2}{n}.$$

Term (2): weighted left-child Gini change. Using $G_L = 2\frac{m}{n_L}\frac{n_L-m}{n_L}$:

$$n\cdot\Delta\big(\tfrac{n_L}{n}G_L\big) = 2(n_L-m)\left[\frac{m-1}{n_L-1} - \frac{m}{n_L}\right] = \frac{-2(n_L-m)^2}{n_L(n_L-1)}.$$

Term (3): weighted right-child Gini change. Using $G_R = 2\frac{k}{n_R}\frac{n_R-k}{n_R}$:

$$n\cdot\Delta\big(\tfrac{n_R}{n}G_R\big) = \frac{2k^2}{n_R(n_R+1)}.$$

Exact formula:

$$n \cdot \Delta F = 2(2p-1) - \frac{2}{n} + \frac{2(n_L-m)^2}{n_L(n_L-1)} - \frac{2k^2}{n_R(n_R+1)}. \tag{14}$$

Bounding $|n \cdot \Delta F| < 2$. Write $T := \frac{n \cdot \Delta F}{2} = M(m) + K(k)$, where:

$$M(m) := \frac{2m - n_L - 1}{n} + \frac{(n_L-m)^2}{n_L(n_L-1)}, \tag{15}$$

$$K(k) := \frac{2k - n_R}{n} - \frac{k^2}{n_R(n_R+1)}. \tag{16}$$

$K(k)$ is a downward-opening quadratic in $k \in \{0, \ldots, n_R\}$ with maximum $K(k^*) = -n_R(n_L - 1)/n^2 \leq 0$ and boundary values satisfying $K(k) > -1$. Hence $-1 < K(k) \leq 0$.

$M(m)$ is an upward-opening quadratic in $m \in \{1, \ldots, n_L\}$ with minimum $M(m^*) = n_R(n_L - 1)/n^2 \geq 0$ and boundary values satisfying $M(m) < 1$. Hence $0 \leq M(m) < 1$.

Therefore $-1 < K(k) + M(m) < 1$, giving $|T| < 1$ and $|n \cdot \Delta F| < 2 < 4$. □

Remark J.5 (Extension to $K > 2$)**.** *In Case B with $K > 2$, the replacement $z_i \to z_i'$ moves one sample from child L to child R (or vice versa), changing exactly two class counts: $n_{y_i,L}$ decreases by 1 and $n_{y_i',R}$ increases by 1, while n_L and n_R change by ∓ 1. The Gini split gain depends on child counts only through $\sum_c p_{c,s}^2 = \sum_c (n_{c,s}/n_s)^2$. Perturbing one class count $n_{c_0,L}$ by -1 and n_L by -1:*

$$\Delta\left(\sum_c \frac{n_{c,L}^2}{n_L^2}\right) = \frac{(n_{c_0,L}-1)^2 + \sum_{c \neq c_0} n_{c,L}^2}{(n_L-1)^2} - \frac{\sum_c n_{c,L}^2}{n_L^2}.$$

Direct computation shows $|n \cdot \Delta F| < 2$ holds for all non-negative integer counts by the same bounding argument as the $K = 2$ case: the numerator change is bounded by $2n_L - 1$ and the combined effect satisfies $|T| = |M(m) + K(k)| < 1$. The binary case ($K = 2$) is thus the binding extremal configuration, confirming that $4/n$ is tight for all K.

K Additional Regime and Inheritance Experiments

This section reports saved experiments from the review period separately from the original benchmark tables. Results use five seeds unless stated otherwise; error bars denote standard deviation across seeds. These supplementary diagnostics are reported separately because their protocols and configurations differ from the main benchmark.

