
A STATISTICAL-ML HYBRID APPROACH

Robust Selectivity Estimation on Skewed Data

The Hybrid FD-CDF Estimator - partitioning,
waterfall model selection, and adaptive
maintenance under fixed memory & latency
budgets.

AUTHOR

Unaspekov Timur

DEPARTMENT CHAIR

Michael Lewis

SUPERVISOR

Yesdaulet Izenov

DATE

4 May 2026

Agenda

I. Motivation & Problem

II. Related Work & Research Gap

III. Hybrid FD-CDF Design

IV. Experimental Setup

V. Static & Drift Results

VI. Conclusion & Future Work

Selectivity drives the plan cost

A small cardinality error at the leaves
compounds exponentially up the plan tree.

- Commercial DBMSs use **Equi-Width** and **Equi-Depth** histograms - $O(1)$ and $O(\log N)$ lookups.
- Both assume **intra-bucket uniformity**.
- Real workloads are **skewed**, correlated, and non-stationary.

$$s = |\sigma_P(T)| / N$$

Selectivity of predicate P over table T

Two competing failure modes

CLASSICAL

Efficient but inaccurate under skew

Equi-Width and Equi-Depth offer constant-time evaluation, but rely on an intra-bucket uniformity assumption that fails on non-uniform distributions.

p95 Q-err on Zipf ($\alpha=1.5$) $\rightarrow$ 25.0

NEURAL

Accurate but too slow

MSCN, Naru, DeepDB improve accuracy but incur 10–100× inference overhead - unworkable when a workload asks for millions of predicate estimates.

Naru latency $\gg$ per-predicate budget

And: data drifts. Any estimator must adapt without full retraining.

Four design constraints

01

Accuracy on skew

Competitive with neural estimators on Zipf, sparse clusters, heavy-hitter regions.

02

Inference latency

Within an equi-width order of magnitude. Sub-millisecond per candidate plan.

03

Drift adaptivity

Incremental updates. No full rebuild for non-stationary distributions.

04

Memory bound

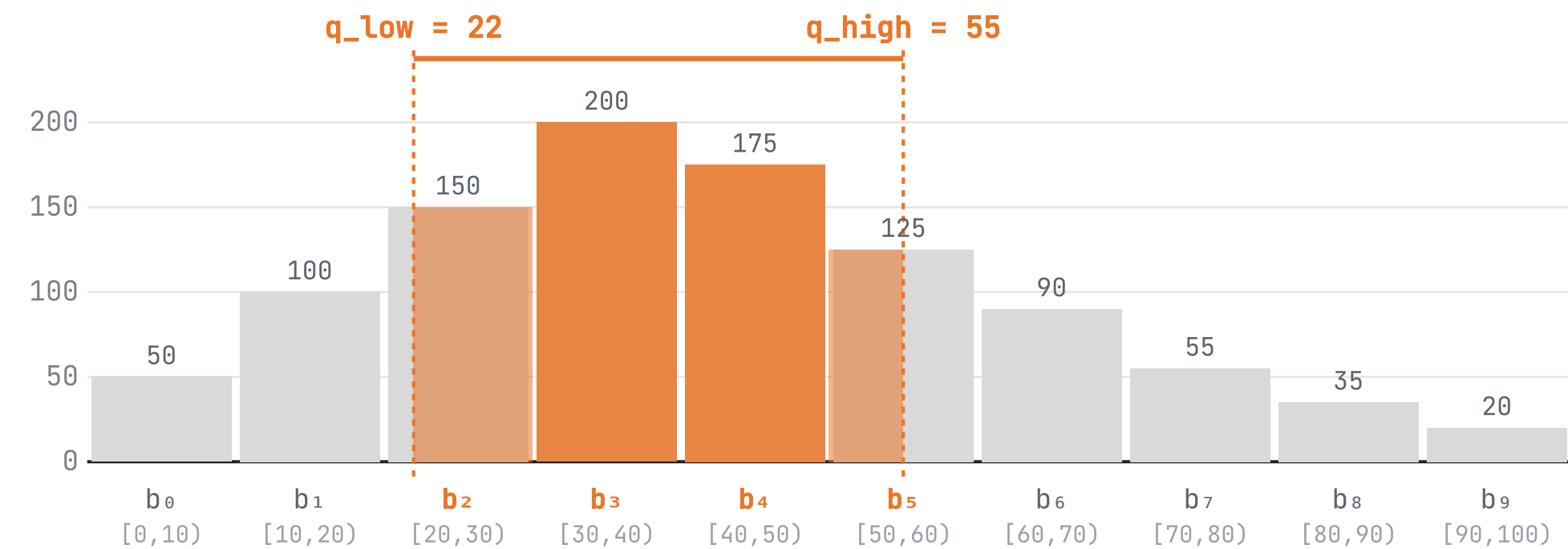
Fixed per-column budget - deployable inside a real DBMS optimizer.

Classical summaries share a common limitation

APPROACH	STRENGTH	STRUCTURAL WEAKNESS	COST
Equi-Width	$O(1)$ lookup, trivial build	Intra-bucket uniformity	cheap
Equi-Depth	Balanced row counts	Heavy-hitter hallucination	cheap
V-Optimal	Minimises intra-bucket variance	$O(N^2)$ construction	infeasible
KDE	Smooth, differentiable	Bandwidth selection under skew	expensive
Sampling	Captures correlation	Fails on high-selectivity predicates	variable
LEO / ISOMER	Adaptive from feedback	Max-entropy per-query cost	high

Every row assumes the density inside a partition is constant. It is not.

