SpacSec 2026

Analytic Unary Density Trees for Inspectable Anomaly Scoring

Andrei V. Konstantinov Lev V. Utkin Stanislav R. Kirpichenko

Peter the Great St. Petersburg Polytechnic University

Security motivation

- Normal telemetry is abundant; reliable labels for new attacks are often scarce or delayed.
- A density model can rank events by rarity and expose which feature ranges produced a low score.
- A useful monitoring component should be **inspectable** and work when labels are unavailable.

Scope of this work

Density is a representation-dependent rarity signal. It does not, by itself, define malicious behavior or an operational alert threshold.

Density estimation as unary classification

Let $X \sim f$ on $\mathcal{X} = [0,1]^d$ and $Z \sim U(\mathcal{X})$. Classify real observations as $Y = 1$ and uniform background as $Y = 0$, such that $P(Y = 1) = \alpha$:

$$T = \begin{cases} X, & Y = 1 \\ Z, & Y = 0 \end{cases}$$

$$q(t) = P(Y = 1 \mid T = t) = \frac{\alpha \cdot f(t)}{\alpha \cdot f(t) + (1-\alpha) \cdot 1} \quad \Longrightarrow \quad \boxed{f(x) = \frac{q(x)}{1 - q(x)}}$$

Equal class priors and a unit-volume uniform background give this identity. The classifier's posterior odds become a density estimate.

Where sampled background costs arise

A sampled unary objective draws m artificial background points:

$$\widehat{\mathcal{R}}_{\mathrm{MC}}(g) = \frac{1}{m}\sum_{j=1}^{m} \ell(g(Z_j), 0) + \frac{1}{n}\sum_{i=1}^{n} \ell(g(X_i), 1).$$

- The first term is a Monte Carlo estimate of an expectation over $[0,1]^d$.
- Background points become sparse in a high-dimensional cube.
- A tree has a simpler option: **integrate over each leaf exactly**.

Analytic objective for a tree

Replace sampled uniform points by their population loss:

$$\mathcal{R}(g) = \int_{[0,1]^d} \ell(g(x), 0)\, dx + \frac{1}{n} \sum_{i=1}^{n} \ell(g(X_i), 1).$$

A tree has a constant score s_A within leaf A. With leaf volume $V_A = |A|$ and *global* empirical mass $\mu_A = n_A/n$,

$$\boxed{\mathcal{R}_A = V_A \ell(s_A, 0) + \mu_A \ell(s_A, 1).}$$

Two sufficient statistics: sample count and geometric volume.

Each candidate split has exact background mass

For a rectangular node $A = \prod_{k=1}^{d} [a_k, b_k]$, split feature j at θ:

$$V_L = V_A \frac{\theta - a_j}{b_j - a_j}, \qquad V_R = V_A - V_L.$$

- Sorted observations supply child counts n_L, n_R.
- Bounds supply child volumes V_L, V_R.
- Split gain follows from $(n_L/n, V_L)$ and $(n_R/n, V_R)$.

Normalization detail

Use n_L/n, not n_L/n_{parent}. Local renormalization would break the global density-ratio interpretation.

General second-order update: AUDT-Q

When a node is divided into children L, R, we use second-order loss approximation and consider the splitting as a **single gradient-based update step**

For a child B, current score s, and additive update δ:

$$\Psi_B(\delta) = V_B\ell(s+\delta, 0) + \mu_B\ell(s+\delta, 1) + \frac{\lambda}{2}\delta^2.$$

$$G_B = V_B\ell_0'(s) + \mu_B\ell_1'(s), \qquad H_B = V_B\ell_0''(s) + \mu_B\ell_1''(s).$$

$$\delta_B^{\mathrm{Q}} = -\frac{G_B}{H_B+\lambda}, \qquad \Delta_B^{\mathrm{Q}} = \frac{G_B^2}{2(H_B+\lambda)}.$$

Score each split by $\Delta_L^{\mathrm{Q}} + \Delta_R^{\mathrm{Q}}$; apply learning rate γ to the child update.

