WHEN THE CHANGE CARRIES NO ENERGY: A DETECTABLE REGIME CHANGE THAT NO LEARNED MONITOR DETECTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Latent world models need reliable regime detection, yet existing monitors track only residual energy-shifts in the mean or variance of prediction error. When an environment switches between noise laws of identical variance, every prediction error keeps its size and only its shape changes, so an energy-based monitor, sensitive only to the second moment, is provably blind to the shift, a claim we verify-along with our other guarantees-in a machine-checked proof. We introduce a generative world model that pairs a joint-embedding predictive architecture with a conditional flow-matching predictor over future latent trajectories (F-JEPA-RD), which scores the full predictive density rather than its energy. An analytic Gaussian base coupled with a learned residual yields exact trajectory density ratios while preserving the representation structure the joint-embedding design protects. An audit of sequential detection benchmarks reveals a pervasive validity failure: standard protocols mistake a detection statistic's natural drift during calm periods for genuine detection, creating an illusion of success across existing methods. Under a drift-robust protocol with null controls, an oracle given the true noise law detects variance-matched shifts almost perfectly, while every learned monitor we test-including our own model's readouts-collapses to chance, on both synthetic benchmarks and a four-year real-world recording. A diagnostic locates the loss: the encoder preserves and even sharpens the shape signature, which is then diluted by the multi-lag residual the readouts consume. Applying the same shape core to a single-step innovation residual instead recovers part of the lost power in a single-seed pilot, indicating the information is still present in the representation. When regime changes instead alter the underlying dynamics, F-JEPA-RD's density ratio retains strong power while staying silent under null controls, exactly as the theory requires. We release the benchmark, proofs, and scoring code for independent re-scoring.

1 INTRODUCTION

Sequential regime detection asks a monitor to raise an alarm soon after a stream switches from a calm regime to a different one, and rarely before. When the data are high-dimensional and the regimes have no closed-form density, the common recipe is to learn a representation and a predictor on calm data and to watch the predictor's residuals with a sequential test: CUSUM and its generalized-likelihood variants (Page, 1954; Lorden, 1971; Siegmund & Venkatraman, 1995; Lai, 1998), Shiryaev-Roberts procedures (Shiryaev, 1963; Roberts, 1966), or anytime-valid e-processes (Shin et al., 2023; Ramdas et al., 2023). The alternatives these tests are built around are statements about *energy*: the residual acquires a mean, or its variance grows.

Some regime changes carry no energy at all. A switch from Gaussian innovations to a variance-matched mixture, the signature of intermittent volatility or of a process that alternates between quiet and noisy sub-states, leaves every first and second moment of the residual unchanged. We prove that every statistic that is a function of the residual energy then has identical expected evidence before and after the change (Proposition 3); kernel two-sample statistics (Gretton et al., 2012; Li et al., 2019) can see such changes in principle but need many samples per block. To notice them a predictor must model the *law* of the future, not only its mean. Joint-embedding predictive architectures (JEPAs) predict in latent space (LeCun, 2022; Assran et al., 2023; Bardes et al., 2024), but their predictors

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

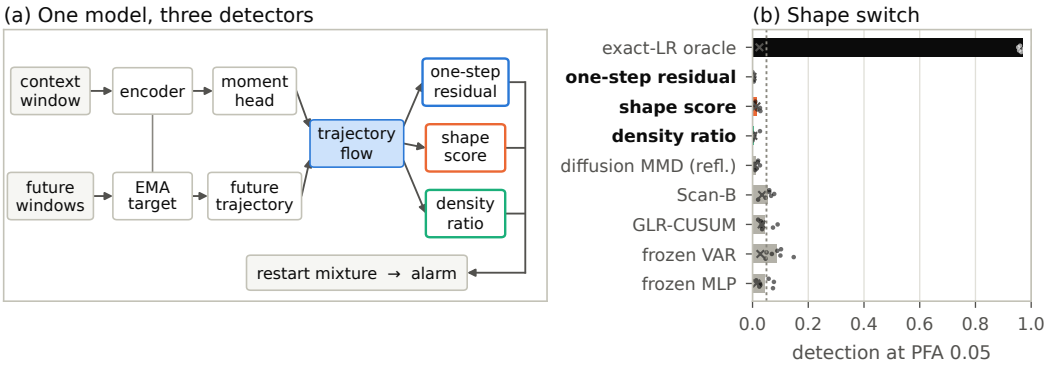


Figure 1: (a) F-JEPA-RD: a JEPA whose predictor is a conditional flow over a short trajectory of future latent codes, with an analytic Gaussian base (moment head) and a learned residual. One trained model yields three detectors, all aggregated by the same restart mixture. **(b)** On a regime change that alters only the *shape* of the innovations (Section 4.1), an exact-likelihood oracle on the true latent state detects almost every episode, while every learned detector - ours and the baselines - stays at chance once the alarm rule is made drift-robust (Section 4.2). Bars: mean over 7 seeds (aligned protocol, calm boundary, config B; density ratio from config A); dots: seeds.

are deterministic; the recent Flow-JEPA replaces the point predictor by conditional flow matching over future latent states (Huo et al., 2026). A conditional flow is a density, so the question becomes concrete: *can a flow-matching JEPA see what energy detectors cannot, and does the flow head leave the representation usable by the standard detectors?*

We answer with F-JEPA-RD (Figure 1). A causal encoder summarizes the recent past; an exponential-moving-average target encoder embeds short future windows into a trajectory of latent codes; and a conditional flow models that trajectory. The flow's velocity is the closed-form optimal-transport velocity of a Gaussian whose moments come from a small moment head, plus a learned residual that attends to the full context. The analytic base gives an exact inverse and an exact density; the residual adds what the base cannot express. From one trained model we read out three detectors: the model's own one-step residual (RESID), a shape score evaluated where the base sends the future trajectory back to its source (SHAPE), and the full trajectory density ratio against a linear-Gaussian null computed with the probability-flow ODE (RATIO).

Our answer has two halves, and the second was a surprise. Under the evaluation rule we pre-registered - the rule this literature uses - all three readouts detected the shape-only change at about the level of an exact-likelihood oracle, and so did a concurrent diffusion-latent detector (Kraevskiy & Prokhorov, 2026) reimplemented on the same latents. None of them was detecting it. The rule calibrates its alarm threshold on a calm *pre-change* window and then watches a window twice as long, so a statistic whose calm path drifts upward crosses the threshold for no reason but the extra time; every learned readout here drifts, because its null is a fitted model rather than the truth. Scored on calm test episodes containing no change, the same rule returns the same detection rates (Section 4.2). We reproduce the effect on a random walk with no model in it, and report every rate in this paper beside its calm-test twin under two drift-robust rules.

With the rule repaired the result is sharp. The energy-free change is detectable: the exact oracle finds it at 0.972 with a calm-test rate of 0.025 and a mean delay of 18.1 steps. Nothing we evaluated finds it: our three readouts, the kernel and CUSUM baselines, the diffusion detector and Bayesian online change-point detection (Adams & MacKay, 2007) all sit at chance (0.003-0.105). The obstruction is not where it appears to be: the embedding *keeps* the change, and the residual our readouts consume dilutes it (Section P). A single-seed pilot applying the same shape core to the one-step code innovation detects at 0.297 with a calm-test rate of 0.037, so the information was reachable all along. The same holds on a real four-year recording with the change injected at exactly constant realized innovation energy (Section R). When the change does alter the dynamics, learned detection works and the theory's predictions hold: RATIO reaches 0.626 against 0.410-0.660 for the kernel detector, and on a linear control that its null already explains it is silent under both drift-robust rules (0.000), as

Proposition 1 leads one to expect - while the pre-registered rule credits it 0.558 there, another instance of the artefact. Our contributions are:

- **A detection-oriented flow-matching JEPA** with an analytic base and residual that yields three complementary detectors from one model, with an exact base inverse, an exact base density and an exact-trace probability-flow density (Section 3).
- **Machine-checked theory.** The flow contains the linear-Gaussian null, representations cannot create evidence, energy statistics are blind to variance-matched changes, the shape score is not, and flexible predictors absorb evidence: 78 Lean theorems on Mathlib, plus 405 machine-checked means, with no `sorry` (Section 3.3, Section B).
- **A validity failure in the standard evaluation, and its fix.** The usual detection-rate protocol credits calm drift as detection. We demonstrate this without any model, and supply the calm test plus two drift-robust alarm rules that restore nominal behaviour for every detector (Section 4.2, Section H).
- **A controlled head-to-head** over two regime types, two training configurations, seven seeds, two bookkeeping protocols and three alarm rules, in which every detector, including the concurrent diffusion detector, is scored on identical latents, and whose raw monitoring paths we release so that any future rule can be applied to them without retraining (Section 4).
- **A limit on predictive embeddings.** An energy-free change that an oracle detects almost always defeats every learned representation we tested, on synthetic and on real backgrounds, and we localize the loss: the encoder preserves and even sharpens the fourth-moment signature, which is then diluted by the multi-lag residual the readouts are computed on (Section P, Section 4.3, Section R).

2 Related work and background

Sequential change detection. Classical procedures accumulate log-likelihood ratios between a known pre-change law and a post-change law that is known (CUSUM, Shiryaev-Roberts) or estimated on a window (GLR); their optimality theory is well developed (Page, 1954; Shiryaev, 1963; Roberts, 1966; Lorden, 1971; Pollak, 1985; Moustakides, 1986; Lai, 1998; Tartakovsky et al., 2014). Mixture and e-process constructions give anytime-valid versions (Robbins, 1970; Shin et al., 2023; Ramdas et al., 2023). Nonparametric detectors compare windows with kernel two-sample statistics (Harchaoui et al., 2008; Gretton et al., 2012; Li et al., 2019; Chang et al., 2019) or density ratios (Liu et al., 2013); Bayesian online detection tracks run lengths (Adams & MacKay, 2007). Surveys and benchmarks (Aminikhanghahi & Cook, 2017; Truong et al., 2020; van den Burg & Williams, 2020) and reviews of deep anomaly detection (Pang et al., 2021; Ruff et al., 2021) cover the wider landscape. Learned detectors monitor contrastive codes (Deldari et al., 2021), latent stochastic differential equations (Ryzhikov et al., 2023), neural CUSUM statistics (Gong et al., 2022) or predictive-coding uncertainty (Hup et al., 2024). What almost all of these share is an alternative class built on a residual's mean and scale; our shape score and density ratio are built on a learned conditional law instead.

JEPAs and generative latent prediction. JEPAs learn by predicting latent codes of unseen parts of the input from visible ones, using an exponential-moving-average target and anti-collapse regularization (LeCun, 2022; Assran et al., 2023; Bardes et al., 2024; Grill et al., 2020; Bardes et al., 2022). Latent predictive learning has been adapted to time-series anomaly prediction (He et al., 2026). Flow-JEPA (Huo et al., 2026) replaces the deterministic transition of a JEPA world model with conditional flow matching that jointly generates future latent states conditioned on observations and actions. We adapt this idea to a setting without actions and use the flow as a density for detection rather than for control.

Flows and densities. Flow matching and the probability-flow ODE give exact densities up to discretization; we score density *ratios* against a calm null and a source-shape score rather than raw likelihoods, which are unreliable out-of-distribution scores (Nalisnick et al., 2019). An extended discussion is in Section U.

Concurrent work. Kraevskiy & Prokhorov (2026) map pre-change observations to a Gaussian through the probability-flow ODE of a conditional diffusion model and monitor a maximum mean

discrepancy with Shiryaev-Roberts aggregation; Moushegian et al. (2025) derive diffusion-based hypothesis tests and change-point rules. Both use diffusion maps with divergence statistics; we use a flow-matching JEPA, three readouts of one conditional trajectory law, and a controlled comparison on identical representations (Section F).

Evaluation protocol. Every detector turns an episode into a path M_t (Section 3.2). Thresholds are set from calm calibration episodes; the *achieved* probability of false alarm (PFA) is the fraction of calm holdout episodes whose path crosses the threshold in the pre-change window; the *detection rate* is the fraction of test episodes whose first crossing occurs after the change at ν; delays are measured from ν. We report the detection rate interpolated at achieved PFA 0.05 and check that 0.05 lies inside the achieved range for every run. Because a false alarm before ν counts as a miss, detection rates of strong detectors saturate near $1 - \mathrm{PFA}$, where delays, not rates, separate them (Section G).

3 METHOD

3.1 A TRAJECTORY-FLOW JEPA

Let $x_t \in \mathbb{R}^D$ be the observation at time t and $X_t = x_{t-63:t}$ a window of 64 frames. A causal dilated convolutional encoder (Bai et al., 2018) maps the window to a code $z_t = f_\theta(X_t) \in \mathbb{R}^d$ with $d = 8$. A target encoder f_ξ, an exponential moving average of f_θ (Grill et al., 2020), embeds future windows, and the prediction target is the *trajectory*

$$y_t = \big(\bar{z}_{t+1}, \bar{z}_{t+2}, \bar{z}_{t+4}, \bar{z}_{t+8}\big) \in \mathbb{R}^{4d}, \qquad \bar{z}_{t+\delta} = f_\xi(X_{t+\delta}). \tag{1}$$

Each target window ends later than the context window and depends on frames the context has not seen. The predictor is a conditional velocity field on $\mathbb{R}^{4d}$ with two parts. A moment head reads the code, $(m_t, \log s_t^2) = h_\psi(z_t)$, and defines the closed-form velocity that carries $\mathcal{N}(0, I)$ to $\mathcal{N}(m_t, \operatorname{diag} s_t^2)$ along the monotone optimal-transport path $x_\tau = (1-\tau)x_0 + \tau(m_t + s_t \odot x_0)$:

$$v_t^\star(x, \tau) = m_t + \frac{s_t - 1}{(1-\tau) + \tau s_t} \odot (x - \tau m_t). \tag{2}$$

A residual network $r_\phi(x, \tau, c)$ corrects it, where $c = \mathrm{Attn}\big(q(x,\tau), H_t\big)$ is a cross-attention readout (Vaswani et al., 2017) from the query point to the encoder's per-frame trunk features H_t, so the residual sees the full context configuration rather than a pooled vector. The velocity is $v_t = v_t^\star + r_\phi$, trained by conditional flow matching (Lipman et al., 2023; Liu et al., 2023) with variance-covariance regularization of the code (Bardes et al., 2022):

$$\mathcal{L} = \mathbb{E}_{t,\tau,x_0} \big\| v_t\big((1-\tau)x_0 + \tau y_t, \tau\big) - (y_t - x_0) \big\|^2 + \lambda_{\mathrm{var}} \mathcal{L}_{\mathrm{var}}(z) + \lambda_{\mathrm{cov}} \mathcal{L}_{\mathrm{cov}}(z), \tag{3}$$

with $x_0 \sim \mathcal{N}(0, I)$, $\tau \sim \mathcal{U}[0, 1]$. The last layers of h_ψ and r_ϕ start at zero, so training starts from the identity flow and "no residual" is realizable. The moment head has no loss of its own: it learns only through Equation (3). We use two configurations that differ only in the variance weight, $\lambda_{\mathrm{var}} = 1$ (config A) and 25 (config B); training details are in Section D.