Three lookups under one assumption



Prefix-sum array $P[i] = \sum \text{counts}[0..i]$

interior sum = $P[B_{\text{end}}] - P[B_{\text{start}}+1]$

counts[i]	50	100	150	200	175	125	90	55	35	20	
P[i]	0	50	150	300	500	675	800	890	945	980	1000
	0	1	2	3	4	5	6	7	8	9	10

Interior buckets $b_3, b_4 \rightarrow P[5] - P[3] = 675 - 300 = 375$ - one subtraction, no matter how many buckets the query spans.

SETUP

Domain $D = [0, 100] \cdot h = 10 \cdot n = 10 \text{ buckets} \cdot N = 1\,000$

Query: $q_{\text{low}} = 22, q_{\text{high}} = 55$

① Locate edge buckets

$B_{\text{start}} = \lfloor (22 - 0) / 10 \rfloor = 2$

$B_{\text{end}} = \lfloor (55 - 0) / 10 \rfloor = 5$

② Edge buckets · uniform pro-rate

b_2 : overlap $(30 - 22)/10 = 0.80 \rightarrow 0.80 \times 150 = 120$

b_5 : overlap $(55 - 50)/10 = 0.50 \rightarrow 0.50 \times 125 = 62.5$

③ Interior · prefix sum

$P[B_{\text{end}}] - P[B_{\text{start}}+1] = P[5] - P[3] = 675 - 300 = 375$

ESTIMATED COUNT

$120 + 375 + 62.5 \approx 557.5$

selectivity = $557.5 / 1000 = 0.5575$ - accurate only when the two edge buckets are internally uniform.

Neural estimators: accuracy vs. spectral bias

Supervised & Autoregressive

- **MSCN** - maps query features to estimate; sensitive to workload shift.
- **Naru / DeepDB** - chain-rule joint density; range predicates require numerical integration.
- **Astrid, SAFE** - specialised for strings / spatial; not portable across data types.

The spectral bias problem

Fully-connected MLPs learn low-frequency components orders of magnitude faster than high-frequency spikes.

Database distributions are Zipfian - sharp, isolated density peaks are exactly what the network smooths away.

$$\gamma(v) = [\cos(2\pi b_1 v), \sin(2\pi b_1 v), \dots]$$

Fourier-feature lift (Tancik et al., NeurIPS 2020)

Existing work forces a binary choice -
model the dataset as uniformly easy
or uniformly hard.
But skew is local.

MISSING: A PRINCIPLED PARTITIONING RULE + A POLICY THAT MATCHES MODEL COMPLEXITY TO LOCAL DIFFICULTY.