MSE update is exact

AUDT-MSE stores probability scores and uses $\ell(s, y) = \frac{1}{2}(s - y)^2$.

$$G_B = sV_B + (s-1)\mu_B, \qquad H_B = V_B + \mu_B,$$

$$\boxed{\delta_B^{\mathrm{MSE}} = -\frac{sV_B + (s-1)\mu_B}{V_B + \mu_B + \lambda}.}$$

The quadratic expansion equals the original MSE objective for a fixed split. The child update therefore minimizes the regularized leaf objective exactly.

BCE has a closed-form density: AUDT-BCE-CF

Store a logit z, with $p = \sigma(z)$. For $\lambda = 0$, the exact leaf optimum is

$$p_B = \frac{\mu_B}{V_B + \mu_B}, \qquad \delta_B^{\mathrm{CF}} = \mathrm{logit}(p_B) - z.$$

Posterior odds reduce to the familiar histogram ratio:

$$\boxed{\hat{f}_B = \frac{p_B}{1 - p_B} = \frac{\mu_B}{V_B}.}$$

The split locations are learned from the unary classification objective, while each final leaf keeps an explicit density estimate.

Normalization and calibration

The closed-form leaf densities integrate to one over the partition:

$$\int_{[0,1]^d} \hat{f}(x)\,dx = \sum_{A \in \text{leaves}} V_A \frac{\mu_A}{V_A} = \sum_A \mu_A = 1.$$

- **Path mode:** retain values accumulated while growing the tree.
- **Refit mode:** assign each final leaf its mass-over-volume value, with optional smoothing.
- A leaf-local KDE can smooth within a cell; sparse leaves fall back to a constant.

From a single tree to a rotation forest

AUDT / AUDF

- One tree yields transparent, axis-aligned feature intervals.
- Bagging many trees reduces variance and blockiness.
- Main experiments aggregate per-tree densities by arithmetic mean.

Rot-AUDF

- Each tree receives an orthogonal feature transform.
- Axis-aligned cells in transformed coordinates still have analytic volumes.
- Rotated rules become linear inequalities in original features.

Random-subspace aggregation is a density *score*, not necessarily a normalized full-joint density.

Anomaly-scoring modes

Unlabelled: fit a density to reference or mostly normal telemetry.

$$a(x) = -\log(\hat{f}(x) + \varepsilon).$$

Unary Density Ratio (labelled): fit one density per class and compare them with priors.

$$s_{\mathrm{DR}}(x) = \log \hat{p}(x \mid y = 1) - \log \hat{p}(x \mid y = 0) + \log \hat{\pi}_1 - \log \hat{\pi}_0.$$

Unary Joint (labelled): fit one density on augmented pairs $(x, e(y))$. Evaluate the same x with both class codes and normalize the resulting scores:

$$r_k(x) = \hat{p}_{XY}(x, e(k)), \qquad \hat{P}(Y = 1 \mid x) = \frac{r_1(x)}{r_0(x) + r_1(x)}.$$

Evaluation roadmap

1. **Density estimation:** grid KL on six low-dimensional synthetic distributions.
2. **Density-ratio ranking:** held-out AUROC on nine binary datasets, including 25- and 64-dimensional tasks.
3. **Anomaly scoring:** synthetic conditional-region tasks and a small UNSW-NB15 network-traffic pilot.

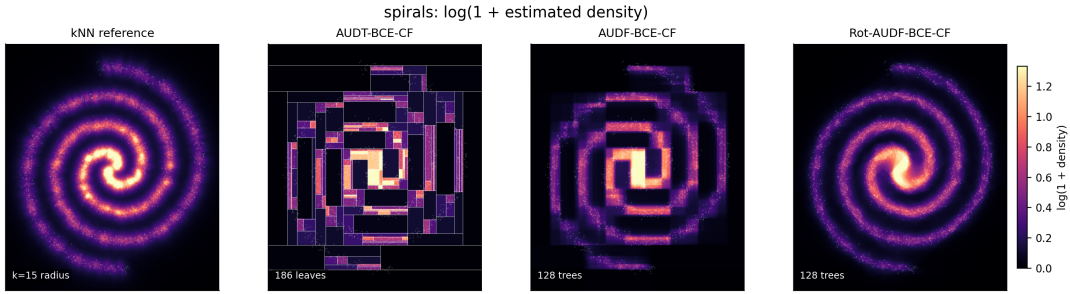
The metrics answer different questions: grid KL measures density fit; AUROC measures ranking in a classification or anomaly task.