K.1 Dependence-only signals

CovOnly uses four zero-mean Gaussian classes, each with two independent two-dimensional signal blocks. Each block has unit marginal variances and correlation $+0.9$ or -0.9, with signs determined by the class bits. Twelve independent standard-normal nuisance dimensions give $d = 16$. CovOnly8 uses three blocks (eight classes) and ten nuisance dimensions, again with $d = 16$. Thus all univariate class marginals coincide, although the joint distributions differ. CovPlusMean restores a first-order signal by adding a class-specific mean shift of 2.5 along one signal coordinate to the four-class construction. CovOnly and CovPlusMean use 700 examples per class; CovOnly8 uses 350. The saved diagnostic standardises the generated pool before a stratified 80/20 split; this transductive scaling is a protocol limitation and does not constitute a deployment-style preprocessing evaluation. Classes arrive in two, four, and two tasks, respectively. The five seeds are 42, 123, 456, 789, and 3330. Both MIST variants remain near chance on the dependence-only streams, whereas SQDA benefits from class-specific covariance. The univariate split search also lacks a population gain signal for its first split. Recovery after adding marginal mean information bounds the interpretation: flexibility of marginal density estimates does not recover arbitrary feature interactions.

K.2 When inheritance helps and hurts

The Clean control contains four Gaussian classes with 750 samples per class. Their cluster centres are $(-8, -1.5)$, $(-8, 1.5)$, $(8, -1.5)$, and $(8, 1.5)$, with standard deviation 0.7 and six Gaussian nuisance

Table 19: Final mean accuracy on covariance-only signals and a mean-shift control. Bayes reference is Monte Carlo accuracy using the known generating model; chance is 0.25, 0.125, and 0.25 respectively.

Method	CovOnly	CovOnly8	CovPlusMean
SQDA	0.717 ± 0.020	0.594 ± 0.028	0.984 ± 0.005
SLDA	0.259 ± 0.019	0.150 ± 0.012	0.911 ± 0.008
VFDT	0.426 ± 0.007	0.218 ± 0.006	0.500 ± 0.000
MIST-G	0.259 ± 0.020	0.142 ± 0.011	0.909 ± 0.007
MIST-K	0.245 ± 0.016	0.130 ± 0.014	0.856 ± 0.009
Batch GNB (offline)	0.259	0.142	0.909
Bayes reference	0.733	0.629	0.983

features. ShiftAtSplit contains two regions centered at $x_0 = \pm 6$, 2,000 examples per region, and two classes with majority proportion 0.85, reversed between regions. The class-to-x_1 mapping also reverses between regions, with means ± 2 and standard deviation 0.6; four nuisance features are added. Both use a 20% test split and class-incremental arrival. In this saved diagnostic, standardisation was fitted before the train/test split; results should therefore be read as a within-protocol mechanistic comparison rather than an independent deployment benchmark. The grace period is 200. Inheritance

Table 20: Accuracy versus inheritance discount, mean and standard deviation over five seeds.

Stream	$\alpha = 0$	0.2	0.6	1.0
Clean	0.544 ± 0.101	0.980 ± 0.004	0.979 ± 0.005	0.981 ± 0.004
ShiftAtSplit	0.573 ± 0.147	0.935 ± 0.039	0.885 ± 0.021	0.864 ± 0.014

improves the Clean control, but its benefit is not monotone in inherited mass under split-induced mismatch. In ShiftAtSplit, increasing α beyond 0.2 reduces accuracy even though all listed inherited variants remain better than no inheritance. This is a bias–variance trade-off, not evidence that the prior is always beneficial.

On the same four-cluster geometry, a separate four-task diagnostic adds independent $\mathcal{N}(0, \tau^2)$ noise to each inherited feature mean in standardised coordinates. The grace period is 200, with the same pool-standardisation caveat as above; the no-inheritance control differs from the preceding experiment. Table 21 reports selected cells of the saved discount-by-noise grid. Large τ can produce substantial seed variability; no setting is uniformly best. Table 22 isolates the cold start that inheritance

Table 21: Mean corruption diagnostic: accuracy over five seeds. The no-inheritance control is 0.641 for every noise level by construction.

τ	$\alpha = 0.1$	0.6	1.0
0.0	0.977 ± 0.004	0.977 ± 0.005	0.979 ± 0.004
0.25	0.959 ± 0.017	0.923 ± 0.095	0.928 ± 0.096
0.5	0.856 ± 0.089	0.914 ± 0.093	0.927 ± 0.096
1.0	0.741 ± 0.004	0.865 ± 0.104	0.916 ± 0.093
1.5	0.741 ± 0.004	0.760 ± 0.119	0.842 ± 0.105
3.0	0.740 ± 0.003	0.713 ± 0.149	0.768 ± 0.098
8.0	0.740 ± 0.003	0.616 ± 0.103	0.698 ± 0.040