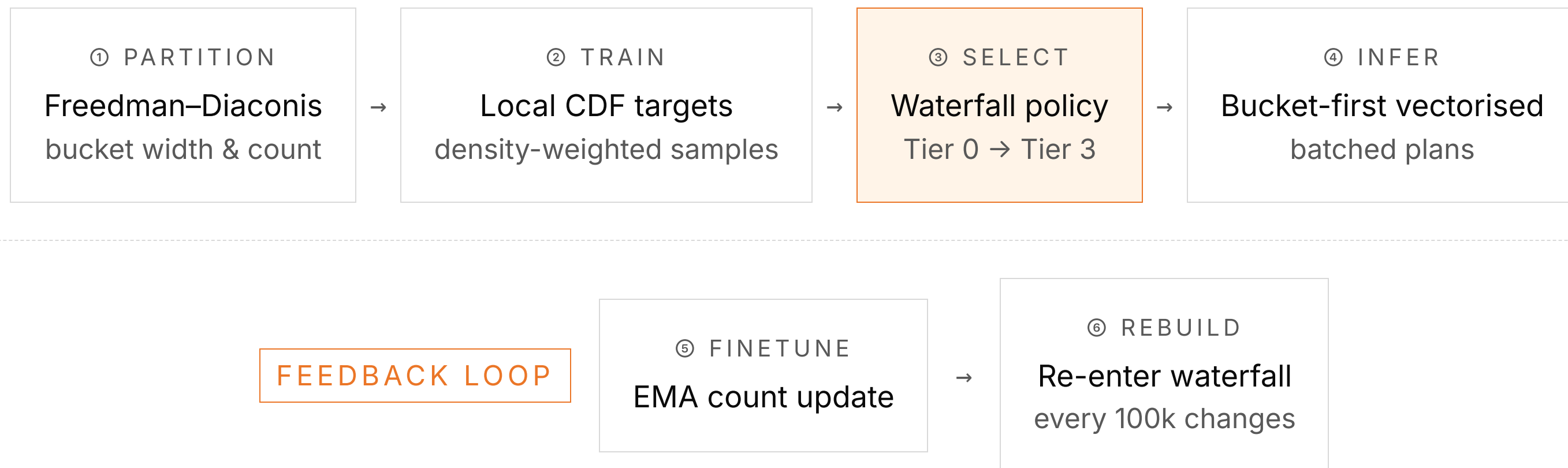


The Hybrid FD-CDF Estimator

DESIGN · PARTITIONING · WATERFALL · INFERENCE · MAINTENANCE



Four stages, built atop equi-width



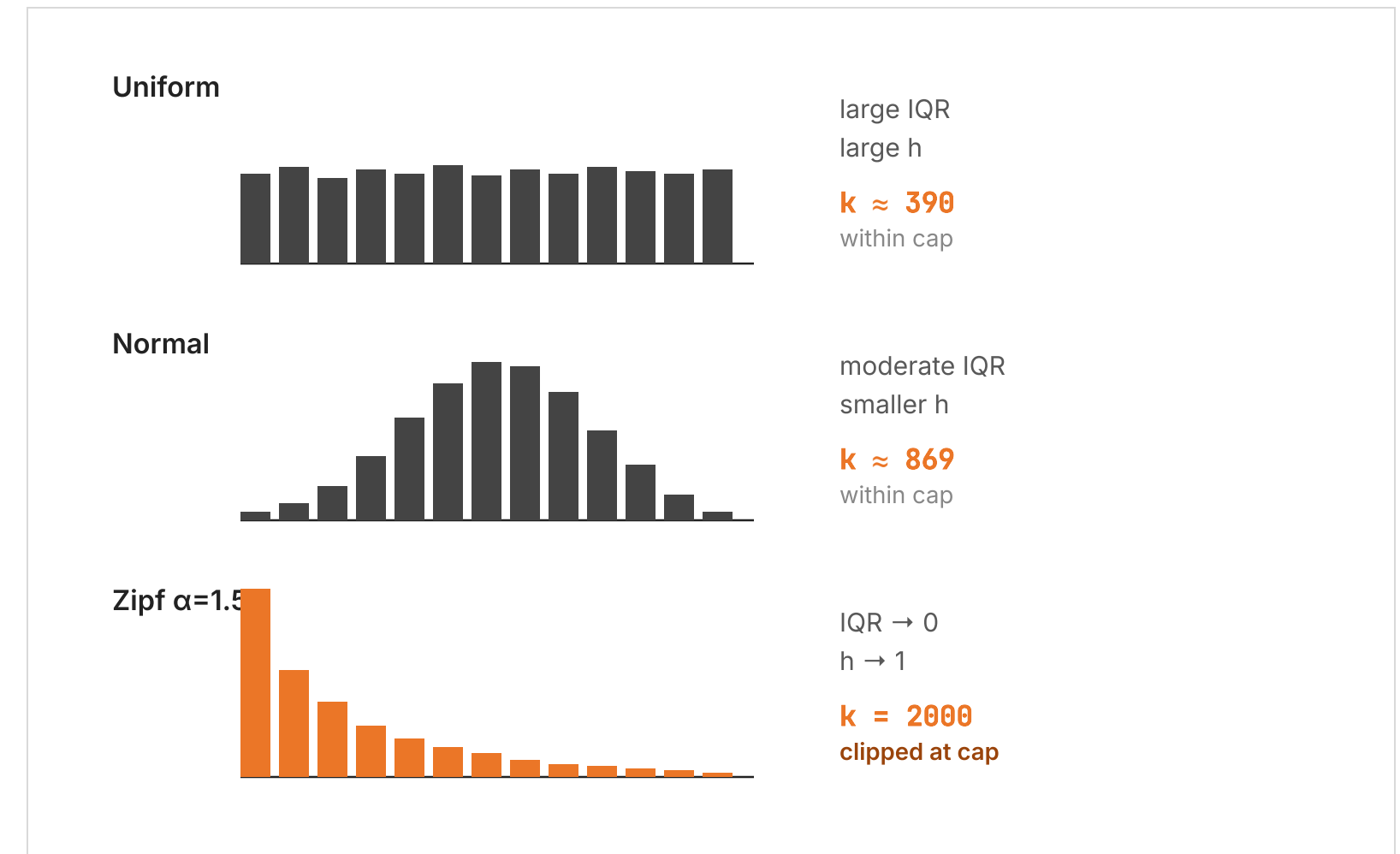
Bucket width derived from data

$$h = 2 \cdot \text{IQR}(x) / \sqrt[3]{N}$$

Freedman & Diaconis, 1981

- **IQR** = $Q_3 - Q_1$. Robust to sentinel values $(-1, 9999)$.
- Computed on a **reservoir sample** $|S| = 10^5 - O(|S| \log |S|)$, not $O(N \log N)$.
- Removes one hardcoded hyperparameter from the histogram.

How the data chooses h



Saturation as a skew signal

$$k_{\text{target}} = \lceil (\text{max} - \text{min}) / h \rceil$$

$$k_{\text{final}} = \min(k_{\text{target}}, k_{\text{max}})$$

$k_{\text{max}} = 2,000$ (realistic one-column budget)

Two regimes

- **Low kurtosis** (uniform, anti-Zipf) → high IQR → large h → $k \approx 400$. Within the memory cap; low-capacity models are sufficient.
- **High kurtosis** (Zipf) → IQR → 0 → $h \rightarrow 1 \rightarrow k_{\text{target}} \gg k_{\text{max}}$.
Clipping is the trigger for heavy-model tiers.

The cap is not a limitation - it is a signal that the waterfall should escalate.