Why an analytic base. Learning the moments directly as the flow (a "naive" head) trains an equally good one-step predictor but destroys the linear predictability of the code that standard detectors rely on (Section 4.5). The base also gives two exact objects for free: the *inverse base point* $u_t = (y_t - m_t) \oslash s_t$ and the *base density* $\log p_t^{\mathrm{base}}(y) = \sum_i \log \varphi(u_{t,i}) - \sum_i \log s_{t,i}$ (✓ `logGauss_affine`, ✓ `inverse_point`).

Full density. With the residual, the density follows from the probability-flow ODE and the instantaneous change of variables (Chen et al., 2018; Grathwohl et al., 2019):

$$\log p_t^{\mathrm{flow}}(y) = \log \varphi\big(x(0)\big) - \int_0^1 \operatorname{tr} \partial_x v_t\big(x(\tau), \tau\big)\, d\tau, \qquad x(1) = y, \;\; \dot{x} = v_t(x, \tau), \tag{4}$$

integrated backward with 16 Euler steps and the exact trace ($4d$ vector-Jacobian products per step). For the base alone the divergence integrates to $\sum_i \log s_{t,i}$ exactly (✓ `integral_div_baseVel`), so Equation (4) reproduces the closed form up to discretization; the residual adds a flow-matching approximation error that we do not certify.

3.2 THREE DETECTORS FROM ONE MODEL

Each detector turns an episode into increments ℓ_t and aggregates them with the same restart mixture

$$M_t = \log \sum_{k \le t} w_k \exp\Big(\sum_{j=k}^{t} \ell_j\Big), \qquad w_k \propto \rho^k,\ \rho = 0.95, \tag{5}$$

whose weights sum to one (✓ `geometric_weights_sum`). All detectors are calibrated on calm episodes only.

RESID (one-step residual). The first block of m_t predicts $\bar{z}_{t+1}$; the residual $\bar{z}_{t+1} - m_t^{(1)}$ is scored by the conjugate-mixture e-process of the frozen detector (a Gaussian mean-shift mixture crossed with a variance grid, Section E) with a covariance fitted on calm residuals, never the model's own s_t.

SHAPE (source-shape score). At the inverse base point, $\ell_t = \sum_{i=1}^{4d} \big[\log t_5(u_{t,i}) - \log \varphi(u_{t,i})\big]$: how much better a heavier-tailed source explains the standardized future than the Gaussian source the model was trained with.

RATIO (trajectory density ratio). $\ell_t = \log p_t^{\text{flow}}(y_t) - \log p^{\text{VAR}}(y_t \mid z_t)$, where p^{VAR} multiplies the lag laws $\mathcal{N}\big(\mu + \Phi^\delta(z_t - \mu),\ \Sigma_\delta\big)$, $\Sigma_\delta = \sum_{k<\delta} \Phi^k S \Phi^{k\top}$, of a VAR(1) fitted on calm target codes (a composite likelihood over lags; Section E).

3.3 WHAT THE THEORY GUARANTEES

All statements below are proved in Lean 4 on Mathlib (de Moura & Ullrich, 2021; The mathlib Community, 2020); the checkmarks name the theorems (Section B maps each to its file and lists the axioms used).

Proposition 1 (The flow contains the null)**.** *If $\Phi = \operatorname{diag}\phi$ and $S = \operatorname{diag}\sigma^2$, the lag covariances are diagonal with entries $\sigma_i^2(1-\phi_i^{2\delta})/(1-\phi_i^2)$, and if the moment head outputs the lag means and variances with a zero residual, then $\log p^{\text{flow}} - \log p^{\text{VAR}} \equiv 0$; the resulting restart mixture never exceeds 0.* ✓ `lagCov_diagonal_closed`, ✓ `flow_LR_zero_diag_ode`, ✓ `restartMixture_inert_le_zero`

Proposition 2 (No created evidence)**.** *Let P, Q be the calm and stress laws of the latent path, κ any observation channel, f any measurable representation and g any statistic. Then $\mathrm{KL}\big((g\circ f)_\#\kappa P \,\|\, (g\circ f)_\#\kappa Q\big) \le \mathrm{KL}\big(f_\#\kappa P \,\|\, f_\#\kappa Q\big) \le \mathrm{KL}(P \,\|\, Q)$; and an exact log-likelihood-ratio increment has non-positive drift under the calm law.* ✓ `evidence_chain_full`, ✓ `calm_drift_nonpos`

Proposition 2 fixes the interpretation of every gain we report: a detector does better only by *aligning* its alternative class with evidence the representation already carries.

Proposition 3 (Energy statistics are blind to variance-matched changes)**.** *Let $Q = \frac{1}{2}\mathcal{N}(0, a^2) + \frac{1}{2}\mathcal{N}(0, b^2)$ with $(a^2 + b^2)/2 = \sigma^2$. For all α, β, $\mathbb{E}_Q[\alpha + \beta e^2] = \mathbb{E}_{\mathcal{N}(0,\sigma^2)}[\alpha + \beta e^2]$. In particular every variance-grid alternative γ of the frozen e-process has the same expected log-evidence, $-\frac{1}{2}(\log\gamma + 1/\gamma - 1) \le 0$, before and after the change. The benchmark's $a = \sigma/2$, $b = \sqrt{7}\,\sigma/2$ is variance matched with kurtosis $75/16$.* ✓ `energy_blind`, ✓ `varGrid_identical_under_both`, ✓ `varGrid_expected_value`, ✓ `shape_mixture_kurtosis`

Proposition 4 (The shape score is not blind)**.** $\log t_\nu(u) - \log\varphi(u) = c_\nu + g_\nu(u^2)$ *with* $g_\nu(w) = \frac{w}{2} - \frac{\nu+1}{2}\log\big(1 + \frac{w}{\nu}\big)$. *The function g_ν decreases on $[0,1]$, increases on $[1,\infty)$ and satisfies $-\frac{w}{2\nu} \le g_\nu(w) \le \frac{w}{2}$; and for every variance-matched scale mixture Q with $(a^2+b^2)/2 = 1$, $\mathbb{E}_{\mathcal{N}(0,1)}\, g_\nu(U^2) \le \mathbb{E}_Q\, g_\nu(U^2)$.* ✓ `shape_score_decomp`, ✓ `shapeCore_strictMonoOn`, ✓ `shapeCore_bounds`, ✓ `shape_score_sees_scale_mixture`

The score rewards mass both near the centre and in the tails and penalizes the shoulders, the signature of a leptokurtic law (Figure 3c). For the benchmark's mixture the gain is 0.034 nats per standardized coordinate (numerical integration). The upper bound also shows that the score contains an energy term, which we return to in Section 4.5.

Proposition 5 (Absorption)**.** *Against a calm-fixed metric, a post-change energy ratio g yields at most $\frac{1}{2}(g - 1 - \log g)$ nats per coordinate and step, attained at the grid point $\gamma = g$; an*

alternative that has rescaled itself by g extracts nothing. A predictor that absorbs a fraction a of a mean shift δ leaves $(1-a)^2\delta^2/(2v)$ expected nats per step under $\mathcal{N}(\delta, v)$. ✓ `gridEvidence_le`, ✓ `gridEvidence_absorbed`, ✓ `meanShift_evidence`

A flexible conditional density that re-reads post-change context can move its location and scale toward the new regime and thereby absorb evidence; a one-step residual scored with a *calm-fixed* metric cannot. Proposition 5 therefore predicts that RESID can match or beat RATIO on energy-carrying changes, a prediction that holds for *delay* but not for power: under the drift-robust rule of Section 4.2, RESID has almost no power on the swirl (Table 1) because it is among the statistics that drift most, so the absorption advantage it enjoys in delay never converts into detections. The first prediction holds. The second holds where it is a statement about exact densities - the oracle detects the energy-free change and the energy statistics do not - but not for our learned readouts, whose shape channel never receives the change - not because the encoder discards it, which it does not, but because the multi-lag residual the score is computed on dilutes it (Section P). Proposition 2 is the reason this is possible without contradiction, and Section 4.2 is the reason we can state it: under the alarm rule we pre-registered, all three readouts appeared to detect the change, and so did a diffusion baseline, because the rule rewards a statistic that drifts.

4 EXPERIMENTS

4.1 SETUP

Benchmark. A latent state of three rotation planes, observed through a fixed random two-layer $\tanh$ network into 24 channels plus 32 noise channels ($D = 56$), runs for 300 steps with a change at $\nu = 150$. The *nonlinear swirl* makes the rotation angle depend on each plane's squared radius at unchanged stationary variance (✓ `stress_stationary_variance`); the *shape switch* replaces Gaussian innovations by the variance-matched mixture of Proposition 3 through a linear observation map; the *linear rotation* control jumps the rotation angle by one radian. Figure 3 and Section C give the construction, the constants and the derivations.

Protocol. Per seed we regenerate the observation map and all splits, including 150 test episodes with the change and 150 *calm test* episodes without; seeds 0-6. Detectors are calibrated on calm episodes only and scored on the same trained latents (Section G).

Detectors. Ours: RESID, SHAPE, RATIO (Section 3.2). Baselines, all on the same latents: frozen VAR and MLP e-processes, GLR-CUSUM (Siegmund & Venkatraman, 1995; Lai, 1998), Scan-B (Li et al., 2019), BOCPD (Adams & MacKay, 2007), and a reimplementation of the concurrent diffusion-latent MMD detector (Kraevskiy & Prokhorov, 2026). The *oracle* applies the exact conditional log-likelihood ratio of the generative model to the true latent path; it is a reference level, not a bound (Sections E, F and J).

4.2 DOES THE ALARM RULE MEASURE EVIDENCE?

A detection rate is a statement about a rule, and the rule this literature uses - the one we pre-registered - does not measure what its name says. It calibrates the alarm threshold on the running maximum of the statistic over the calm *pre-change* window and then counts a crossing anywhere in a window twice as long. For a statistic that is stationary under the calm law the two windows are exchangeable. For a statistic whose calm path drifts upward they are not: the longer window reaches a higher level for no reason except its length, and the rule credits that as a detection (Equation (6)).

The effect needs no model to exhibit. On simulated paths $M_t = \kappa t + \sigma W_t + \delta(t-\nu)^+$, in which δ is the only carrier of evidence, the rule reports detection 0.952 on episodes with $\delta = 0$ once the calm drift reaches $\kappa = 0.5$, and alarms inside 0.999 of calm episodes while its calibrated false-alarm rate reads 0.05 (Figure 6 and Table 6).

Learned readouts drift, because their null is a fitted model rather than the truth: up to $+22.6$ nats per step on calm holdout data, against -0.04 for the oracle and the frozen e-processes, which are exact likelihood ratios under their nulls and must fall (Proposition 2). A statistic that climbs under the null is not a likelihood ratio against the calm law, and Equation (6) says such a statistic will be

Table 1: Detection rate at achieved PFA 0.05 under the calm boundary (Section H), aligned protocol, means over 7 seeds; *calm* is the same rule on calm test episodes and should sit near 0.05. Config A/B differ only in the VICReg variance weight of the F-JEPA encoder (λ_{var}=1 / 25); every detector in a column is scored on the same trained latents. Bold: best non-oracle detector. "–": not run in that config.

	Nonlinear swirl				Shape switch			
	config A		config B		config A		config B	
Detector	det.	calm	det.	calm	det.	calm	det.	calm
One-step residual, RESID (ours)	0.040	0.006	0.036	0.009	0.009	0.005	0.003	0.000
Shape score (ours)	0.107	0.015	0.165	0.014	0.009	0.014	0.014	0.014
Trajectory density ratio (ours)	0.626	0.012	–	–	0.006	0.003	–	–
Diffusion latent MMD (reflected)	–	–	0.450	0.009	–	–	0.014	0.008
Scan-B (kernel MMD)	0.410	0.036	0.660	0.031	0.051	0.039	0.055	0.034
GLR-CUSUM	0.453	0.032	0.452	0.031	0.063	0.037	0.045	0.033
Frozen VAR e-process	**0.715**	0.027	**0.757**	0.032	**0.105**	0.028	**0.086**	0.028
Frozen MLP e-process	0.351	0.030	0.303	0.020	0.082	0.026	0.043	0.014
Exact-LR oracle (true latent)	0.987	0.042	0.987	0.042	0.972	0.025	0.972	0.025

credited with detections it has not made. The diagnostic is one line of extra bookkeeping: score every detector a second time on *calm test* episodes under the identical rule. On the shape switch our three readouts detect at 0.934-0.966 and report 0.943-0.960 where there is nothing to detect; the concurrent diffusion detector behaves the same way. Those rates measure drift, not change.

We therefore report two drift-robust rules throughout (Section H): a flat threshold calibrated on the *full calm window*, and a *calm boundary* $b_t = m_t + c\, s_t$ built from the per-time mean and spread on calm calibration episodes, which absorbs drift of any shape. Both restore calm-test rates to the nominal level for every detector, and both remove up to 0.957 from rates that the pre-registered rule had awarded. The main text reports the calm boundary; Tables 7 and 8 give all rules side by side, and the pre-registered numbers are kept in full in Section I, since they remain the correct output of the rule that produced them.

4.3 A CHANGE WITH NO ENERGY: THE SHAPE SWITCH

The change is there to be found. The exact-likelihood oracle on the true latent path detects it at 0.972 with a calm-test rate of 0.025 and a mean delay of 18.1 steps (Table 1, right; per-seed values in Table 9). Proposition 3 says no energy statistic can do this, and none does: Scan-B reaches 0.051, GLR-CUSUM 0.063, the frozen MLP e-process 0.082 and the frozen VAR e-process 0.105, each against a calm-test rate within a hundredth or two of itself (Table 1) - chance, by the only definition that matters here. BOCPD (Adams & MacKay, 2007) joins them at 0.032 (calm 0.031), which is the sharpest version of the point: its run-length posterior *does* respond to this mixture on raw data, and stops responding once the change has passed through the encoder (Section F).

Our density-aware readouts do not do it either. Under the corrected rule RESID, SHAPE and RATIO reach 0.003-0.014, indistinguishable from their own calm-test rates (0.000-0.014); the concurrent diffusion detector, whose pre-registered 0.767 was inflated by the same mechanism (calm test 0.768), reaches 0.014. Proposition 4 is a statement about an exact density under the true law, and it survives in the oracle; what fails is the path from the change to the learned density. Section P localizes the failure, and not where we expected: the encoder keeps the change (the one-step code innovation gains +0.361 in excess kurtosis, more than the observation innovation does), while the multi-lag residual the readouts consume gains only +0.052, prediction error diluting a fourth-moment signature by the fourth power of its share. A single-seed pilot pointing the same shape core at the code innovation reaches 0.297 (calm test 0.037).