targets. After every structural split we record prequential (test-then-train) accuracy on the next $h \in \{10, 50, 100\}$ training samples, averaged over the splits of a run and then over seeds with at least one split (all five on Clean; four on ShiftAtSplit, where the tree splits at most once). On Clean, any $\alpha > 0$ removes most of the deficit (at h=50: 0.835 at α=0, 0.989 at α=0.6). On ShiftAtSplit the sign reverses after the first few samples: at h=50 and h=100 the no-inheritance control beats every inherited variant (0.960 versus 0.860, and 0.980 versus 0.872), even though every inherited variant still wins on final accuracy (Table 20). The three $\alpha > 0$ columns coincide on this stream. These are training-stream accuracies over few windows, so they complement the held-out results above.

Table 22: Prequential accuracy on the next h training samples after a split (mean $\pm$ std over seeds with a split; five seeds on Clean, four on ShiftAtSplit). α=0 is no inheritance.

Stream	h	$\alpha = 0$	0.2	0.6	1.0
Clean	10	0.785 ± 0.116	0.970 ± 0.040	0.977 ± 0.029	0.980 ± 0.027
Clean	50	0.835 ± 0.105	0.974 ± 0.014	0.989 ± 0.008	0.988 ± 0.010
Clean	100	0.845 ± 0.107	0.980 ± 0.013	0.992 ± 0.006	0.985 ± 0.011
ShiftAtSplit	10	0.800 ± 0.000	0.825 ± 0.083	0.825 ± 0.083	0.825 ± 0.083
ShiftAtSplit	50	0.960 ± 0.000	0.860 ± 0.049	0.860 ± 0.049	0.860 ± 0.049
ShiftAtSplit	100	0.980 ± 0.000	0.872 ± 0.008	0.872 ± 0.008	0.872 ± 0.008

K.3 Predictor choice and optional blending

We compare both fixed predictors with LL-hard and LL-soft on 25 datasets: the 18 original standard/stress datasets and seven additional real streams (electricity, MagicTelescope, phoneme, segment, wall-robot, JapaneseVowels, and wine_quality). At each class-feature model in a leaf, the optional extension scores Gaussian and sketch densities on retained decile quantiles. LL-hard chooses the higher likelihood; LL-soft combines their densities with softmax likelihood weights. This reuses existing summaries, requires no raw data replay, and adds prediction computation. It is a separate diagnostic, not a component of the original benchmark methods. Always-K has a smaller average

Table 23: Signed accuracy difference from the better fixed predictor on each dataset (five seeds). Positive values beat this dataset-level reference; they are not negative regret in the usual convention.

Strategy	Mean difference	Worst difference
Always G	-0.0379	-0.1983
Always K	-0.0090	-0.0911
LL-hard	-0.0106	-0.0911
LL-soft	$+0.0021$	-0.0111

deficit than always-G in this study, supporting it as a default when marginal shape is unknown. LL-hard does not improve on always-K, so these results do not establish a reliable model selection rule. LL-soft can exceed a dataset-level fixed-model reference because it blends models locally. Its small average improvement and reduced worst-case deficit support considering blending as an optional extension.

K.4 Structural storage and measured allocation peaks

Structural storage counts retained model statistics; it is distinct from temporary allocations and operating-system resident memory. Table 25 uses the shared structural estimator in the saved real-stream experiments. MB denotes 10^6 bytes. The table reports 19 real streams, including cases where MIST costs more than SQDA. The results do not show a monotone crossover in dimension: local class counts, tree size, and sketch capacity also matter. For costly high-dimensional covariance models, the archived measurement includes a one-seed, early-stopped allocation-convergence check for SQDA and SLDA on HAR and five-seed fit-only runs for these models on gas-drift; these do not imply completed full-stream accuracy evaluation. Accordingly, the storage table is not combined with main-table accuracy to declare a Pareto frontier.

Table 26 reports Python-traced allocation peaks during the saved training/evaluation run. These peaks include temporary allocations tracked by `tracemalloc`; they are not OS RSS and do not measure all native-library or pre-existing dataset memory. They can substantially exceed the retained structural state. No universal "less than twice final memory" claim is supported.