Per-bucket CDF targets in unit space

$$x = (v - L) / (H - L + 1)$$
$$y = (\text{rank}(v) - \text{rank}(L-1)) / C$$

C = bucket cardinality · unit-bounded for tier-agnostic training

Complementary sampling

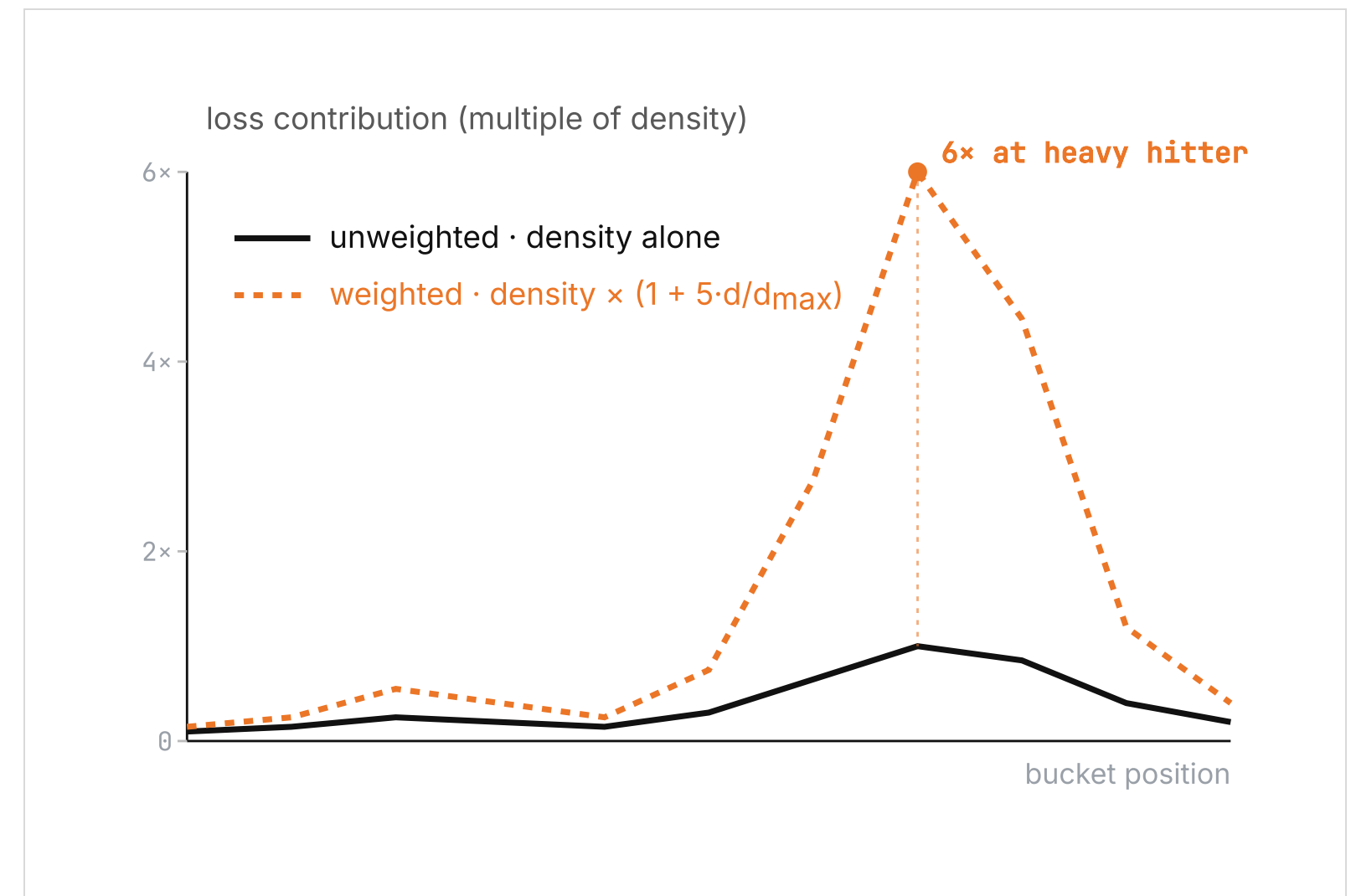
- **Uniform** - coarse coverage of the range.
- **Grid** - $2 \cdot p/\text{bucket}$ evenly-spaced points catch non-linear transitions.
- **Edge** - forces $L, L+1, H-1, H$ into every training set → no seam errors between buckets.
- Independent validation set of $2 \cdot p/\text{bucket}$ random points.

Weighting errors in proportion to cost-model impact

$$w = 1.0 + 5.0 \cdot (P_{\text{local}} / \max(P_{\text{bucket}}))$$

$w \in [1, 6]$ - concentrates capacity on dense regions

The low-capacity parametric learners are ordinary least squares. Without weighting, every sample counts equally - a heavy-hitter's selectivity error and a low-density sample's error weigh the same on the loss. Downstream, they do not.