4.4 A CHANGE IN THE DYNAMICS: THE NONLINEAR SWIRL

When the change does move energy, learned detection works and the ordering follows the alternative classes (Table 1, left). The oracle reaches 0.987 with a delay of 5.4 steps. Among learned detectors the frozen VAR e-process on our codes leads (0.715-0.757), with RATIO (0.626) and BOCPD (0.655)

next and the kernel, CUSUM and diffusion statistics behind them; the two drift-robust rules disagree about the order at the top, so we claim only that these are the strongest learned detectors here.

At the horizon an ablation prefers. The pre-registered lags are not the best choice: re-running the whole campaign at $(1, 2, 4)$ gives RATIO 0.818 against 0.682 for the frozen VAR, making it the strongest learned detector rather than the second (Table 26). That horizon was selected on this setting, so Table 1 stays the headline; Section 4.3 holds at both.

What the representation buys. Table 1 cannot compare representations - every column is a readout on the same codes. Two experiments do. Removing the analytic base takes the code's one-step predictability from 0.691 to 0.064 and the frozen VAR e-process from 0.500 to 0.002, while RESID is untouched. The base is what keeps a flow-trained code usable by detectors that never saw the flow.

4.5 CONTROLS, ABLATIONS, AND WHAT DID NOT WORK

Five further results are in the appendix. The linear-rotation control behaves as the theory demands, with RATIO silent (Section V, Proposition 1). The pre-registered bookkeeping additionally depresses the baselines through warm-up windows, separately from the alarm rule and in the same direction (Section G). The prediction horizon, fixed before measuring, is not optimal (Section N). The learned shape channel mixes shape with scale (Section L), and no method we tried succeeds on public TCPD series (Section S).

5 LIMITATIONS

What our own detectors do and do not do. Under a drift-robust alarm rule, none of our three readouts detects the energy-free change (Section 4.3); the positive results are confined to the change in the dynamics, where RATIO and an e-process on the same codes are the strongest learned detectors, and to the linear control, where RATIO's silence matches Proposition 1. Propositions 3 and 4 remain statements about exact densities under the true law; they are confirmed by the oracle, not by our model.

Scope of the evidence. All positive results are on controlled data with known generative laws; that is what makes an exact oracle and a provably energy-free change possible, but it is not evidence for any particular application. Section R injects the change into a real four-year recording at constant realized innovation energy: the conclusion survives, with the warning that the recording's own heavy tails hide our synthetic mixture even from an informed statistic. A public benchmark with naturally occurring shape-only changes remains future work, and our attempt at one failed (Section 4.5); the encoder is small and we did not study scaling. The prediction horizon was fixed before measuring and is suboptimal: RATIO is about twice as strong at lags $(1, 2, 4)$ and collapses at $(1, 2, 4, 8, 16)$ (Section N). We leave the main results at the pre-registered horizon, since re-running them at a horizon chosen by a post-hoc sweep is the selection our own calm test exists to catch.

What the shape score measures. Proposition 4 is proved for symmetric Gaussian scale mixtures in standardized coordinates. In the learned latent the shape switch barely reaches the inverse base point at all (Section L), so the score has little to work with; skewness changes and other shapes are not covered by the theory either.

Densities and metrics. The trajectory density is exact only up to Euler discretization and the unquantified flow-matching error of the residual; the exact trace costs one vector-Jacobian product per trajectory coordinate. The null inside RATIO is a composite likelihood over lags. At PFA 0.05 strong detectors saturate near $1 - \mathrm{PFA}$, so detection rates alone cannot separate them; delays can, but depend on the threshold. The oracle is a reference, not a bound, for this empirical metric.

The alarm rules. Our correction is a diagnosis plus two rules, not a theory of valid sequential testing; what each rule assumes, where the two disagree, and why an e-process on a correct null needs neither are set out in Section H.

Protocol and baselines. The aligned protocol and both drift-robust alarm rules were introduced after pre-registration, the rules in response to the failure they diagnose. They are post-hoc changes; we therefore report the pre-registered numbers in full beside them (Tables 10 and 11) rather than

replacing them. Seven seeds and two configurations do not exhaust training variability, and the diffusion baseline in particular varies strongly between trainings; our reimplementation of it (small MLP score network, 100-step variance-preserving schedule, 50 ODE steps) may understate a tuned version.

What the propositions cover, and what we did not test. Proposition 1 is an algebraic containment result whose hypotheses (diagonal Φ and S, exact lag moments, $r_\phi \equiv 0$) our benchmark does not satisfy, so the near-silence on the linear control is empirical rather than an instance of the theorem; and Proposition 3 constrains energy statistics only, not learned likelihood-ratio or segmentation statistics. Our negative result is thus about the monitors we evaluated. Two untested families - neural likelihood-ratio CUSUM (Gong et al., 2022) and contrastive predictive representations (Deldari et al., 2021) - are the natural next tests; Section O says what our diagnosis predicts for each.

What Lean certifies. The Lean development certifies the mathematical statements, not the code (Section B).

6 Conclusion

A monitor that scores the energy of its prediction errors cannot see a change that carries no energy, however good its representation. That is a statement about alternative classes and we proved it. It motivates a predictor that models the law of the future rather than its mean, and a flow-matching JEPA with an analytic Gaussian base provides one: from a single model come a one-step residual, a shape score, and a trajectory density ratio selective for structure a linear null cannot express - silent when the null already explains the change, and among the strongest learned detectors when it does not.

The energy-free change is where the interesting answer lies. It is detectable: an exact-likelihood oracle finds almost every instance, almost immediately. It is not detected: every learned monitor we built or reimplemented - flow readouts, kernel test, diffusion detector, Bayesian online change-point detection - sits at chance, on synthetic data and on a real recording with the change injected at exactly constant energy. The reason is subtler than a representation that forgets: the embedding keeps the change and even sharpens it, and the loss happens in the residual the readouts are computed on, where prediction error dilutes a fourth-moment signature. Changing the alternative class is necessary; it is not sufficient until the statistic is computed on a residual in which the change survives. A single-seed pilot suggests that needs no change of architecture.

That conclusion is only visible once the metric is right. The standard scoring rule credits a statistic that merely drifts upward, and under it every monitor we tested appeared to solve a problem none solves.

Impact statement

This work concerns statistical detection of regime changes in data streams. Better detectors of changes that do not increase the energy of a signal can help monitoring in engineering, health and environmental settings, where an unnoticed switch between regimes can be costly. The same tools could be used to monitor people or markets; we evaluate only on synthetic data and one public change-point collection, and we make no claim of readiness for any deployment. Detectors that raise alarms on distributional shape can also produce false alarms with confident-looking statistics, so any use should keep calibration on representative calm data and human review of alarms. We release code, results and proofs to make the claims checkable.

AI USE STATEMENT

In this work, generative AI tools were used to (i) create or modify visual figures based on our experimental results, (ii) edit English grammar, phrasing, and vocabulary, and (iii) assist in producing Lean formalizations for checking selected mathematical equations. The tools were not used to generate data, formulate the research questions or theoretical claims, design or run experiments, analyze or interpret experimental results, or make scientific decisions. They were also not used to replace the authors' judgment about the scientific content.

We reviewed all AI-assisted work. The experimental data and results were checked against the original runs and stored result files; AI-assisted figures were checked against those results; and the Lean formalizations were compiled and checked by the authors to ensure that they corresponded to the equations stated in the paper. AI-assisted language edits were manually reviewed for meaning and accuracy. The authors take responsibility for the final content of this work, including all text, claims, figures, and formalizations produced with AI assistance.

REFERENCES

Ryan Prescott Adams and David J. C. MacKay. Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*, 2007.

Michael S. Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *International Conference on Learning Representations*, 2023.

Samaneh Aminikhanghahi and Diane J. Cook. A survey of methods for time series change point detection. *Knowledge and Information Systems*, 51(2):339–367, 2017. doi: 10.1007/s10115-016-0987-z.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15619–15629, 2023. doi: 10.1109/cvpr52729.2023.01499.

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video. *arXiv preprint arXiv:2404.08471*, 2024.

Wei-Cheng Chang, Chun-Liang Li, Yiming Yang, and Barnabás Póczos. Kernel change-point detection with auxiliary deep generative models. In *International Conference on Learning Representations*, 2019.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley, 2 edition, 2006. doi: 10.1002/047174882X.

Leonardo de Moura and Sebastian Ullrich. The Lean 4 theorem prover and programming language. In *Automated Deduction – CADE 28*, Lecture Notes in Computer Science, pp. 625–635, 2021. doi: 10.1007/978-3-030-79876-5_37.

Shohreh Deldari, Daniel V. Smith, Hao Xue, and Flora D. Salim. Time series change point detection with self-supervised contrastive predictive coding. In *Proceedings of the Web Conference 2021*, pp. 3124–3135, 2021. doi: 10.1145/3442381.3449903.

Tingnan Gong, Junghwan Lee, Xiuyuan Cheng, and Yao Xie. Neural network-based CUSUM for online change-point detection. *arXiv preprint arXiv:2210.17312*, 2022.

Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. FFJORD: Free-form continuous dynamics for scalable reversible generative models. In *International Conference on Learning Representations*, 2019.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Zaïd Harchaoui, Eric Moulines, and Francis R. Bach. Kernel change-point analysis. In *Advances in Neural Information Processing Systems*, volume 21, 2008.

Yanan He, Yunshi Wen, Xin Wang, and Tengfei Ma. SC-JEPA: Stabilizing latent predictive learning for time-series anomaly prediction. *arXiv preprint arXiv:2602.04643*, 2026.

Georges Hébrail and Alice Bérard. Individual household electric power consumption. UCI Machine Learning Repository, 2006. 2,075,259 one-minute measurements, Dec. 2006 – Nov. 2010.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Yanchen Huo, Ziying Song, and Yadan Luo. Flow-JEPA: Flow matching for robust latent dynamics in JEPA world models. *arXiv preprint arXiv:2608.29029*, 2026.

Roelof G. Hup, Julian P. Merkofer, Alex A. Bhogal, Ruud J. G. van Sloun, Reinder Haakma, and Rik Vullings. Anomalous change point detection using probabilistic predictive coding. *arXiv preprint arXiv:2405.15727*, 2024.

M. F. Hutchinson. A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines. *Communications in Statistics – Simulation and Computation*, 18(3):1059–1076, 1989. doi: 10.1080/03610918908812806.

Marcel Kollovieh, Marten Lienen, David Lüdke, Leo Schwinn, and Stephan Günnemann. Flow matching with Gaussian process priors for probabilistic time series forecasting. In *International Conference on Learning Representations*, 2025.

Artem Kraevskiy and Artem Prokhorov. Change detection in probability flow ODE: Online testing in diffusion latent spaces. *arXiv preprint arXiv:2608.22807*, 2026.

Tze Leung Lai. Information bounds and quick detection of parameter changes in stochastic systems. *IEEE Transactions on Information Theory*, 44(7):2917–2929, 1998. doi: 10.1109/18.737522.

Yann LeCun. A path towards autonomous machine intelligence. OpenReview, version 0.9.2, 2022. `https://openreview.net/forum?id=BZ5a1r-kVsf`.

Shuang Li, Yao Xie, Hanjun Dai, and Le Song. Scan B-statistic for kernel change-point detection. *Sequential Analysis*, 38(4):503–544, 2019. doi: 10.1080/07474946.2019.1686886.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Song Liu, Makoto Yamada, Nigel Collier, and Masashi Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013. doi: 10.1016/j.neunet.2013.01.012.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

G. Lorden. Procedures for reacting to a change in distribution. *The Annals of Mathematical Statistics*, 42(6):1897–1908, 1971. doi: 10.1214/aoms/1177693055.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Sean Moushegian, Taposh Banerjee, and Vahid Tarokh. Diffusion-based hypothesis testing and change-point detection. *arXiv preprint arXiv:2506.16089*, 2025.

George V. Moustakides. Optimal stopping times for detecting changes in distributions. *The Annals of Statistics*, 14(4):1379–1387, 1986. doi: 10.1214/aos/1176350164.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2019.

E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1-2):100–115, 1954. doi: 10.1093/biomet/41.1-2.100.

Guansong Pang, Chunhua Shen, Longbing Cao, and Anton van den Hengel. Deep learning for anomaly detection: A review. *ACM Computing Surveys*, 54(2):1–38, 2021. doi: 10.1145/3439950.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.

Sebastian Pfister, Benjamin Holzschuh, and Nils Thuerey. StocBench: A benchmark for generative modeling of stochastic dynamics. *arXiv preprint arXiv:2608.22309*, 2026.

Moshe Pollak. Optimal detection of a change in distribution. *The Annals of Statistics*, 13(1):206–227, 1985. doi: 10.1214/aos/1176346587.

Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4), 2023. doi: 10.1214/23-sts894.

Kashif Rasul, Calvin Seward, Ingmar Schuster, and Roland Vollgraf. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8857–8868, 2021.

Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning*, 2015.

Herbert Robbins. Statistical methods related to the law of the iterated logarithm. *The Annals of Mathematical Statistics*, 41(5):1397–1409, 1970. doi: 10.1214/aoms/1177696786.

S. W. Roberts. A comparison of some control chart procedures. *Technometrics*, 8(3):411–430, 1966. doi: 10.1080/00401706.1966.10490374.

Lukas Ruff, Jacob R. Kauffmann, Robert A. Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G. Dietterich, and Klaus-Robert Müller. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5):756–795, 2021. doi: 10.1109/JPROC.2021.3052449.

Artem Ryzhikov, Mikhail Hushchyn, and Denis Derkach. Latent stochastic differential equations for change point detection. *IEEE Access*, 11:104700–104711, 2023. doi: 10.1109/access.2023.3318318.

Jaehyeok Shin, Aaditya Ramdas, and Alessandro Rinaldo. E-detectors: A nonparametric framework for sequential change detection. *The New England Journal of Statistics in Data Science*, pp. 229–260, 2023. doi: 10.51387/23-nejsds51.

A. N. Shiryaev. On optimum methods in quickest detection problems. *Theory of Probability & Its Applications*, 8(1):22–46, 1963. doi: 10.1137/1108002.

D. Siegmund and E. S. Venkatraman. Using the generalized likelihood ratio statistic for sequential detection of a change-point. *The Annals of Statistics*, 23(1):255–271, 1995. doi: 10.1214/aos/1176324466.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Alexander Tartakovsky, Igor Nikiforov, and Michèle Basseville. *Sequential Analysis: Hypothesis Testing and Changepoint Detection*. CRC Press, 2014. doi: 10.1201/b17279.

Yusuke Tashiro, Jiaming Song, Yang Song, and Stefano Ermon. CSDI: Conditional score-based diffusion models for probabilistic time series imputation. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

The mathlib Community. The Lean mathematical library. In *Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs*, pp. 367–381, 2020. doi: 10.1145/3372885.3373824.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024.