Table 27 isolates dimension on random-means streams (class means uniform in $[0, 5]^d$; Gaussian, Student-t, three-mode and lognormal noise; σ=1, 60 examples per class, two tasks, five seeds, the same estimator as Table 25). Raising d from 20 to 100 grows MIST-K storage by about $5\times$ and SQDA storage by $24.3\times$. MIST-K kept a single leaf, except in the heavy-tailed K=50, d=100 cell (1.2 leaves, hence its larger ratio), so the table does not show local class sparsity. Eight of the 80 planned cells (K=50, d=100: skewed for all methods, multimodal for SQDA, SLDA, VFDT) did not complete and are omitted; the K=10 rows are identical across noise families.

Table 24: Complete predictor-choice diagnostic, mean accuracy over five seeds. All four strategies are evaluated within the same experiment.

Dataset	G	K	LL-hard	LL-soft
Wine	0.970	0.957	0.962	0.975
Iris	0.933	0.927	0.927	0.933
Synth-10	0.994	0.996	0.996	0.996
Skewed	0.899	0.976	0.974	0.969
Multimodal	0.919	0.981	0.979	0.988
HeavyTail	0.881	0.895	0.891	0.933
AngularSectors	0.241	0.439	0.430	0.439
Antipodal	0.834	0.822	0.836	0.853
Pendigits	0.889	0.877	0.880	0.890
Split-MNIST	0.863	0.772	0.772	0.862
Covertype	0.559	0.694	0.696	0.699
Letter	0.678	0.692	0.687	0.689
HAR	0.607	0.625	0.624	0.614
Shuttle	0.733	0.783	0.783	0.782
electricity	0.692	0.721	0.726	0.729
MagicTelescope	0.649	0.688	0.691	0.696
phoneme	0.748	0.769	0.769	0.775
segment	0.798	0.896	0.891	0.888
wall-robot	0.660	0.844	0.844	0.849
JapaneseVowels	0.848	0.824	0.820	0.854
wine_quality	0.249	0.253	0.243	0.246
Synth-50	0.999	1.000	1.000	0.998
Ring	0.975	0.962	0.952	0.972
NoisyFeatures	0.808	0.764	0.762	0.802
ConceptDrift	0.993	0.984	0.966	0.990

Table 25: Structural MB from a common estimator. MIST denotes MIST-K. Most cells average five seeds; high-dimensional covariance-model cells include fit-only or allocation-convergence measurements (see text), and are not accuracy results.

Dataset	d	K	MIST	SQDA	SLDA	MIST/SQDA
phoneme	5	2	0.022	0.001	0.001	16.18
electricity	7	2	0.032	0.003	0.002	12.31
Shuttle	9	7	0.068	0.013	0.006	5.44
MagicTelescope	10	2	0.038	0.005	0.004	7.31
wine_quality	11	7	0.076	0.022	0.010	3.51
JapaneseVowels	14	9	0.150	0.044	0.019	3.38
Pendigits	16	10	0.296	0.064	0.027	4.61
Letter	16	26	0.330	0.135	0.053	2.45
segment	19	7	0.225	0.063	0.028	3.58
eye_movements	24	3	0.144	0.043	0.024	3.37
wall-robot	24	4	0.128	0.057	0.029	2.24
pol	48	11	0.588	0.617	0.248	0.95
MiniBooNE	50	2	0.240	0.122	0.082	1.97
Split-MNIST	50	10	1.172	0.608	0.248	1.93
jannis	54	4	0.389	0.283	0.143	1.37
Covertype	54	6	0.587	0.425	0.192	1.38
optdigits	64	10	0.755	0.993	0.404	0.76
gas-drift	128	6	1.021	2.372	1.061	0.43
HAR	561	6	7.353	37.811	17.669	0.19

Table 26: Mean retained structural MB and peak traced allocation MB. The seed count is shown for each cell; the two columns measure different quantities.

Method	Dataset	Seeds	Structural final	Traced peak
MIST_G	Letter	5	0.330	4.091
MIST_G	HAR	5	7.353	124.602
MIST_K	Letter	5	0.330	4.093
MIST_K	HAR	5	7.353	124.603
VFDT	Letter	5	0.065	0.804
VFDT	HAR	5	0.202	15.271
SLDA	Letter	5	0.053	0.671
SLDA	HAR	5	17.669	41.235

Table 27: Structural MB at $d = 20$ and $d = 100$ (ratio in parentheses) on random-means streams, five seeds. Standard deviations are zero except for MIST-K on the heavy-tailed K=50 cell at d=100 (± 0.47 MB across seeds, from the leaf count).