Allocate capacity in proportion to bucket complexity

TIER	MODEL • SELECTED WHEN BUCKET IS	STORED / ACCEPT	INFERENCE
<i>Tier 0</i>	Identity • $F(x) = x$ Approximately uniform. Early-exit - no parameters stored.	STORED • ACCEPT 0 params • ≤ 1.30	COST $O(1)$
<i>Tier 1</i>	Parametric • ridge $\alpha=1$ Monotone, mild curvature. Linear $\rightarrow \sqrt{x}$ power $\rightarrow$ log-linear; density-weighted.	STORED • ACCEPT 2 coefs • ≤ 1.05	COST $O(1)$
<i>Tier 2</i>	Cubic polynomial • ridge $\alpha=0.1$ Smoothly non-linear. Captures curvature parametric forms miss.	STORED • ACCEPT 4 coefs • ≤ 1.05	COST $O(1)$
<i>Tier 3</i>	Decision tree (depth 4) + isotonic Staircase / sparse-cluster. Handles discontinuities & enforces monotonicity.	STORED • ACCEPT piecewise • ≤ 1.05	COST $O(\log)$
<i>Tier 4</i>	Fourier-feature MLP • 32 bands Chaotic / heavy-hitter Zipf head. 66-d lift • 2-layer (64, 32) • Adam, 500 iters. Fallback when no lower tier accepts.	STORED • ACCEPT 66-d lift • fallback	COST $O(1)$ • MLP

Decision rule. Accept the cheapest tier whose median Q-err on a 500-query held-out probe is at or below its threshold; otherwise escalate. Thresholds: 1.30 early-exit • 1.05 promote. Per-bucket training is parallelised with a ThreadPoolExecutor.

Bucket-first batched prediction

$$\text{pred} = \max(\text{CDF}(x_{hi}) - \text{CDF}(x_{lo}) \cdot C, 0.05 \cdot \text{uniform})$$

Uniform floor = safety net against learned-CDF underflow

- Vectorised overlap mask over buckets with NumPy broadcasting.
- Dispatch table over mod_types (identity / linear / power / log / poly3 / tree / iso / Fourier-MLP).
- Predicates grouped by bucket, not by query - amortises per-bucket work across the workload.

Pipeline

STEP 1

Workload → overlap mask over buckets



STEP 2

Group predicates by touched bucket



STEP 3

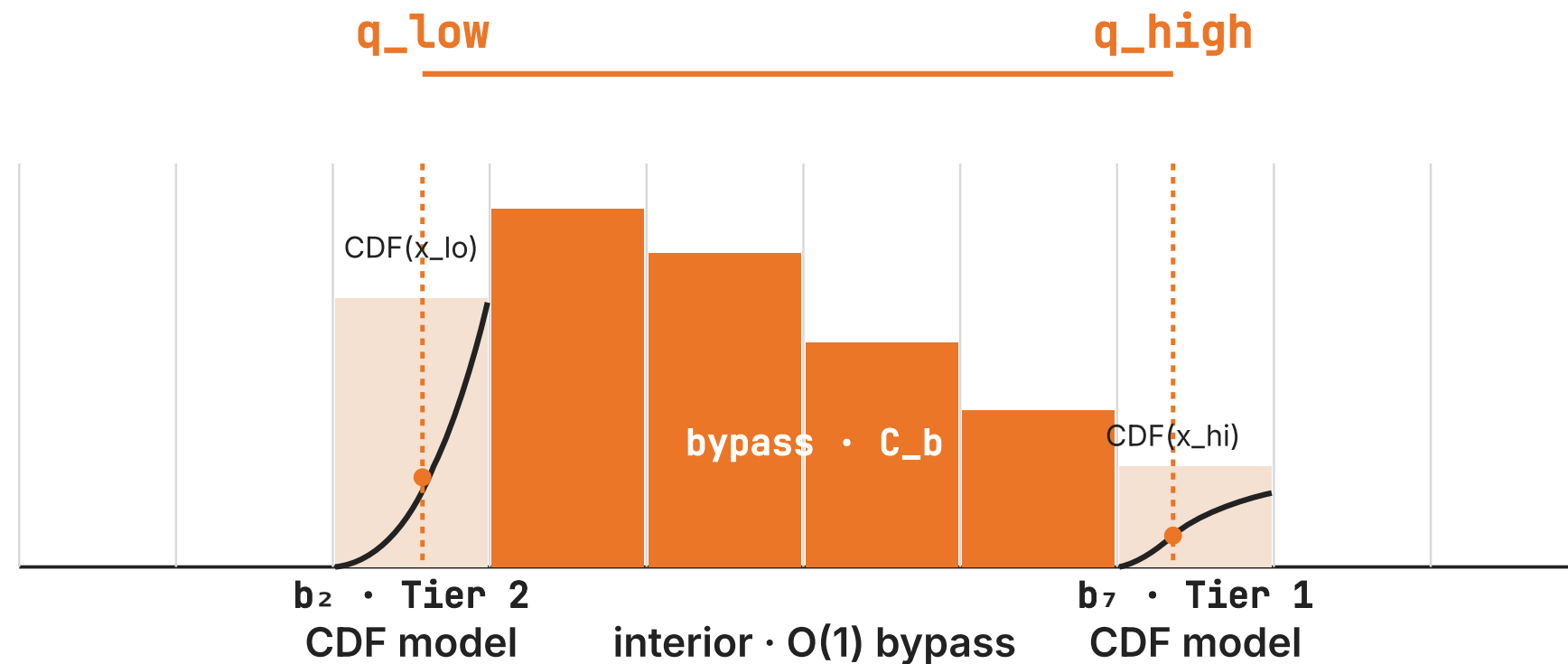
Vectorised per-bucket model eval



STEP 4

Aggregate selectivity per query

Same query, learned edge correction



Only edge buckets invoke a learned CDF - the bypass path keeps the interior cost at equi-width.