Charles Truong, Laurent Oudre, and Nicolas Vayatis. Selective review of offline change point detection methods. *Signal Processing*, 167:107299, 2020. doi: 10.1016/j.sigpro.2019.107299.

Gerrit J. J. van den Burg and Christopher K. I. Williams. An evaluation of change point detection algorithms. *arXiv preprint arXiv:2003.06222*, 2020.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

APPENDIX OVERVIEW

Section A Notation	Section I Full results, curves and statistics
Section B Lean formalization and proofs	Section J The exact oracle
Section C Benchmark construction	Section K The divergence correction
Section D Model and training details	Section L Shape specificity
Section E Detector definitions	Section Q Analytic-base ablation
Section F Baselines	Section S Public TCPD pilot
Section G Evaluation protocol	Section T Pre-registration and deviations
Section H Validity of the alarm rule	Section M Cost and discretization
Section R Real background, injected change	Section N Prediction horizon
Section P Where the fourth moment goes	Section Y Reproducibility
Section U Extended related work	Section V Controls figure
Section W What did not work	

A NOTATION

Symbol	Meaning
$x_t \in \mathbb{R}^{56}$, $X_t = x_{t-63:t}$	observation, context window of 64 frames
$s_t \in \mathbb{R}^6$	true latent state of the benchmark (three rotation planes)
$z_t = f_\theta(X_t) \in \mathbb{R}^8$	context code (online encoder)
$\bar{z}_{t+\delta} = f_\xi(X_{t+\delta})$	target code (EMA encoder), $\delta \in \{1, 2, 4, 8\}$
$y_t \in \mathbb{R}^{32}$	future trajectory Equation (1)
(m_t, s_t^2)	moment-head mean and diagonal variance of the Gaussian base
$v_t^\star$, r_ϕ, $v_t = v_t^\star + r_\phi$	base velocity Equation (2), residual, full velocity
$u_t = (y_t - m_t) \oslash s_t$	inverse base point
ℓ_t, M_t	per-step increment, restart-mixture path Equation (5)
$\nu = 150$, $T = 300$, $t_0 = 63$	change time, episode length, first full-context index
$\Gamma = \{1, 1.5, 2.5, 4\}$, $\tau^2 = 1$, $\rho = 0.95$	variance grid, mean-shift prior scale, restart decay
$\lambda_{\text{var}} \in \{1, 25\}$	VICReg variance weight (config A / config B)
$g_\nu(w) = \frac{w}{2} - \frac{\nu+1}{2}\log(1 + \frac{w}{\nu})$	shape core, $\nu = 5$
$d_{\text{IS}}(g) = g - 1 - \log g$	Itakura-Saito divergence of an energy ratio

B LEAN FORMALIZATION AND PROOFS

What Lean certifies (moved from Section 5). The Lean development certifies the mathematical statements of Section 3.3, not the code. The correspondence between the Lean objects and the implementation rests on inspection and on numerical tests of the same identities (Section B).

The Lean 4 project (`lean/FJEPA`, toolchain `v4.33.1`, Mathlib `v4.33.1`) contains 78 theorems in seven modules and a generated module of 405 arithmetic checks. `lake build` succeeds; every paper-facing theorem is followed by `#print axioms`, and all of them depend only on `propext`, `Classical.choice` and `Quot.sound`; `sorryAx` appears nowhere and the sources contain no `sorry`. Table 2 maps each statement in the paper to its theorem. The proofs are short enough to read in the sources; below we give the mathematical argument behind each proposition.

Base flow. On the path $x_\tau = (1-\tau)x_0 + \tau(m + sx_0) = \tau m + s_\tau x_0$ with $s_\tau = 1 - \tau + \tau s > 0$, the velocity is $m + (s-1)x_0 = m + \frac{s-1}{s_\tau}(x_\tau - \tau m)$, which is Equation (2). The divergence of $v^\star$ in x is $(s-1)/s_\tau = \frac{d}{d\tau}\log s_\tau$, so its integral over $[0, 1]$ is $\log s$, and $\log \varphi(x_0) - \log s = \log \mathcal{N}(x_1; m, s^2)$ by the change of variables for the affine map.

Proposition 1. Unrolling $z_n = \mu + \Phi(z_{n-1} - \mu) + \epsilon_{n-1}$ gives $z_n = \mu + \Phi^n(z_0 - \mu) + \sum_{k<n} \Phi^k \epsilon_{n-1-k}$, hence the lag means and $\Sigma_{h+1} = S + \Phi\Sigma_h\Phi^\top$. For diagonal Φ, S the covariance is diagonal with the geometric sum $\sigma^2(1-\phi^{2h})/(1-\phi^2)$. A diagonal multivariate Gaussian log-density is a sum of scalar ones; with $s_i = \sqrt{v_i}$ the base-flow density coincides with it term by term, so the ratio is zero. With zero increments the restart mixture is $\log \sum_{k \le t} w_k \le \log 1 = 0$.

Table 2: Paper statements and the Lean theorems that prove them.

Statement	Lean theorem(s)	Module
Base velocity generates the OT path	`hasDerivAt_interp`, `baseVel_on_path`	`BaseFlow`
Inverse base point	`inverse_point`, `inverse_point_unique`	`BaseFlow`
Exact base density	`logGauss_affine`, `sum_logGauss_affine`	`BaseFlow`
$\int_0^1 \operatorname{div} v^\star = \log s$; ODE = closed form	`integral_div_baseVel`, `ode_logdensity_base`, `sum_ode_logdensity_base`	`BaseFlow`
VAR lag moments, unrolling	`lagCov_succ`, `lagMean_succ`, `varPath_unroll`	`VARLag`
Diagonal closed form	`scalar_lagVar`, `lagCov_diagonal_closed`	`VARLag`
Diagonal Gaussian density factorizes	`inv_diagonal_pos`, `logGaussMV_diagonal`	`VARLag`
Proposition 1	`flow_LR_zero_diag`, `flow_LR_zero_diag_ode`, `restartMixture_inert_le_zero`	`VARLag`
Restart mixture weights, monotonicity	`geometric_weights_sum`, `restartMixture_mono`	`VARLag`
Proposition 2	`evidence_chain`, `evidence_chain_stat`, `evidence_chain_full`, `gibbs`, `calm_drift_nonpos`	`Information`
Mixture moments and kurtosis	`shape_mixture_variance`, `shape_mixture_fourth`, `shape_mixture_kurtosis`	`Shape`
Proposition 3	`integral_sq_gaussianReal`, `integral_sq_mixLaw`, `energy_blind`, `paper_mixture_matched`	`Shape`
Variance grid	`varGrid_logratio`, `varGrid_expected_value`, `varGrid_expected_nonpos`, `varGrid_identical_under_both`	`Shape`
Proposition 4, shape of g_ν	`shape_score_decomp`, `shapeCore_strictAntiOn`, `shapeCore_strictMonoOn`, `shapeCore_bounds`	`Shape`
Proposition 4, Jensen gain	`shapeCore_midpoint`, `integral_sq_scale`, `shape_score_sees_scale_mixture`, `shape_switch_instance`	`ShapeJensen`
Student-t_5 fourth moment	`student5_fourth_moment`	`Shape`
Proposition 5	`dIS_nonneg`, `gridEvidence_le`, `gridEvidence_at_g`, `gridEvidence_absorbed`, `meanShift_evidence`, `meanShift_evidence_antitone`	`Absorption`
Benchmark constants	`stress_stationary_variance`, `swirl_rho_stress_stable`, `swirl_sigma_stress_pos`, `shape_scales`	`Swirl`
Every reported mean	405 generated theorems `mean_k_...`	`Numbers`

Proposition 2. This is the data-processing inequality (Cover & Thomas, 2006) in Mathlib's measure-theoretic form, for Markov kernels (`klDiv_comp_right_le`) and for measurable maps (`klDiv_map_le`), applied twice through $\operatorname{map}(g \circ f) = \operatorname{map} g \circ \operatorname{map} f$. The drift statement is Gibbs' inequality $\int \log \frac{dP}{dQ}\, dP \geq 0$ for probability measures.

Proposition 3. For Mathlib's Gaussian measures $\int x^2\, d\mathcal{N}(0, v) = v$ follows from the variance and mean lemmas; the mixture integral splits linearly, $\int x^2\, dQ = (a^2 + b^2)/2$. Every variance-grid log-ratio is $-\frac{1}{2}\log\gamma + \frac{1-1/\gamma}{2\sigma^2}e^2$, an energy statistic, so it has the same expectation under both laws; that expectation is $-\frac{1}{2}(\log\gamma + 1/\gamma - 1) \leq 0$ because $\log\gamma \geq 1 - 1/\gamma$.

Proposition 4. The decomposition is algebra. $g'_\nu(w) = (w-1)/(2(\nu+w))$ gives the monotonicity; $0 \leq \log(1+y) \leq y$ gives the bounds. Midpoint convexity follows from $(1 + \frac{x+y}{2\nu})^2 \geq (1 + \frac{x}{\nu})(1 + \frac{y}{\nu})$. For $Q = \frac{1}{2}\mathcal{N}(0, a^2) + \frac{1}{2}\mathcal{N}(0, b^2)$ with $(a^2 + b^2)/2 = 1$, the Gaussian scaling law turns $\int g_\nu(x^2)\, d\mathcal{N}(0, a^2)$ into $\int g_\nu(a^2 z^2)\, d\mathcal{N}(0, 1)$, and pointwise midpoint convexity gives $g_\nu(z^2) \leq \frac{1}{2}[g_\nu(a^2 z^2) + g_\nu(b^2 z^2)]$; integrability follows from $|g_\nu(w)| \leq (\frac{1}{2} + \frac{1}{2\nu})w$.

Proposition 5. With $y = g/\gamma$, $\frac{1}{2}d_{\mathrm{IS}}(g) - [-\frac{1}{2}\log\gamma + \frac{1-1/\gamma}{2}g] = \frac{1}{2}d_{\mathrm{IS}}(y) \geq 0$, with equality at $\gamma = g$; at $g = 1$ every grid point has non-positive evidence. For the location statement the pointwise log-ratio is affine in x and its Gaussian expectation is $(1-a)^2\delta^2/(2v)$. Figure 2 plots these closed forms.

C Benchmark construction

The three regimes (full statement, moved from Section 4.1). A latent state $s_t \in \mathbb{R}^6$ made of three rotation planes evolves as $s_{t+1} = \rho R(\theta_t) s_t + \epsilon_t$ and is observed through a fixed random two-layer $\tanh$ network into 24 channels, concatenated with 32 channels of pure noise ($D = 56$). Episodes have 300 steps with a change at $\nu = 150$. We use three regime changes (Figure 3; constants and derivations in Section C). *Nonlinear swirl:* after the change the rotation angle depends on each plane's squared radius, while the contraction and noise are retuned so that the stationary variance

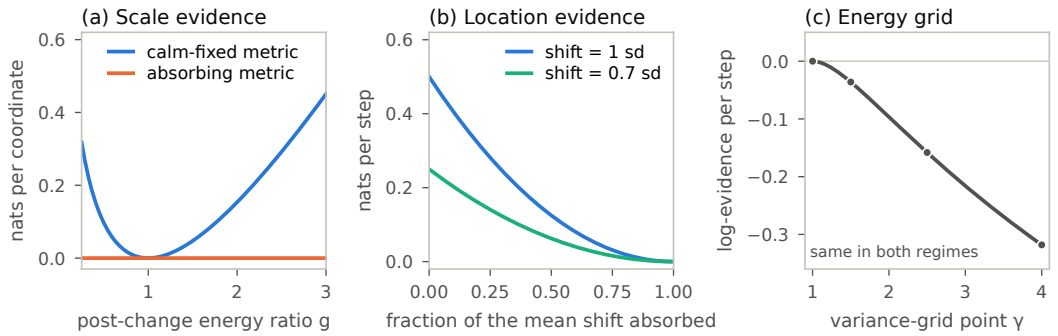


Figure 2: Closed forms of Propositions 3 and 5. **(a)** Scale evidence per coordinate and step against a calm-fixed metric ($\frac{1}{2}d_{\mathrm{IS}}(g)$) and against a metric that has absorbed the scale (0). **(b)** Remaining location evidence when a fraction of the mean shift is absorbed. **(c)** Expected log-evidence of the variance-grid points $\gamma \in \Gamma$ (dots) under a variance-matched change: identical before and after the change and never positive.

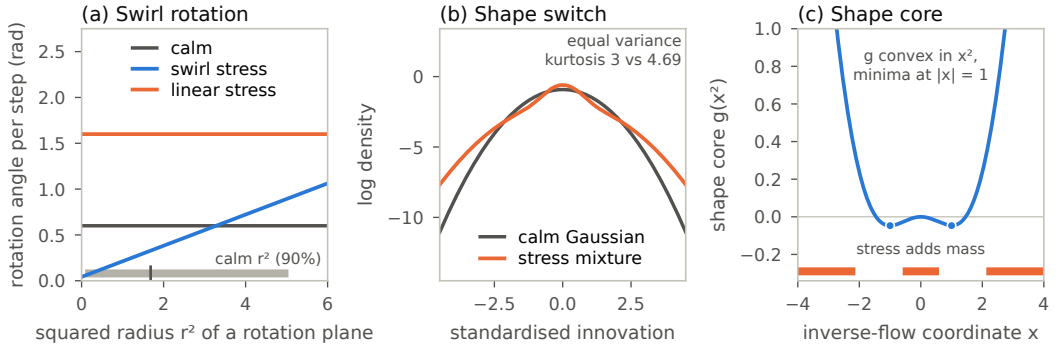


Figure 3: The benchmark and the shape score. **(a)** Swirl: after the change the rotation angle of a plane depends on its squared radius; the grey bar marks the central 90% of calm radii. **(b)** Shape switch: calm Gaussian and stress mixture have equal variance (Proposition 3) but different shape. **(c)** Shape core g_5 of Proposition 4: minima at $|x| = 1$; the orange rug marks where the stress law has more mass than the Gaussian (centre and tails), which is where g_5 is large.

is unchanged (✓ `stress_stationary_variance`). *Shape switch:* the innovations switch from Gaussian to the variance-matched mixture of Proposition 3; the observation map is linear so that the shape reaches the observations. *Linear rotation:* a control in which the rotation angle jumps by one radian, a change that a linear-Gaussian model expresses exactly.

Latent dynamics. $s_t \in \mathbb{R}^6$ consists of three planes. In the calm regime each plane evolves as $s_{t+1}^{(i)} = \rho R(\theta_0) s_t^{(i)} + \epsilon_t^{(i)}$ with $\rho = 0.9$, $\theta_0 = 0.6$, $\epsilon \sim \mathcal{N}(0, \sigma_s^2 I)$, $\sigma_s = 0.4$; the stationary per-coordinate variance is $c = \sigma_s^2/(1-\rho^2) = 16/19$ and the mean squared radius of a plane is $m = 2c$ (✓ `swirl_c`). Episodes start from the stationary law after a burn-in of 200 steps.