K	Noise	MIST-K		SQDA	
		d=20	d=100	d=20	d=100
10	all four	0.108	0.538 (4.97×)	0.099	2.416 (24.3×)
50	Gaussian	0.477	2.387 (5.01×)	0.496	12.080 (24.3×)
50	heavy-tailed	0.477	2.621 (5.50×)	0.496	12.080 (24.3×)

K.5 Accuracy against global Gaussian baselines on 19 real streams

Table 28 compares MIST-K with the better of SQDA and SLDA on the 19 real streams of Table 25: six OpenML streams from the numerical classification benchmark of Grinsztajn et al. [27] (electricity, MagicTelescope, pol, jannis, eye_movements, MiniBooNE); seven further OpenML streams we added for their non-Gaussian class-conditionals (wine_quality, phoneme, segment, wall-robot, JapaneseVowels, gas-drift, optdigits); and six benchmarks of Table 2 (Covertype, Shuttle, Letter, HAR, Pendigits, Split-MNIST). Four more streams in the sweep script (Bioresponse, credit, heloc, Isolet) are not part of this saved run and are not reported. OpenML streams use numeric features, are standardised on the pooled data before an 80/20 stratified split (same caveat as above), are capped at 25,000 examples, and present one class per task. All methods use the default hyperparameters of the main benchmark and five seeds. The baseline is whichever of SQDA and SLDA has the higher mean, which favours it; the gap is MIST-K minus the baseline, from unrounded means, and $|\mathrm{gap}| \leq 0.02$ is a tie. Only means were saved.

MIST-K wins on 4 streams, ties on 5, and loses on 10. Dimension alone does not explain the outcome: eye_movements and wall-robot both have d=24 and fall on opposite sides, and the correlation between d and the gap is -0.42. MIST-K also keeps a single leaf on 15 of the 19 streams, so the comparison is mostly between leaf models.

Table 28: MIST-K versus the better of SQDA and SLDA on 19 real streams, five seeds, sorted by gap. "Leaves" is the mean number of MIST-K leaves.

Stream	d	Leaves	MIST-K	Best global	Gap	Outcome
wall-robot	24	1.0	0.844	0.757 (SQDA)	$+0.086$	win
pol	48	1.0	0.210	0.188 (SQDA)	$+0.021$	win
Covertype	54	1.0	0.694	0.673 (SLDA)	$+0.021$	win
MiniBooNE	50	1.0	0.832	0.811 (SLDA)	$+0.021$	win
electricity	7	1.0	0.721	0.709 (SLDA)	$+0.012$	tie
Shuttle	9	1.0	0.783	0.774 (SQDA)	$+0.009$	tie
wine_quality	11	1.0	0.253	0.260 (SQDA)	-0.007	tie
phoneme	5	1.0	0.769	0.777 (SQDA)	-0.008	tie
segment	19	2.4	0.896	0.912 (SLDA)	-0.016	tie
eye_movements	24	1.0	0.443	0.487 (SLDA)	-0.044	loss
MagicTelescope	10	1.0	0.688	0.733 (SLDA)	-0.044	loss
jannis	54	1.0	0.444	0.519 (SQDA)	-0.075	loss
optdigits	64	1.0	0.876	0.965 (SQDA)	-0.089	loss
Pendigits	16	2.4	0.877	0.982 (SQDA)	-0.106	loss
JapaneseVowels	14	1.0	0.824	0.968 (SQDA)	-0.144	loss
Split-MNIST	50	2.0	0.772	0.940 (SQDA)	-0.168	loss
HAR	561	2.0	0.625	0.815 (SLDA)	-0.190	loss
Letter	16	1.0	0.692	0.895 (SQDA)	-0.203	loss
gas-drift	128	1.0	0.678	0.964 (SQDA)	-0.286	loss