Synthetic density: grid KL

Method	Gaussian 2D	Gaussian 3D	Ring	Moons	Ind. mix.	Octagon
Histogram	179.8	663.8	298.6	156.8	417.6	373.2
KDE	35.5	41.1	119.7	126.8	109.7	**70.8**
kNN	63.7	113.9	220.1	216.7	92.1	102.1
MLP, published	**8.34**	**34.63**	**76.07**	141.0	126.0	225.0
MLP, reproduced	98.4	152.7	351.3	149.5	531.7	270.8
AUDT-BCE-CF	118.8	241.3	317.9	280.2	217.0	226.1
Rot-AUDF	28.7	35.2	114.6	**108.9**	**68.2**	81.2

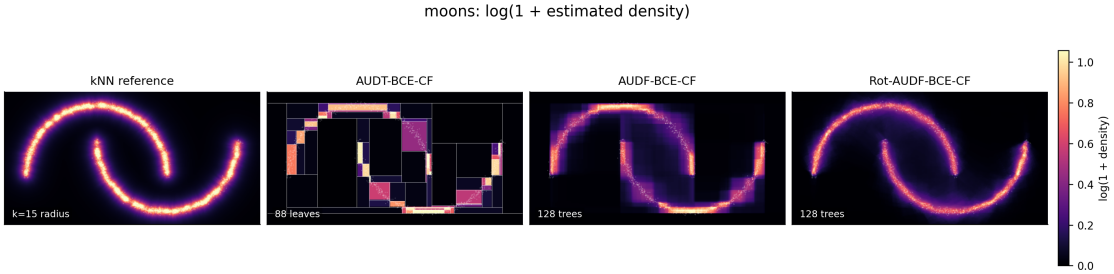
Rot-AUDF improves on the *published* MLP values for Moons, Independent mixture, and Octagon; KDE remains best on Octagon.

Curved support: intertwined spirals



Same color scale: $\log(1 + \hat{f})$. A single tree reveals leaf boxes; rotation forests follow the curved support more smoothly.

Curved support: moons



Tree boxes show the partition geometry. Bagging and rotations smooth the estimate along the two arcs.

Higher dimensions: density-ratio ranking

Grid KL becomes impractical as dimension rises. Fit $\hat{p}(x \mid y = 0)$ and $\hat{p}(x \mid y = 1)$, then test their log-density ratio on held-out folds.

- Nine binary datasets, dimensions 10 to 64.
- Five stratified folds; preprocessing is fit on each training fold.
- Report mean AUROC for the best preliminary-sweep configuration.

Interpretation

AUROC tests class ranking from estimated densities. It is not a direct measurement of density error or a nested model-selection study.

Density-ratio AUROC, moderate and high dimensions

Dataset	d	Unary DR	Unary Joint	KDE/kNN	Discriminative
Classification 10	10	0.954	0.966	0.969	0.969
Classification 20	20	0.954	0.953	0.949	0.967
Digits 3 vs 8	64	1.000	1.000	0.999	1.000
Digits 4 vs 9	64	1.000	0.999	0.999	1.000
Digits even vs odd	64	0.993	0.998	0.998	0.999
Embedded moons	25	0.982	0.973	0.970	0.999
Hastie 10	10	0.934	0.958	0.934	0.958
Sparse XOR 10	10	0.739	0.725	0.744	0.891
Sparse XOR 50	50	0.503	0.505	0.488	0.500

Unary density ratios rank well on several structured tasks. Sparse high-order interactions remain difficult.

Unlabelled synthetic anomaly scoring

Mean AUROC $\pm$ standard deviation; higher is better.