① Bucket lookup

Same as equi-width - divide by global width h . $O(1)$.

② Interior bypass

For every fully-enclosed bucket, the estimator bypasses model inference and reads the stored mass C_b . The cost is independent of the assigned tier - Tier 0 through Tier 3.

③ Edge buckets • learned CDF

Normalize overlap to $x \in [0, 1]$, evaluate the model at both ends:

$$\text{Count}_{\text{pred}} = \max(0, \text{CDF}(x_{\text{hi}}) - \text{CDF}(x_{\text{lo}})) \cdot C_b$$

Eq. 3.9 - only the two edges pay model cost

Resulting property. A query touching N buckets invokes at most 2 model evaluations - independent of N .

DBS - overlap-proportional EMA on true counts

$$\text{share}_b = \text{overlap}_b / \sum \text{overlap}$$

$$m_{\text{true},b} += y_{\text{true}} \cdot \text{share}_b$$

$$C_{\text{new}} = C_{\text{old}} \cdot (1 - \alpha) + (C_{\text{old}} \cdot r) \cdot \alpha$$

$$r = m_{\text{true}} / m_{\text{pred}} \text{ clipped to } [0.2, 5.0] \cdot \alpha = 0.15$$

$$\text{if } C_{\text{old}} < 1 \wedge m_{\text{true}} > 0 : C_{\text{new}} = m_{\text{true}} \cdot \alpha$$

Zero-count revival - corrects reservoir blind-spots

Trigger & propagation

- Bad query: $Q\text{-err} > \epsilon_{\text{bad}} = 1.2$.
- Error is distributed across overlapping buckets in proportion to spatial overlap - not uniformly.
- Clamp prevents a single outlier from erasing learned counts.
- Vectorised arrays re-baked after update; no global sort.

Checkpointed re-entry into the waterfall

RUNTIME

Detect

Per-bucket median Q-err over ≥ 5 touching queries exceeds 1.30
→ marked.

PER QUERY

ACCUMULATE

Checkpoint

Every 100,000 inserts,
updates, or deletes - amortised
over real workload.

BATCHED

RE-ENTER

Waterfall

A previously marked bucket re-enters tiers 0 through 3 using the current reservoir sample. A bucket that has shifted from uniform to Zipf escalates accordingly.

BOUNDED OVERHEAD

RESUME

Serve

Estimator is re-aligned; finetune loop continues with the new model for that bucket.

CONVERGED

IV

Experiments & Results

SYNTHETIC FRACTAL DATASETS · DSB · STATIC & DRIFT



Datasets, workloads, baselines

DATASETS

- **Synthetic - Simple.** Fractal generation; macro & micro distributions matched (Uniform, Normal, Anti-Zipf).
- **Synthetic - Hard.** Macro & micro paired adversarially (e.g. Normal-macro × Zipf-micro) to maximise intra-bucket non-uniformity.
- **Real - Microsoft DSB.** Three columns · ws_item_sk, cr_item_sk, cr_returned_time_sk. N = 6×10⁷ rows.

BASELINES

Equi-Width - O(1) static; the deployable reference.	EquiHist - adaptive feedback, $\eta = 0.5$, updates every query.	10 ⁶ queries / run · seed 42
--	--	---

WORKLOADS

- **Wide.** Width U(100, 10 000) - exercises the interior bypass.
- **Narrow.** Width U(1, 100) - stresses intra-bucket precision; this is the demanding regime.
- **Data-driven.** Anchors sampled from the reservoir → every query targets a populated region; no trivial-zero queries.

Q-error - degree of estimation error

$$\text{Q-err}(q) = \max(y_{\text{true}}/y_{\text{pred}}, y_{\text{pred}}/y_{\text{true}})$$

Moerkotte et al., VLDB 2009 · symmetric ratio

- **What it measures.** The degree of estimation error: a Q-err of 2 means a 2× over- *or* under-estimate; 1 is exact.
- **Why this metric.** Cardinality propagates multiplicatively up the plan tree, so a multiplicative metric predicts plan-quality degradation.
- **Reported.** Median, P95, Average over 10^6 queries; **MAE** separately for absolute magnitude (hash-join RAM).

Drift decomposition

- **M_{shock}** - Q-err immediately after a drift step.
- **M_{ft}** - after finetune converges.
- **M_{rb}** - after rebuild at the checkpoint.
- **Finetune / Rebuild counts** - structural cost.