Nonlinear swirl. After ν, the angle of plane i is $\theta_t^{(i)} = \alpha + \kappa(\|s_t^{(i)}\|^2 - m)$ with $\kappa = 0.17$ and $\alpha = \theta_0 + \kappa m - 2\arctan(\kappa m)$; the contraction becomes $\rho_s = \rho(1 + (\kappa m)^2) \approx 0.97378$ and the noise variance $\sigma^2_{\mathrm{stress}} = c(1 - \rho_s^2)$, so the stationary variance is unchanged (✓ `stress_stationary_variance`, ✓ `swirl_rho_stress_stable`, ✓ `swirl_sigma_stress_pos`). The choice of α keeps the second-order statistics of the latent close to their calm values: in a 400k-step simulation of each regime (`swirl_moments.py`) the stationary variance changes by +0.6%, the lag-one autocovariance by -0.2%, and the fitted VAR(1) diagonal and rotation coefficients by -0.8% and +0.5%, leaving the state-dependence of the rotation as the main signal.

Shape switch. After ν the rotation and contraction are unchanged and each innovation coordinate is drawn from $\frac{1}{2}\mathcal{N}(0, a^2) + \frac{1}{2}\mathcal{N}(0, b^2)$ with $a = \sigma_s/2$, $b = \sqrt{7}\sigma_s/2$ (✓ `shape_scales`). The observation map is linear in this setting ($x = sW_1W_2 + b_1W_2 + b_2 +$ noise) because the tanh map

would compress the tails that carry the only signal. Empirically on the true latent the post/calm residual-energy ratio is 1.00 ± 0.01 and the kurtosis rises from 3.00 to 4.64-4.72 across seeds (Table 21), matching $75/16$.

Linear rotation. After ν the angle jumps to $\theta_0 + 1$, everything else unchanged.

Observation. In the swirl and linear-rotation settings, $x = \tanh(\tanh(sW_1 + b_1)W_2 + b_2) + \mathcal{N}(0, 0.15^2)$ with random $W_1 \in \mathbb{R}^{6\times32}$, $W_2 \in \mathbb{R}^{32\times24}$ (entries $\mathcal{N}(0, 1.6^2/\text{fan-in})$, biases $\mathcal{N}(0, 0.1^2)$), drawn once per seed, concatenated with 32 channels of $\mathcal{N}(0, 1)$ noise. Because each seed draws a new map, seeds vary both data and the observation geometry.

D Model and training details

Table 3: Architecture and optimization (identical for all runs except λ_{var}).

Encoder f_θ	1×1 input convolution to width 64; 3 residual blocks of causal convolution (kernel 3, dilation 1, 2, 4), GroupNorm, GELU; linear readout of the last position to $d = 8$ (41.6k parameters)
Target f_ξ	EMA of f_θ, decay 0.996
Moment head h_ψ	MLP $8 \to 64 \to 64$ (GELU), outputs $(m, \log s^2) \in \mathbb{R}^{32\times2}$, last layer zero-initialized
Residual r_ϕ	query MLP $(x, \tau) \to 64$; 4-head cross-attention over 64 trunk tokens; MLP $(x, \tau, c) \to 64 \to 64 \to 32$, last layer zero-initialized (flow head total 44.4k parameters)
Loss	Equation (3); VICReg variance hinge at 1, covariance off-diagonal penalty, $\lambda_{\text{cov}} = 1$
Optimizer	AdamW (Loshchilov & Hutter, 2019), lr 10^{-3}, weight decay 10^{-4}, batch 256, 2000 steps, 5% warm-up, cosine decay, gradient clipping at 1
Model selection	every 250 steps: reject checkpoints whose code has median per-dimension std < 0.5 or effective rank < 4; among the rest minimize the normalized one-step linear prediction error of the code on validation episodes (patience 8)
Density	16 Euler steps of Equation (4), exact trace (32 VJPs per step)
Configs	A: $\lambda_{\text{var}} = 1$; B: $\lambda_{\text{var}} = 25$

Scoring every detector on one seed takes three to seven minutes, of which the exact-trace density ratio takes about three. Training on MPS is deterministic: an independent rerun of the whole config-A swirl campaign reproduced every per-seed value of every detector under both protocols exactly (zero mismatches at machine precision), and the original-divergence density ratio reproduces the stored pre-registered runs (Table 22).

E Detector definitions

The detector list (moved from Section 4.1). Ours: RESID, SHAPE, RATIO (Section 3.2). Baselines: the frozen VAR e-process and its nonlinear variant with an MLP Gaussian null, GLR-CUSUM with windows 10-80 (Siegmund & Venkatraman, 1995; Lai, 1998), the Scan-B kernel statistic (Li et al., 2019), and a reimplementation of the concurrent diffusion-latent MMD detector with Shiryaev-Roberts aggregation (Kraevskiy & Prokhorov, 2026). The *oracle* applies the exact conditional log-likelihood ratio of the generative model to the true latent path with the same restart mixture; it is a reference level, not a bound on detection rate at a fixed empirical PFA (Section J). Implementation details are in Sections E and F.

Frozen conjugate-mixture e-process. For residuals $r_j \in \mathbb{R}^d$ with calm covariance Σ, restart k, window length $n = t - k + 1$, $SS = \sum_{j=k}^{t} r_j^\top \Sigma^{-1} r_j$ and $Q = S^\top \Sigma^{-1} S$ with $S = \sum_{j=k}^{t} r_j$, the log-evidence of the alternative "$r_j \sim \mathcal{N}(\mu, \gamma\Sigma)$ with $\mu \sim \mathcal{N}(0, \tau^2\gamma\Sigma)$" against $\mathcal{N}(0, \Sigma)$ is

$$\log E_{k,t}(\gamma) = -\tfrac{nd}{2}\log\gamma + \tfrac{1}{2}SS\big(1 - \tfrac{1}{\gamma}\big) - \tfrac{d}{2}\log(1 + n\tau^2) + \frac{\tau^2 Q}{2\gamma(1 + n\tau^2)},$$

and the path is $M_t = \log \sum_{k\le t} w_k |\Gamma|^{-1} \sum_{\gamma\in\Gamma} E_{k,t}(\gamma)$. Its variance channel is exactly the grid statistic of Proposition 3. The *frozen VAR* detector applies it to the residuals of a VAR(1) fitted (ridge

$10^{-4}n$) on calm codes whitened with calm statistics (shrinkage 0.1); the *frozen MLP* detector replaces the VAR mean by a heteroskedastic Gaussian MLP null (two hidden layers of 128, 1500 steps, early stopping) and standardizes residuals by the predicted scale.

RESID. The residual $\bar{z}_{t+1} - m_t^{(1)}$ is scored by the same e-process with Σ the covariance of calm training residuals (plus $10^{-4}I$).

SHAPE. $\ell_t = \sum_{i=1}^{32} [\log t_5(u_{t,i}) - \log \varphi(u_{t,i})]$ at $u_t = (y_t - m_t) \oslash s_t$, aggregated by Equation (5). The standardized variant (Section L) replaces u_t by $(u_t - \hat{\mu}) \oslash \hat{\sigma}$ with calm per-coordinate statistics.

RATIO. $\ell_t = \log p_t^{\text{flow}}(y_t) - \sum_{\delta \in \Delta} \log \mathcal{N}\big(\bar{z}_{t+\delta}; \mu + \Phi^\delta(z_t - \mu), \Sigma_\delta\big)$ with the VAR(1) fitted on calm target-encoder codes and evaluated at the context code. The product over lags ignores the covariance between lags, so p^{VAR} is a composite likelihood; Proposition 1 holds for this object. Section E summarizes one detector pass.

Algorithm 1: scoring one episode with a native readout

Input: trained $(f_\theta, f_\xi, h_\psi, r_\phi)$, episode $x_{0:T-1}$, calm statistics, readout $\in \{\text{RESID}, \text{SHAPE}, \text{RATIO}\}$.
for $t = t_0, \ldots, T-9$: $z_t \leftarrow f_\theta(X_t)$; $(m_t, s_t) \leftarrow h_\psi(z_t)$; $y_t \leftarrow (f_\xi(X_{t+\delta}))_{\delta \in \Delta}$
RESID: $r_t \leftarrow \bar{z}_{t+1} - m_t^{(1)}$ SHAPE: $\ell_t \leftarrow \sum_i \log t_5(u_{t,i}) - \log \varphi(u_{t,i})$
RATIO: integrate $\dot{x} = v_t(x, \tau)$ from $x(1) = y_t$ back to $x(0)$ with 16 Euler steps, accumulating $\operatorname{tr} \partial_x v_t$;
$\ell_t \leftarrow \log \varphi(x(0)) - \int \operatorname{tr} \partial_x v_t - \log p^{\text{VAR}}(y_t \mid z_t)$
aggregate: M_t by the e-process (RESID) or Equation (5) (SHAPE, RATIO); aligned protocol: store M_t at the time of the latest observation used ($t+1$ or $t+8$).
Output: path $M_{t_0:T-1}$; thresholds come from calm calibration episodes (Section G).

Figure 4 shows all five statistics on one test episode of each setting, each shifted by its own alarm threshold at achieved PFA 0.05, so the alarm level is 0 for every curve. The figure separates the statistics into two groups, and the separation matters more than the post-change behaviour. The exact oracle and the two frozen e-processes drift *down* before the change, as Proposition 2 requires of an exact log-likelihood ratio under the null, and the oracle then rises sharply at ν. Our three learned readouts instead drift *up* throughout, calm data included: on the shape switch their mean per-step increment on calm holdout episodes, averaged over the seven seeds of config A, is $+2.05$ (one-step residual), $+16.29$ (density ratio) and $+22.61$ (shape score), against -0.04 for the oracle and -0.04 for the frozen VAR e-process. A statistic that climbs under the null is not a likelihood ratio against the true calm law, and Section H shows that this upward drift, not the change, is what carried most of their reported detection rate under the pre-registered alarm rule.

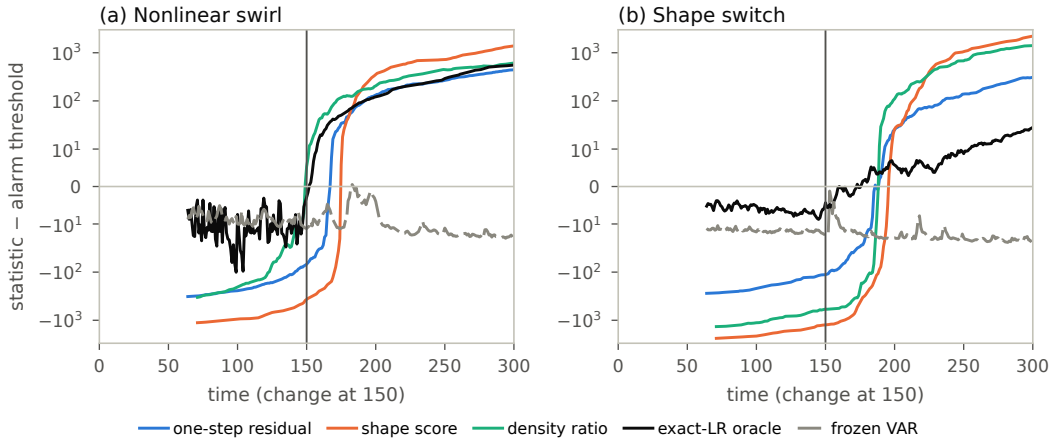


Figure 4: Detector statistics on one test episode (seed 0, config A, aligned labels), each minus its own threshold at achieved PFA 0.05; the vertical line is the change at $t = 150$ and the horizontal line the alarm level. Symmetric-log vertical scale.

F BASELINES

Bayesian online change-point detection. Adams & MacKay (2007) maintain the posterior over the *run length* - the time since the last change - under a constant hazard and a conjugate predictive, and are the natural Bayesian counterpart to the frequentist detectors here. We run BOCPD on the same whitened codes as the frozen e-processes, with a Normal-Inverse-Gamma predictive whose prior is matched to calm data, hazard $1/250$, and monitor $M_t = \log P(r_t < 20)$, the posterior belief that a change happened within the last twenty steps. Table 4 gives the result.

Two things are worth recording. First, BOCPD is *not* covered by Proposition 3, and we checked this rather than assuming it: on i.i.d. Gaussian data the statistic rises by $+0.99$ nats for a mean shift, $+0.89$ for a doubled variance and $+0.81$ for the variance-matched mixture of Proposition 3, against -0.30 under no change. Because it re-estimates the variance inside each candidate segment, a scale mixture produces stretches that genuinely look like new regimes; it is a segmentation model, not a fixed-reference energy statistic. Second, that ability does not survive the path to the code it is scored on: on the shape switch BOCPD detects 0.032 against a calm-test rate of 0.031. The reason is visible in Table 28. BOCPD's predictive is a per-coordinate location-scale model of the code *level* z_t, and the level gains only +0.105 in excess kurtosis after the change, whereas the one-step code *innovation* gains +0.361. A detector that can see this change in the raw signal loses it when it is applied to a quantity in which the change is diluted - which is the same mechanism that costs our own multi-lag readouts their power, and not evidence that the representation has discarded anything.

On the swirl BOCPD reaches 0.655, second among learned detectors, and on the linear-rotation control it fails completely (0.001) - a rotation leaves the per-coordinate marginal law almost unchanged, and BOCPD's predictive, unlike the frozen VAR's, carries no dynamics. Its calm drift is -0.005 nats per step, so unlike our readouts it is well behaved under the pre-registered rule too.

Is BOCPD scored on the campaign's latents? It is scored on a model retrained with the same seed and configuration, which is not the same statement as "the campaign's model", so Table 4 carries the check: the frozen VAR e-process is re-scored in the same run and compared with the value the campaign stored for that seed. Over 18 seed-setting pairs the two agree exactly in a third of cases and differ by 0.015 on average, 0.085 at worst. We chased the two obvious explanations and excluded both: running the identical script twice in separate processes reproduces every value bit for bit, so training on this device is deterministic; and perturbing the stored paths at the relative scale of the `float32` in which they are dumped moves no detection rate at all. What remains is that a rate *interpolated* at achieved PFA 0.05 moves in discrete jumps when the threshold grid shifts slightly, so two numerically near-identical runs can report values a few hundredths apart. BOCPD's numbers should therefore be read with that tolerance rather than as exact entries of Table 1; it does not affect the conclusions drawn from them, which are the difference between 0.03 and 0.66, not between 0.62 and 0.64.