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction accurately state all three contributions (tight McDiarmid calibration, Bayesian inheritance, unified KLL sketching) and their scope. Theoretical claims are backed by Theorem 5.1 and Theorem 5.3; empirical claims are supported by Tables 1– 4. Aspirational goals (time-uniform confidence sequences) are explicitly deferred to future work in Section 7.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Section 7 explicitly discusses two limitations: MIST-G's reliance on Gaussian class-conditional assumptions (addressed by MIST-K via sketch-based estimation), and the absence of time-uniform confidence sequences over infinite horizons (identified as future work). Memory scaling and throughput trade-offs are characterized in Appendix G.1 with Pareto-frontier analysis.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

   Justification: All theoretical results (Theorem 5.1, Theorem 5.3, Corollary 5.4, Proposition 5.5) include full proofs in Appendix J, with all assumptions tabulated in Table 18. Proof sketches with intuition are provided in the main text for each result. All supporting lemmas are numbered and cross-referenced.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: Section 6.1 specifies all datasets, baselines, and the strict Online CIL protocol. Table 5 (Appendix D.1) lists all fixed hyperparameters applied uniformly without per-dataset tuning. Algorithms 1–3 provide complete pseudocode for the MIST update loop, Bayesian inheritance split, and sketch-based leaf prediction.

   Guidelines:

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

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [No]

   Justification: Code is not released at submission time to preserve anonymity. All standard datasets used are publicly available. The algorithm is fully described via pseudocode in Algorithms 1-2 and hyperparameters in Table 5, providing a sufficient basis for independent reimplementation. Code will be released with the camera-ready version.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: Section 6.1 describes all datasets, preprocessing steps (PCA for Split-MNIST), and the strict single-pass exemplar-free Online CIL protocol. Table 5 (Appendix D.1) reports all hyperparameters with their theoretically motivated defaults. Appendix D.2 documents the no-reset protocol rationale for tree-based baselines.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

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

   Answer: [Yes]

Justification: All main results (Tables 1, 2, 4, 7) report mean $\pm$ std over 5 independent random seeds. Ablation results (Table 3) use 5 seeds as stated in the caption. Error bars represent one standard deviation across seeds under fixed hyperparameters and data ordering.

Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: Table 12 (Appendix G.1) reports memory footprint (KB) and throughput (samples/s) per method on two representative datasets. All experiments were conducted on a single machine with an Intel Core i9-14900HX CPU (24 cores, up to 5.8 GHz), 16 GB DDR5 RAM, and an NVIDIA GeForce RTX 4060 Laptop GPU (8 GB GDDR6); all methods run CPU-only with no GPU utilization during training or inference.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: This work presents foundational algorithmic research on streaming decision trees for tabular data. No human subjects, personal data, or sensitive information are involved. The research conforms with the NeurIPS Code of Ethics in all respects.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: This paper presents foundational algorithmic research without direct deployment in a specific application. The method's exemplar-free design is a positive privacy consideration: no raw data is retained, which is beneficial for sensitive tabular domains such as finance and healthcare. No direct path to negative societal impact is identified for this class of algorithm.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The paper introduces a streaming decision tree algorithm and synthetic benchmark datasets. No pre-trained language models, image generators, or internet-scraped datasets are released. No safeguards beyond standard practice are required.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All datasets used (Covertype, Pendigits, Shuttle, Letter, Wine, Iris, HAR, Split-MNIST) are publicly available benchmarks cited via their original publications. Baseline implementations use published defaults cited accordingly. Standard UCI/OpenML dataset licenses are documented at `paperswithcode.com/datasets`.

    Guidelines:

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

13. **New assets**

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

    Answer: [Yes]

    Justification: Eight stress-test synthetic datasets are introduced and fully described in Appendix I, including their generative distributions, parameters, and protocol. Generation code is provided in the supplementary materials. No new pre-trained models or external datasets are released.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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

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

    Answer: [N/A]

    Justification: This paper does not involve any crowdsourcing or experiments with human subjects. All data are either synthetic or pre-existing public benchmarks.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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

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

    Answer: [N/A]

    Justification: No human subjects are involved in this research. IRB approval is not applicable.

    Guidelines:

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

16. **Declaration of LLM usage**

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

    Answer: [N/A]

    Justification: LLMs are not used as any component of the core methodology. Any use of LLMs was limited to writing assistance only and does not impact the scientific rigor or originality of the research.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.