Fractal · Narrow · Data-Driven

DISTRIBUTION	MODEL	MEDIAN	P95	AVERAGE	INFER (S)
Normal	EquiWidth	1.410	2.585	1.620	1.716
	EquiHist	1.359	2.528	1.588	86.21
	Hybrid FD-CDF	1.104	1.671	1.244	1.720
Zipf ($\alpha=1.5$)	EquiWidth	1.045	11.13	2.988	0.618
	EquiHist	1.250	11.45	503.9	187.6
	Hybrid FD-CDF	1.001	1.017	1.049	0.655
Sparse Cluster	EquiWidth	4.223	20.55	6.318	0.208
	EquiHist	1.469	5.704	2.108	40.06
	Hybrid FD-CDF	1.061	5.206	2.300	0.225

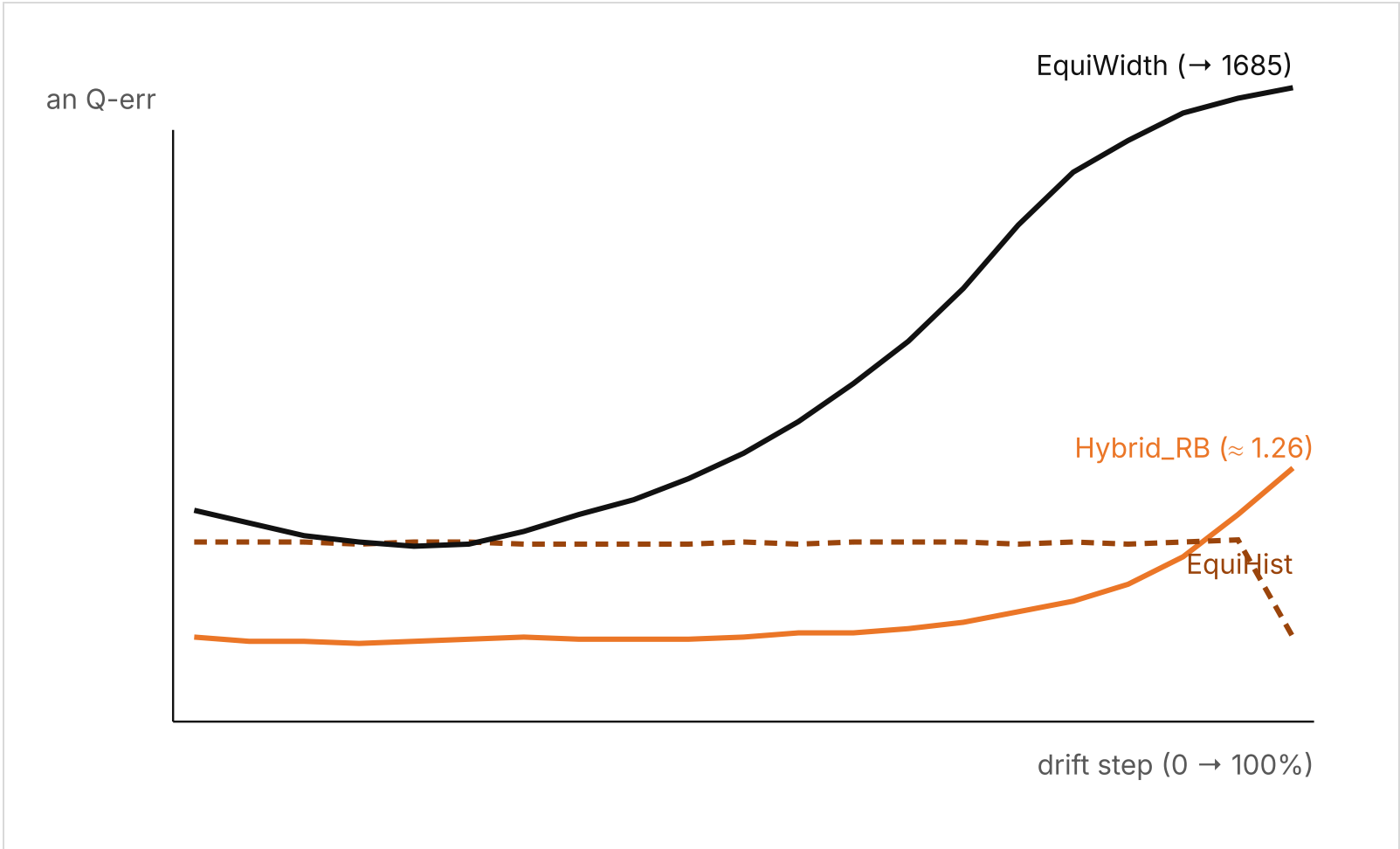
Hybrid reaches best median on every skewed distribution while inference latency stays within ~10% of Equi-Width. EquiHist's accuracy is swamped by its per-query update cost.

DSB benchmark - real-world validation

COLUMN	MODEL	MEDIAN	P95	AVG	TRAIN (S)	INFER (S)
ws_item_sk	EquiWidth	1.024	1.900	1.170	0.002	0.339
	EquiHist	1.055	1.725	1.167	9.596	9.271
	Hybrid	1.014	1.057	1.020	0.200	0.376
cr_item_sk	EquiWidth	1.021	2.081	1.186	0.002	0.239
	EquiHist	1.100	1.634	1.183	6.774	6.388
	Hybrid	1.010	1.038	1.026	0.164	0.278
cr_ret_time_sk	EquiWidth	1.011	1.040	1.014	0.003	0.283
	EquiHist	1.004	1.013	1.005	7.950	7.568
	Hybrid	1.000	1.039	1.014	0.221	0.300

cr_item_sk: Equi-Width's p95 of 2.08 crosses the join-order threshold; Hybrid holds 1.04. EquiHist matches the Hybrid on accuracy but at ~30× inference latency.