Table 4: Bayesian online change-point detection (Adams & MacKay, 2007) on the same latent codes as the frozen e-processes (`bocpd_baseline.py`, 7 seeds). *calm* is the calm-test rate. The right-hand columns are a reproduction check: the frozen VAR e-process scored in this run against the value the campaign stored for the same seed, which is how one can see that the retrained model is the campaign's.

	pre-registered		calm boundary			frozen VAR check	
Setting	det.	calm	det.	range	calm	here	campaign
Shape switch	0.129	0.127	0.032	0.000–0.076	0.031	0.118	0.105
Nonlinear swirl	0.876	0.036	0.655	0.581–0.800	0.053	0.719	0.715
Linear rotation	0.005	0.036	0.001	0.000–0.007	0.053	0.973	0.973

GLR-CUSUM. On calm-VAR residuals of whitened codes, $\mathrm{GLR}_t = \max_{w\in\{10,20,40,80\}} \frac{w}{2}\big[\log\det\Sigma_0 - \log\det\hat{\Sigma}_w + \mathrm{tr}(\Sigma_0^{-1}S_w) - d\big]$, with S_w the second moment of the last w residuals about zero and $\hat{\Sigma}_w$ their covariance; this is the window-limited GLR for a Gaussian with unknown mean and covariance (Siegmund & Venkatraman, 1995; Lai, 1998).

Scan-B. Online Scan-B statistic (Li et al., 2019) with block size $B_0 = 20$, $N = 8$ fixed disjoint reference blocks from calm training codes, a Gaussian kernel with median-heuristic bandwidth and

the unbiased MMD^2 estimator; the variance normalization of the original is a constant for fixed B_0 and is absorbed by the empirical calibration.

Diffusion latent MMD (after Kraevskiy & Prokhorov (2026)). A conditional ϵ-prediction DDPM (Ho et al., 2020) (d-dimensional MLP with 128 hidden units and a learned time embedding, 52.9k parameters; linear β schedule from 10^{-4} to 0.02 over 100 steps; 2000 AdamW steps, batch 512) models $p(z_{t+1} \mid z_t)$ on whitened calm codes. The forward probability-flow ODE (50 Euler steps) maps each test code to the prior; the block MMD^2 (block 20, Gaussian kernel with the median distance of calm codes as bandwidth) against a fixed block of 20 standard-normal samples is accumulated by the reflected recursion $R_t = \max(0, R_{t-1} + \text{MMD}_t^2)$. Differences from the original: a small MLP score network, a fixed reference block, the reflected sum in place of their Shiryaev-Roberts statistic, and empirical calibration identical to all other detectors instead of their closed-form thresholds. The baseline is trained on the same latents as every other detector of its configuration. A stored run of the same code with a separate CUDA training of the encoder gave markedly lower values (Table 14); training variability of this baseline is large.

Exact-LR oracle. On the true latent path, swirl and linear rotation use $\log \mathcal{N}(s_{t+1}; \mu_{\text{stress}}(s_t), \sigma^2_{\text{stress}}) - \log \mathcal{N}(s_{t+1}; \mu_{\text{calm}}(s_t), \sigma^2_s)$; the shape switch uses the mixture-vs-Gaussian log-density of the calm-model innovation. Increments are aggregated by Equation (5).

G Evaluation protocol

Splits and scoring (full statement, moved from Section 4.1). Per seed we regenerate the observation map and all splits: 150 calm training, 30 validation, 200 calibration, 200 holdout, 150 test episodes containing the change, and 150 *calm test* episodes containing none; seeds 0-6. Each detector is calibrated on calm episodes only; every reported value has achieved PFA 0.05 inside its measured range. Within a configuration, all detectors are scored on the *same* trained latents. Statistics are stored at the time of their latest observation and restarted at the first full-context index, so that lookahead costs delay rather than being free (the *aligned* protocol of Section G; the pre-registered bookkeeping is reported beside it in Section I).

Curves. For a detector with paths $M_t^{(e)}$, let $W^{(e)} = \max_{t \in [12,150)} M_t^{(e)}$ on calibration episode e. For 40 levels $\alpha \in [0.005, 0.5]$ the threshold h_α is the $(1-\alpha)$ quantile of $\{W^{(e)}\}$; its achieved PFA is the fraction of holdout episodes with $W > h_\alpha$; on test episodes the first crossing $\tau_e = \min\{t \geq 12 : M_t^{(e)} > h_\alpha\}$ is a detection if $\tau_e \geq \nu$, with delay $\tau_e - \nu$. We interpolate detection rate and delay linearly in achieved PFA at 0.05, and report a value only if 0.05 lies inside the range of achieved PFAs (true for every value in this paper).

Chance level and saturation. A detector blind to the change behaves after ν as before it. If its calm crossing hazard is constant, calibration to PFA 0.05 over a pre-change window of L steps gives hazard $\lambda = -\log 0.95/L$, and its detection rate is about $0.95\,(1 - e^{-150\lambda})$: 0.09 for the aligned window and 0.05 for the original one. Conversely, a detector that catches every change is capped at one minus its false-alarm fraction on the test episodes. Because swirl and linear rotation share all calm data of a seed, saturated detectors have *identical* detection rates in the two settings while their delays differ (Table 5); this is why we report delays.

The aligned protocol as introduced in the setup. In the pre-registered (original) protocol, our readouts exist only from the first full-context index $t_0 = 63$, while the baselines run on latents computed from left-padded windows and are monitored from $t = 12$; several statistics also store at time t a value that uses later observations. The *aligned* protocol recomputes every statistic on $t \geq t_0$ data with nulls refitted there, restarts the mixture at t_0, and stores each value at the time of the latest observation it uses. The aligned protocol was introduced after pre-registration; it raises every baseline and changes our detectors by at most a few hundredths, so it is conservative for our claims. We report it in the main text and the original protocol on the same models in Section I.

Table 5: Saturation of the detection rate. Swirl and linear rotation share the calm training, calibration and pre-change test data of each seed; for detectors that catch every change, the detection rate equals one minus the pre-change false-alarm fraction and is identical across the two settings, while delays differ (config A, aligned, seeds 0–3).

Detector	Setting	s0	s1	s2	s3
One-step residual, RESID	swirl	0.940 (11.1)	0.993 (11.5)	0.965 (9.2)	0.962 (9.9)
One-step residual, RESID	lin. rot.	0.940 (11.4)	0.993 (10.9)	0.965 (8.7)	0.962 (9.3)
Shape score	swirl	0.973 (21.8)	0.968 (20.1)	0.929 (15.5)	0.953 (17.0)
Shape score	lin. rot.	0.973 (18.7)	0.968 (17.0)	0.929 (13.2)	0.953 (14.0)
Frozen VAR e-process	swirl	0.473 (52.2)	0.516 (48.5)	0.511 (52.5)	0.553 (50.5)
Frozen VAR e-process	lin. rot.	0.924 (5.4)	0.936 (4.5)	0.949 (4.3)	0.953 (4.8)

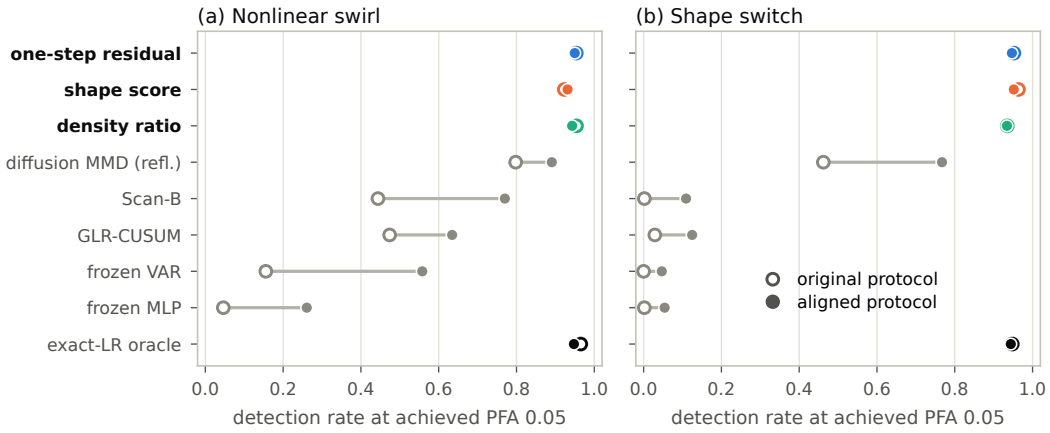


Figure 5: Original vs. aligned protocol on the same trained models (config B; density ratio from config A), both under the pre-registered alarm rule. Removing warm-up windows raises every baseline, by up to 0.51; our readouts move by at most 0.012. Figure 7 shows what happens to both groups when the alarm rule itself is corrected.

Protocol and configuration. Under the pre-registered bookkeeping the baselines are additionally depressed, some below chance, by warm-up windows in which left-padded latents inflate the calibrated threshold (Figure 5, Table 11). That effect is separate from the calibration artefact of Section 4.2 and points the same way: both flattered our readouts relative to the baselines. Our readouts are stable across configurations under the corrected rule (0.003-0.626), as is the oracle.

Original vs. aligned bookkeeping. In the original protocol (i) the frozen, GLR, Scan-B and diffusion statistics run on codes computed from windows left-padded with the first frame for $t < 63$, with nulls, whiteners and reference pools fitted on all time steps including the padded ones, and are monitored from $t = 12$; (ii) the native readouts exist from $t_0 = 63$ only; (iii) RESID, GLR-CUSUM and the diffusion statistic store at index t a value that uses the observation at $t + 1$, and SHAPE and RATIO one that uses observations up to $t + 8$. In the aligned protocol every statistic is recomputed on $t \geq t_0$ data with nulls, whiteners and reference pools refitted on $t \geq t_0$ calm data, the restart mixture restarts at t_0, and each value is stored at the time of its latest observation, so lookahead costs pre-change credit and delay instead of being free. The padded codes are atypical, inflate the calibration maxima of the baselines and push them below chance (Figure 5).

H VALIDITY OF THE ALARM RULE: THE CALM TEST

The protocol of Section G is the one we pre-registered and the one this literature reports. It has a failure mode that we did not anticipate and that invalidates part of our own headline claim, so we describe it, measure it, and re-evaluate every detector under rules that do not have it. Nothing is removed: the pre-registered numbers stay in Section I, next to the corrected ones.

The failure mode. The threshold h_α is calibrated on the running maximum over the *pre-change* window $[12, \nu)$, but a detection is a crossing anywhere in $[12, T)$, a window twice as long. For a statistic that is stationary under the calm law the two windows are exchangeable and nothing goes wrong. For a statistic whose calm path *drifts upward* they are not. Write $M_t = \kappa t + \varepsilon_t$ with $\kappa > 0$ and ε calm-stationary. Then $h_\alpha \approx \kappa\nu + q_{1-\alpha}$ where q is a quantile of the noise maximum, while the post-ν path reaches about κT. The crossing is therefore certain as soon as

$$\kappa\,(T - \nu) \;>\; q_{1-\alpha}, \tag{6}$$

which is a statement about the drift and the window lengths only - the change never enters it. A detector satisfying Equation (6) reports a high detection rate on episodes that contain no change at all, while its *calibrated* false-alarm rate is still 0.05, because that rate was measured on the shorter window.

Measuring it without a model. Figure 6 and Table 6 evaluate the three rules on paths simulated directly from $M_t = \kappa t + \sigma W_t + \delta\,(t - \nu)^+$, with no encoder, no detector and no data: δ is the only carrier of evidence and is zero in the calm arm. At $\kappa = 0.5$ the pre-registered rule reports detection 0.95 on episodes with $\delta = 0$ - as high as on episodes with $\delta > 0$ - and its probability of alarming somewhere inside a calm episode is 1.00 although it was calibrated to 0.05. The inflation is a property of the rule, not of any particular detector.

Two rules that do not have it. We re-evaluate every path under two alternatives. The *full-window* rule keeps a flat threshold but calibrates it on the whole calm episode $[12, T)$, the window that is actually monitored. The *calm-boundary* rule replaces the flat threshold by $b_t = m_t + c\,s_t$, where m_t, s_t are the per-time mean and spread of the path over calm calibration episodes and the single scalar c fixes the false-alarm probability; any drift is absorbed into m_t, whatever its shape, which the full-window rule assumes to be mild and the drift-centred variant of Table 7 assumes to be linear. Under both rules an alarm anywhere in a calm episode is a false alarm. In the simulation they report 0.04-0.06 and 0.01 on calm episodes at every drift level, and 0.99 and 0.92 on episodes that contain the change.

The calm test. For every detector, every setting and every seed we therefore also score a fourth episode set (seed $+3500$) that is calm throughout, under the identical rule, and report the resulting detection rate as *calm-test detection*. It is the rate the rule reports when there is nothing to report, and it is the diagnostic that should accompany any detection rate in this setting.

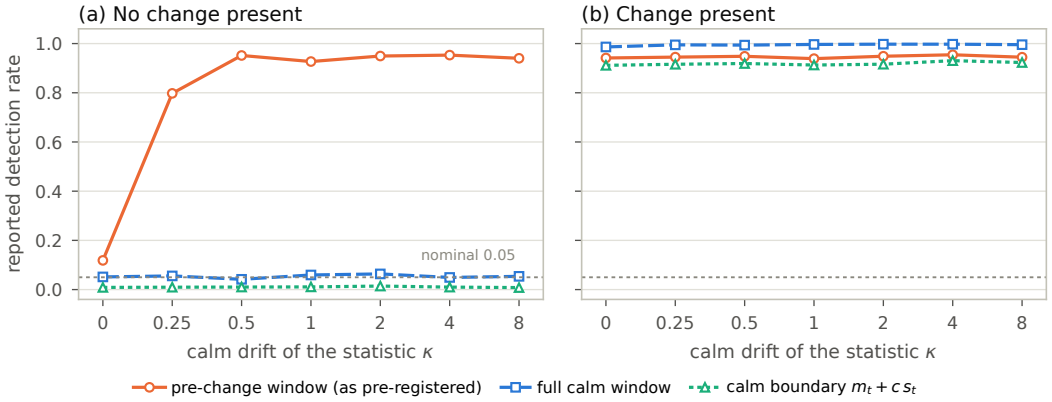


Figure 6: The alarm rule measured on synthetic paths, with no model in the loop (`metric_artifact.py`). **(a)** Detection rate reported on episodes containing *no* change, against the calm drift κ of the statistic. **(b)** Detection rate on episodes containing one, with identical evidence in every column. The pre-registered rule stops distinguishing the two once the statistic drifts; the drift-robust rules do not.