Dataset	d	Unary density	Isolation Forest
Conditional cells	8	0.964 ± 0.004	0.906 ± 0.006
Conditional cells	20	0.695 ± 0.060	0.632 ± 0.011
Sparse conditional	50	0.886 ± 0.013	0.570 ± 0.008
Sparse conditional	100	0.871 ± 0.019	0.536 ± 0.011

Unary density exceeds Isolation Forest on all four tasks

UNSW-NB15: small network-traffic pilot

Mean AUROC over three stratified folds; higher is better.

Subset	n	Anom. frac.	Labels hidden AUDF	Labels hidden IF	Normal-only AUDF	Normal-only IF	Labelled AUDF-DR	Labelled RF
Mixed	3000	0.639	**0.808**	0.795	**0.862**	0.798	0.950	**0.982**
Exploits	1605	0.325	**0.826**	0.768	**0.869**	0.764	0.961	**0.993**
Fuzzers	1360	0.204	**0.600**	0.563	**0.633**	0.589	0.882	**0.928**

Rot-AUDF exceeds Isolation Forest in the two unlabelled settings on these subsets. With labels, Random Forest remains stronger.

Conclusion

1. A unary tree integrates uniform background loss exactly through leaf volumes.
2. Closed-form BCE leaves yield the explicit density $\hat{f}_A = \mu_A / V_A$ and an auditable partition.
3. Rotation forests improve density fit on several structured benchmarks and provide useful rarity rankings.
4. The UNSW pilot supports further validation as a security scoring component, with representation and deployment conditions still decisive.

Thank you. Questions?

`github.com/NTAILab/audt_ias`

What the model exposes to an analyst

A leaf records its feature bounds, volume V_A, sample count n_A, and density $n_A/(nV_A)$.

- A low score can be traced to a broad, weakly populated region.
- Original-feature intervals are immediate for an unrotated tree.
- Rotated paths remain auditable through stored transforms, but are harder to read.

Computational shape

A straightforward node scan is $O(pn_A \log n_A)$ for p candidate features. Constant-leaf prediction takes $O(\text{depth})$ per tree.

Synthetic density protocol

- Train on 2,000 observations per synthetic distribution, scaled into $[0, 1]^d$.
- Compare histogram, KDE, kNN, unary MLP, single AUDT, and rotation forest.
- Integrate grid KL numerically; lower is better.
- Gaussian 5D is excluded from grid KL: its 50^5 evaluation grid has 312.5 million points.

Reading the MLP comparison

Published MLP results and the authors' lightweight reproduction are separate entries. Their implementation settings may differ.

Synthetic anomaly benchmark: what is tested

Normal samples follow conditional interval rules. Anomalies include invalid rule configurations, boundary cases, and local shifts.

- Four tasks with 8, 20, 50, and 100 features.
- Training anomaly rate: 5%; test anomaly rate: 20%.
- Means over three independently generated train/test splits.
- Unlabelled models see the contaminated stream without labels; labelled models use class information.

This is a controlled test of rare conditional-region recovery, not a deployment simulation.

Labelled synthetic anomaly scoring

Mean AUROC $\pm$ standard deviation; higher is better.

Dataset	d	Unary DR	Unary Joint	MLP-DR	RF	IF, normal
Conditional cells	8	0.958 ± 0.009	0.955 ± 0.007	0.975 ± 0.010	0.975 ± 0.004	$0.903 \pm .005$
Conditional cells	20	0.750 ± 0.111	0.709 ± 0.074	0.951 ± 0.004	0.890 ± 0.018	0.637 ± 0.015
Sparse conditional	50	0.900 ± 0.012	0.890 ± 0.012	0.930 ± 0.016	0.827 ± 0.022	0.568 ± 0.017
Sparse conditional	100	0.901 ± 0.012	0.885 ± 0.012	0.861 ± 0.034	0.762 ± 0.007	0.535 ± 0.010

Unary DR leads at 100 dimensions. The large variability at 20 dimensions cautions against a broad superiority claim.