Drift - equi-width collapses, hybrid converges



Across 21 drift steps

HYBRID MEDIAN

~1.19

stable through drift

EQUIWIDTH AVG

1719

final step

FT + RB TIME

≈ EW

single-rebuild budget

EQUIHIST INFER

166 s

per 10⁶ queries

DSB drift - partitioning absorbs distributional shifts

CR_ITEM_SK • HYBRID
MEDIAN

1.013

→ 1.033

across all 21 steps

REBUILT BUCKETS / STEP

0

finetune alone suffices

CR_ITEM_SK • EW P95

1.81

final step

EQUIHIST INFERENCE

~13 s

vs. Hybrid sub-second

Real-world drift is structurally less severe than the synthetic fractal hard case - only single-digit numbers of buckets require rebuild per checkpoint.

This confirms the hypothesis: a Freedman–Diaconis partitioning, once established, is robust to typical real-world drift. Most adaptation load is absorbed by the low-cost finetune path; rebuild is a safety net.

Equi-width latency, neural-level accuracy, drift-adaptive

Contributions

- **FD-partitioned** equi-width replacement - one fewer hyperparameter.
- **Waterfall** Tier-0 → Tier-3 selection - capacity matched to local skew.
- **Vectorised** bucket-first inference - parity with equi-width latency.
- **Finetune + Rebuild** maintenance - bounded drift overhead.

Future work

- Reservoir maintenance strategies under high-velocity inserts.
- Richer selection policies (cost-aware tier promotion).
- Checkpoint interval tuning & online scheduling of rebuilds.
- Standardised drift benchmarks with explicit checkpoints.

Thank you - I welcome your questions.

Unaspekov Timur · Supervisor: Y. Izenov

A

Appendix Tier Details

PER-TIER DESCRIPTION · FOR Q&A



Tiers 0–1 – baseline cases

TIER 0 • IDENTITY

Early termination at $F(x)=x$

If the bucket is approximately uniform, additional capacity yields no measurable benefit. Threshold: median Q-error ≤ 1.3 on the tier's probe.

No stored parameters; inference reduces to a coordinate transform.

TIER 1 • PARAMETRIC

Linear • $\sqrt{x}$ power • log-linear

- Linear accepts first if Q-err < 1.05 – a hard early-exit shortcut.
- Ridge $\alpha=1 + \text{density weights} \cdot \ln(x + 10^{-7})$ for zero-boundary stability.
- Per-bucket training is parallelised (ThreadPoolExecutor).

Tiers 2–3 - curvature, staircases, monotonicity

Cubic polynomial

Ridge $\alpha=0.1$ on degree-3 features - captures smooth curvature parametric forms cannot.

4 coefficients, still $O(1)$ inference.

Decision tree (depth = 4)

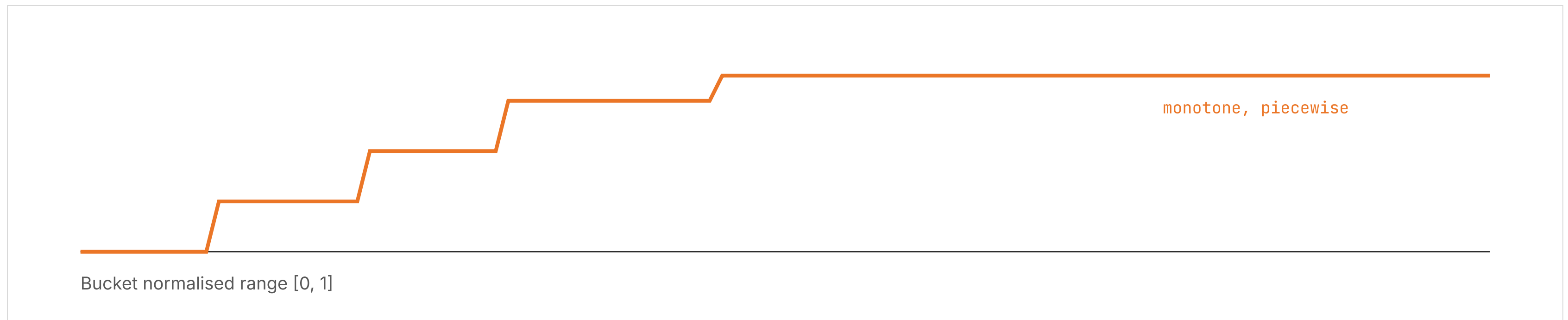
Staircase handler - long flats followed by cliffs, common in sparse clusters.

Depth cap prevents reservoir-sample overfit.

Isotonic regression

Enforces monotone CDF - prevents ridge's classic failure of overestimating count across an empty gap.

Trades local precision for global consistency.



Tier 4 - Fourier features mitigate spectral bias

$$\gamma(x) = [\cos(\pi f_i x), \sin(\pi f_i x)] \oplus x \oplus \ln(x + 10^{-7})$$

$K = 32$ bands · log-spaced $10^0 \rightarrow 10^3$ · output dim = 66

Low bands preserve curvature; high bands isolate heavy-hitters. The MLP then learns a linear combination of scales instead of fighting its own low-pass bias.

ARCHITECTURE

64→32

2 hidden layers

OPTIMIZER

Adam

500 iterations

INPUT LIFT

66-d

from 1 scalar

PROMOTION

> 1.05

median Q-err fallback