Where the drift comes from, and how much of it is structural. Calling the drift "a misspecified null" is true but not specific enough, and for RATIO a part of it is structural rather than an artefact of fitting. The null factorizes over horizons, $p_{\mathrm{VAR}}(y_t \mid z_t) = \prod_\delta \mathcal{N}(\bar{z}_{t+\delta}; \mu + \Phi^\delta(z_t - \mu), \Sigma_\delta)$, while

Table 6: The alarm rule on synthetic paths $M_t = \kappa t + \sigma W_t + \delta(t-\nu)^+$, no model involved (`metric_artifact.py`, 400 episodes per set, 5 replications, $\delta = 0.5$). *Calm-test detection* is the rate each rule reports on episodes with $\delta = 0$; *calm alarm* is its probability of alarming anywhere inside a calm episode, which the rule calibrates to 0.05 on its own calibration window.

	calm-test detection			detection, change present			calm alarm
κ	pre-change	full window	boundary	pre-change	full window	boundary	pre-change
0	0.119	0.051	0.009	0.941	0.986	0.912	0.181
0.25	0.798	0.056	0.010	0.945	0.995	0.916	0.854
0.5	0.952	0.042	0.010	0.948	0.994	0.919	0.999
1	0.927	0.059	0.011	0.939	0.996	0.913	1.000
2	0.949	0.064	0.014	0.948	0.998	0.916	1.000
4	0.953	0.049	0.010	0.954	0.997	0.931	1.000
8	0.940	0.054	0.008	0.945	0.996	0.923	1.000

Table 7: Shape switch, aligned protocol: the same paths scored by every alarm rule of Section H. *calm* is the calm-test detection rate — the rate the rule reports on episodes with no change (seed+3500); it should be near the nominal 0.05. *drift* is the mean per-step increment of the statistic on calm holdout episodes. Detection rates are means over the seeds at achieved PFA 0.05.

		pre-change window		full calm window		calm boundary	
Detector	drift	det.	calm	det.	calm	det.	calm
config A							
One-step residual, RESID (ours)	+2.05	0.946	0.948	0.056	0.050	0.009	0.005
Shape score (ours)	+22.61	0.966	0.960	0.056	0.051	0.009	0.014
Trajectory density ratio (ours)	+16.29	0.934	0.943	0.059	0.056	0.006	0.003
Scan-B (kernel MMD)	0.00	0.106	0.093	0.044	0.036	0.051	0.039
GLR-CUSUM	0.00	0.147	0.089	0.056	0.041	0.063	0.037
Frozen VAR e-process	-0.04	0.051	0.012	0.039	0.006	0.105	0.028
Frozen MLP e-process	-0.04	0.058	0.017	0.044	0.014	0.082	0.026
Exact-LR oracle (true latent)	-0.04	0.944	0.001	0.945	0.001	0.972	0.025
config B							
One-step residual, RESID (ours)	+0.86	0.947	0.953	0.052	0.044	0.003	0.000
Shape score (ours)	+11.91	0.952	0.958	0.057	0.050	0.014	0.014
Diffusion latent MMD (reflected)	+0.02	0.767	0.768	0.069	0.059	0.014	0.008
Scan-B (kernel MMD)	0.00	0.109	0.092	0.047	0.034	0.055	0.034
GLR-CUSUM	0.00	0.125	0.084	0.044	0.030	0.045	0.033
Frozen VAR e-process	-0.04	0.047	0.014	0.039	0.011	0.086	0.028
Frozen MLP e-process	-0.03	0.054	0.026	0.039	0.010	0.043	0.014
Exact-LR oracle (true latent)	-0.04	0.944	0.001	0.945	0.001	0.972	0.025

the flow models the joint law on $\mathbb{R}^{4d}$. Whenever the learned residual captures conditional cross-lag dependence - and in a persistent system $\mathrm{Cov}(\bar{z}_{t+\delta_1}, \bar{z}_{t+\delta_2} \mid z_t) \neq 0$ - the expected calm increment is bounded below by the conditional multi-information among the lags,

$$\mathbb{E}_{\mathrm{calm}}\big[\log p_t^{\mathrm{flow}}(y_t) - \log p_{\mathrm{VAR}}(y_t \mid z_t)\big] \;\geq\; I\big(\bar{z}_{t+1}; \bar{z}_{t+2}; \bar{z}_{t+4}; \bar{z}_{t+8} \,\big|\, z_t\big) \;>\; 0, \tag{7}$$

so a strictly positive calm drift is guaranteed no matter how well the VAR marginals are fitted. Proposition 1 escapes this only because it assumes diagonal Φ and $r_\phi \equiv 0$, which makes the joint equal to the product. The remedy is available and we did not take it: the VAR null has a closed-form joint covariance across the lags, $\Sigma_{\delta_1,\delta_2} = \Phi^{|\delta_2-\delta_1|}\Sigma_{\min(\delta_1,\delta_2)}$, so scoring p_{VAR} as one $4d$-dimensional Gaussian rather than a product of marginals would remove Equation (7) as a driver and leave only genuine misspecification. We expect this to reduce RATIO's calm drift substantially and to change its power very little, since the drift-robust rules already absorb the drift; testing that is the obvious next experiment.

Why the shape score drifts, and why the pilot does not. The same kind of accounting explains SHAPE. Its increment uses a *standard* Student-t_5 against a standard normal. A standard t_5 has variance $\nu/(\nu-2) = 5/3$, so the comparison is not made at matched energy: under the calm law the expected increment is negative, and under a variance-matched mixture the shape gain has to overcome that offset before the restart mixture can rise. The pilot of Section P avoids this by construction, because it rescales the Student-t to unit variance before taking the ratio, which is what makes it a test of shape at matched energy in the sense of Proposition 4. Defining SHAPE with the unit-variance t in the first place is the cleaner choice and we recommend it.

Table 8: Nonlinear swirl, aligned protocol: the same paths scored by every alarm rule of Section H. *calm* is the calm-test detection rate — the rate the rule reports on episodes with no change (seed+3500); it should be near the nominal 0.05. *drift* is the mean per-step increment of the statistic on calm holdout episodes. Detection rates are means over the seeds at achieved PFA 0.05.

		pre-change window		full calm window		calm boundary	
Detector	drift	det.	calm	det.	calm	det.	calm
config A							
One-step residual, RESID (ours)	+5.54	0.950	0.945	0.113	0.063	0.040	0.006
Shape score (ours)	+17.95	0.957	0.944	0.308	0.069	0.107	0.015
Trajectory density ratio (ours)	+3.89	0.943	0.952	0.792	0.068	0.626	0.012
Scan-B (kernel MMD)	0.00	0.548	0.091	0.408	0.031	0.410	0.036
GLR-CUSUM	0.00	0.636	0.100	0.442	0.034	0.453	0.032
Frozen VAR e-process	-0.04	0.536	0.005	0.523	0.005	0.715	0.027
Frozen MLP e-process	-0.04	0.265	0.014	0.254	0.013	0.351	0.030
Exact-LR oracle (true latent)	-0.05	0.948	0.001	0.948	0.001	0.987	0.042
config B							
One-step residual, RESID (ours)	+3.55	0.950	0.949	0.102	0.065	0.036	0.009
Shape score (ours)	+6.27	0.931	0.841	0.371	0.062	0.165	0.014
Diffusion latent MMD (reflected)	+0.01	0.891	0.544	0.671	0.048	0.450	0.009
Scan-B (kernel MMD)	0.00	0.770	0.104	0.661	0.031	0.660	0.031
GLR-CUSUM	0.00	0.635	0.101	0.437	0.037	0.452	0.031
Frozen VAR e-process	-0.04	0.558	0.004	0.560	0.005	0.757	0.032
Frozen MLP e-process	-0.04	0.261	0.014	0.237	0.011	0.303	0.020
Exact-LR oracle (true latent)	-0.05	0.948	0.001	0.948	0.001	0.987	0.042

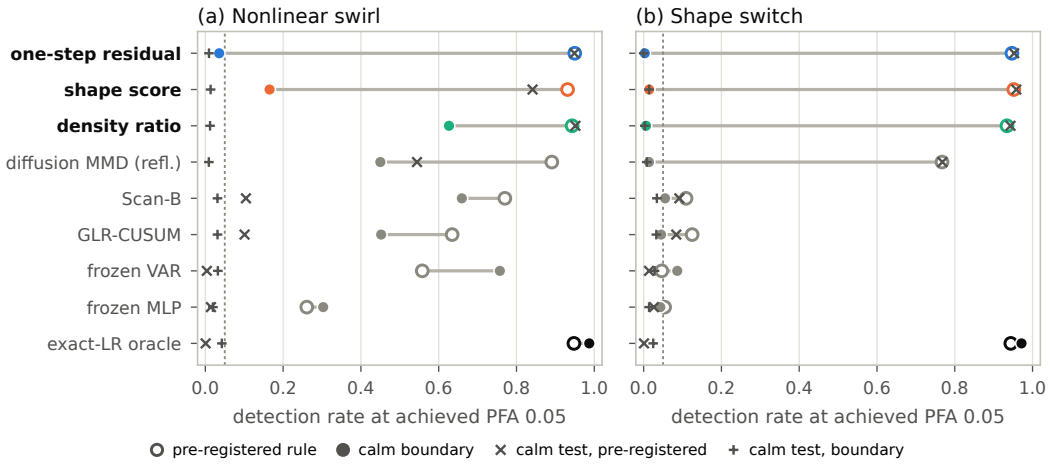


Figure 7: What the calm test removes. For each detector, the detection rate under the pre-registered rule (hollow) and under the calm boundary (filled), with the calm-test rate of each (crosses). A detector whose hollow marker sits far from its filled one was being credited for drift, not evidence; a detector whose pre-registered cross is high was reporting detections on episodes with no change.

What these rules assume (moved from Section 5). Our correction is a diagnosis plus two rules, not a theory of valid sequential testing. The full-window rule assumes the calm drift is mild over the monitored window; the calm boundary assumes only that calm episodes are exchangeable with the pre-change part of test episodes, but it estimates a per-time mean and spread from finitely many calm episodes and is conservative in small samples. The two rules disagree about the ordering of the strongest detectors on the swirl (Section 4.4). An anytime-valid statistic whose null is genuinely correct - an e-process on a null the data actually follow - needs neither rule, which is exactly why the frozen e-processes and the oracle are unaffected by the correction. Our calm test detects the failure; it does not repair a misspecified null.

What the calm test shows. Tables 7 and 8 give every detector under every rule, with its calm-test twin beside it and its measured calm drift. The exact oracle and the frozen e-processes have calm drift at or below zero and calm-test detection at the nominal level under all three rules: their reported detection rates mean what they say. Our three learned readouts have large positive calm drift, and under the pre-registered rule their calm-test detection is as high as their detection - the inflation

predicted by Equation (6). Under the drift-robust rules their calm-test detection returns to the nominal level and their detection rate falls accordingly. Section 4 reports the drift-robust numbers, and Section I keeps the pre-registered ones.

I FULL RESULTS, CURVES AND STATISTICS

Table 10 is the table this paper would have reported under the pre-registered alarm rule, on exactly the runs of Table 1; Table 11 adds the pre-registered bookkeeping as well. Neither is deleted and neither is retracted as a *measurement*: both report what that rule returns. What Section 4.2 establishes is that the rule does not measure what its name says whenever the *calm* column beside a detection rate is close to it.

Table 9: Per-seed detection rate under the calm boundary, aligned protocol, config A — the seed-level values behind Table 1. *calm* is the mean calm-test rate over the same seeds. Every other per-seed table in this appendix reports the pre-registered rule; this one reports the rule the main text uses.

Detector	s0	s1	s2	s3	s4	s5	s6	mean	calm
Nonlinear swirl									
One-step residual, RESID (ours)	0.045	0.050	0.047	0.057	0.020	0.025	0.033	0.040	0.006
Shape score (ours)	0.133	0.107	0.160	0.147	0.080	0.093	0.030	0.107	0.015
Trajectory density ratio (ours)	0.440	0.597	0.693	0.670	0.619	0.630	0.737	0.626	0.012
Scan-B (kernel MMD)	0.268	0.301	0.413	0.367	0.452	0.456	0.612	0.410	0.036
GLR-CUSUM	–	0.500	0.385	0.485	0.460	0.432	0.457	0.453	0.032
Frozen VAR e-process	0.640	0.630	0.612	0.773	0.767	0.813	0.773	0.715	0.027
Frozen MLP e-process	–	0.307	0.251	0.311	0.467	–	0.418	0.351	0.030
Exact-LR oracle (true latent)	0.993	0.997	0.987	0.960	0.993	1.000	0.980	0.987	0.042
Shape switch									
One-step residual, RESID (ours)	0.020	0.027	0.007	0.000	0.000	0.007	0.000	0.009	0.005
Shape score (ours)	0.012	0.013	0.013	0.013	0.000	0.000	0.013	0.009	0.014
Trajectory density ratio (ours)	0.000	0.000	0.003	0.007	0.007	0.000	0.027	0.006	0.003
Scan-B (kernel MMD)	0.020	0.053	0.093	0.033	0.019	0.060	0.080	0.051	0.039
GLR-CUSUM	0.087	0.060	0.067	0.030	–	0.027	0.111	0.063	0.037
Frozen VAR e-process	–	0.120	0.173	0.069	0.063	0.044	0.163	0.105	0.028
Frozen MLP e-process	0.077	0.043	0.120	0.037	0.156	0.027	0.113	0.082	0.026
Exact-LR oracle (true latent)	0.964	0.957	0.973	0.953	0.960	0.993	1.000	0.972	0.025

What the flow buys (moved from Section 4.4). The comparison that survives is between detectors run on the *same* representation: an exact-density readout (RATIO) and a linear e-process (frozen VAR) both beat the kernel and diffusion statistics on those codes, and RATIO is the only flow-native readout with real power. RESID and SHAPE, which looked strongest under the pre-registered rule, are the two that drift most and keep the least.

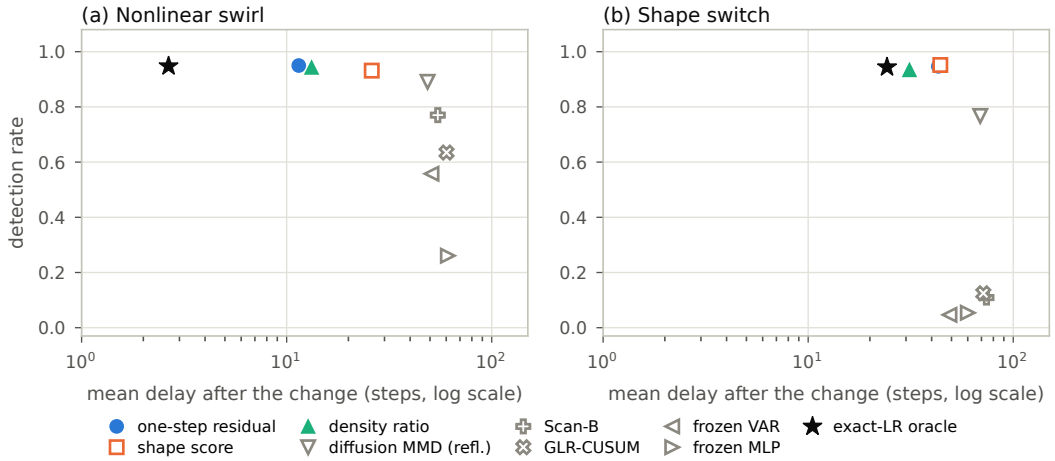


Figure 8: Mean delay vs. detection rate, aligned protocol (config B; density ratio from config A). The oracle is fastest; our readouts detect as often as the oracle but later; energy and kernel baselines detect less often and much later. Hollow markers are drawn on top where points overlap.

Table 10: The same runs under the *pre-registered* alarm rule, which credits calm drift as detection (Section H). Every detection rate whose *calm* twin is close to it was measuring drift, not the change.

	Nonlinear swirl				Shape switch			
	config A		config B		config A		config B	
Detector	det.	calm	det.	calm	det.	calm	det.	calm
One-step residual, RESID (ours)	0.950	0.945	**0.950**	0.949	0.946	0.948	0.947	0.953
Shape score (ours)	**0.957**	0.944	0.931	0.841	**0.966**	0.960	**0.952**	0.958
Trajectory density ratio (ours)	0.943	0.952	–	–	0.934	0.943	–	–
Diffusion latent MMD (reflected)	–	–	0.891	0.544	–	–	0.767	0.768
Scan-B (kernel MMD)	0.548	0.091	0.770	0.104	0.106	0.093	0.109	0.092
GLR-CUSUM	0.636	0.100	0.635	0.101	0.147	0.089	0.125	0.084
Frozen VAR e-process	0.536	0.005	0.558	0.004	0.051	0.012	0.047	0.014
Frozen MLP e-process	0.265	0.014	0.261	0.014	0.058	0.017	0.054	0.026
Exact-LR oracle (true latent)	0.948	0.001	0.948	0.001	0.944	0.001	0.944	0.001

Table 11: Same as Table 10 under the original (pre-registered) protocol *and* the pre-registered alarm rule, on the same trained models. Baselines are additionally depressed by warm-up windows (Section G).

	Nonlinear swirl				Shape switch			
	config A		config B		config A		config B	
Detector	det.	calm	det.	calm	det.	calm	det.	calm
One-step residual, RESID (ours)	0.945	0.946	**0.954**	0.948	0.946	0.950	0.952	0.955
Shape score (ours)	0.953	0.941	0.923	0.836	**0.963**	0.965	**0.964**	0.961
Trajectory density ratio (ours)	**0.954**	0.945	–	–	0.935	0.939	–	–
Diffusion latent MMD (reflected)	–	–	0.798	0.291	–	–	0.462	0.452
Scan-B (kernel MMD)	0.043	0.000	0.444	0.008	0.001	0.001	0.002	0.001
GLR-CUSUM	0.460	0.046	0.474	0.047	0.063	0.037	0.029	0.021
Frozen VAR e-process	0.120	0.000	0.156	0.000	0.001	0.000	0.000	0.000
Frozen MLP e-process	0.094	0.000	0.046	0.000	0.008	0.001	0.002	0.001
Exact-LR oracle (true latent)	0.965	0.000	0.965	0.000	0.948	0.000	0.948	0.000

Replicates. Besides the runs of Tables 1 and 11, Table 14 lists the stored runs of the pre-registered protocol, produced earlier by separate trainings on MPS and CUDA. For our readouts they agree with the reruns (config A on MPS is reproduced exactly). The stored diffusion baseline, run with its own CUDA training of the encoder, is much lower than the same-latent rerun (swirl 0.798 vs. 0.611), which is why the main text uses same-latent numbers only.

J THE EXACT ORACLE

Correction. An earlier version of the shape-switch oracle passed one-episode arrays of shape $(1, T, d)$ to a function written for (T, d); its slices then selected time steps instead of latent dimensions, left uninitialized rows in the predicted mean, overflowed to NaN, and produced a degenerate achieved-PFA range. We had interpreted this as "the exact likelihood ratio is heavy-tailed and cannot be calibrated at PFA 0.05". That interpretation was wrong. With the fix the oracle is well behaved on every seed (Table 21): detection 0.880-0.980 (mean 0.948), mean calm increment -0.277 and mean post-change increment +0.297 nats per step, matching $-\mathrm{KL}(P\|Q) = -0.278$ and $\mathrm{KL}(Q\|P) = 0.299$ obtained by numerical integration.

The oracle is a reference, not a bound. Its likelihood ratio is optimal for tests with a known change time and its restart mixture is a Shiryaev-Roberts-type statistic, but neither is optimal for "detection rate at a fixed empirical PFA over a finite window". Its value depends on bookkeeping: starting the restart mixture at the first full-context index instead of $t = 0$ moves it from 0.965 to 0.891 on the swirl, because early path values are not diluted by a burn-in (Table 21). Where our readouts

Table 12: Mean detection delay (steps after the change, detected episodes only) at achieved PFA 0.05, aligned protocol, 7 seeds, *under the pre-registered alarm rule*. That rule is the one Section H retires, so these delays are not comparable with the calm-boundary delays quoted in Section 4 (for the shape-switch oracle, 24.2 here against 18.1 under the calm boundary); for a detector whose calm-boundary detection rate is at chance the delay is not informative at all. Delays of the native detectors are measured from the time of the latest observation they use.

	Nonlinear swirl		Shape switch	
Detector	config A	config B	config A	config B
One-step residual, RESID (ours)	9.3	11.5	27.7	43.1
Shape score (ours)	16.4	26.0	32.9	44.1
Trajectory density ratio (ours)	13.2	–	31.2	–
Diffusion latent MMD (reflected)	–	48.7	–	69.0
Scan-B (kernel MMD)	63.4	54.6	72.5	74.0
GLR-CUSUM	60.8	60.2	76.2	71.5
Frozen VAR e-process	51.0	50.9	49.7	48.9
Frozen MLP e-process	63.4	61.1	56.8	60.0
Exact-LR oracle (true latent)	2.7	2.7	24.2	24.2

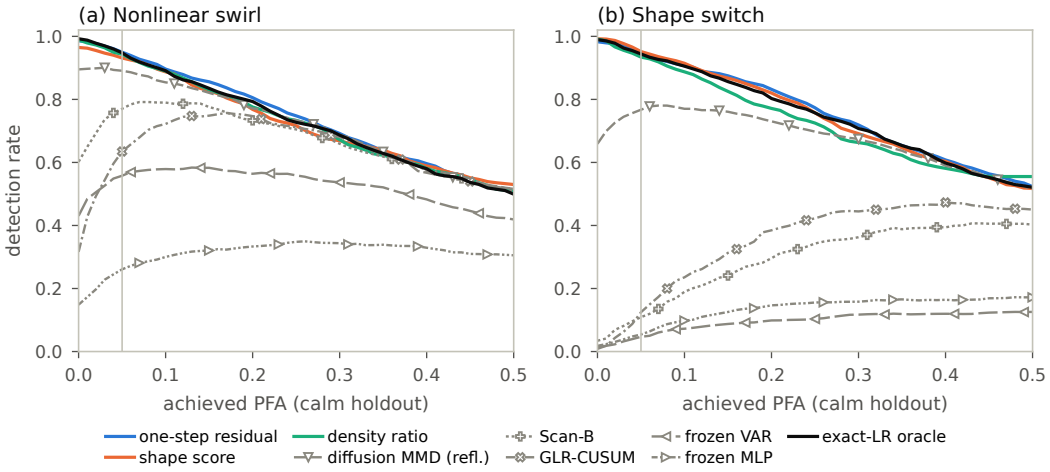


Figure 9: Operating curves: detection rate against achieved PFA, averaged over seeds (aligned protocol). The vertical line marks PFA 0.05.

exceed it per seed they do so within the saturation band of Section G; the oracle remains several times faster (Table 12).

K THE DIVERGENCE CORRECTION

The earlier implementation computed $\operatorname{div} v$ as the gradient of $\sum_i v_i$ summed over coordinates, that is $\mathbf{1}^\top J \mathbf{1}$ rather than $\operatorname{tr} J$. The two coincide when J is diagonal, which is why the Gaussian validity tests (base velocity only, residual at its zero initialization) passed. With a trained residual they differ: on a lightly trained arm the per-step increments of the two versions correlate at 0.95 with a mean absolute difference of 1.4 nats. The exact trace costs one vector-Jacobian product per coordinate (32 per step), about three times the original cost on MPS thanks to batching. Table 22 gives per-seed values: the original-divergence rows reproduce the stored runs, and the exact trace raises the density ratio on the swirl from 0.899 to 0.954 under the original protocol.

Absorption in practice. Proposition 5 predicts that a density re-evaluated on post-change context gives up part of the evidence. The prediction is about delay, and delay is unaffected by the calibration artefact: RATIO alarms 85.9 steps after the change while the oracle, which never re-estimates, alarms after 5.4. An earlier version of our pipeline computed the ODE divergence as the sum of all Jacobian

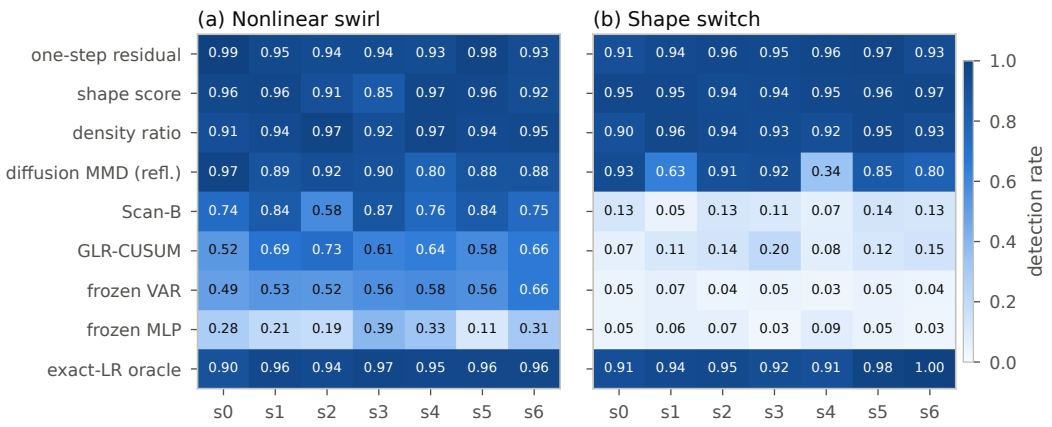


Figure 10: Per-seed detection rates (aligned protocol; config B, density ratio from config A).

entries instead of the trace; that version gave RATIO only 0.899 under the pre-registered rule and overstated the gap (Section K).

L SHAPE SPECIFICITY

Table 23 compares the shape score with its calm-standardized variant. Standardization removes calm-level location and scale of the inverse base point, which by Proposition 4 removes the energy term of the score. The collapse on the shape switch shows that in the learned latent the change is carried mostly by a scale increase of u_t: the encoder and moment head, trained on calm data, map heavier-tailed innovations to inverse base points with inflated variance rather than to standardized points with excess kurtosis. The score is therefore best described as a variance-and-shape discrepancy between the future and the model's base. It still fires on the linear control after standardization because a rotation change distorts the joint law of the four lags. A pure shape test would need the moment head to track post-change scale (which would, by Proposition 5, cost scale evidence) or a score orthogonal to energy; we leave this open.

Figure 11 measures how much of the change reaches the inverse base point at all, and the answer is uncomfortable. On the shape switch the per-coordinate standard deviation of u moves from 1.447 to 1.456 ($+0.6\%$) and its kurtosis from 3.161 to 3.154 (-0.2%): to two even moments, the post-change inverse base point is the calm one. The mixture that doubles the innovation kurtosis in the latent state (Section C) is therefore almost entirely absent from *this* statistic, before any detector is applied. That is not the same as being absent from the representation: Section P shows the one-step code innovation gains +0.361 in excess kurtosis, more than the observation innovation does, and the loss occurs when the multi-lag flow residual dilutes it. On the swirl, by contrast, the same coordinates do move - standard deviation $+9.2\%$, kurtosis $+3.9\%$ - which is why the flow readouts retain power there and not here. The obstruction is thus not the encoder - Table 28 shows the code innovation gains +0.361 in excess kurtosis - but the pair of operations between the code and this statistic: the energy term of the score (Proposition 4), and the dilution of a fourth-moment signature by the prediction error carried in the multi-lag residual. Section H takes this up again, since it explains why the learned readouts carry so little shape evidence once the evaluation rule is made drift-robust.

M COST, DISCRETIZATION AND THE CHOICE OF INTEGRATOR

Two properties of RATIO are implementation choices rather than results, and both are measurable without training anything: what the readout costs, and how much the probability-flow ODE's discretization costs it in accuracy. Table 24 reports both.

Cost. At the configuration of this paper (latent dimension 8, four lags, trajectory dimension 32, 16 Euler steps) the trajectory density costs 7.4 ms per monitored window with the exact Jacobian trace, or 1.75 s per 300-step episode, against 0.6 ms with the $\mathbf{1}^\top J\mathbf{1}$ surrogate. The gap is structural: the

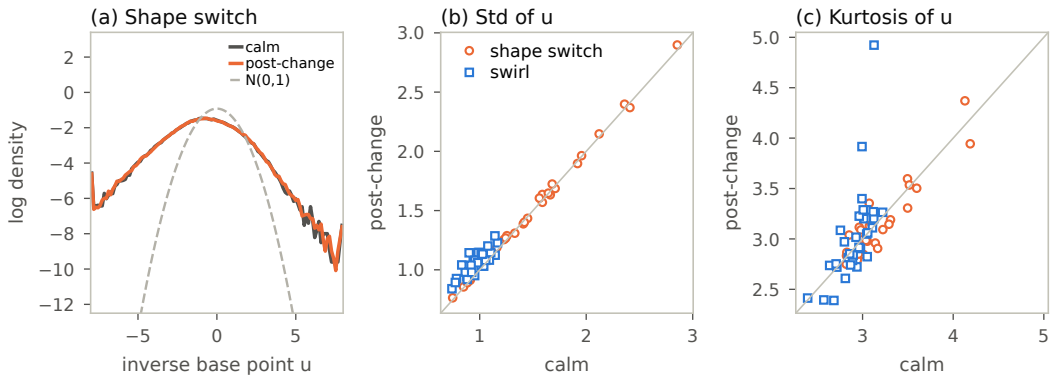


Figure 11: Where the change appears in the inverse base point $u = (y - m) \oslash s$ (seed 0, config A). **(a)** Pooled log-density of u on calm windows and on post-change windows of the shape switch, with the standard normal for reference. **(b,c)** Per-coordinate standard deviation and kurtosis, calm vs. post-change, for both settings; the diagonal is "no change".

exact trace needs one vector-Jacobian product per trajectory coordinate, so it is $O(d_{\text{traj}})$, while the surrogate is $O(1)$ - measured as $13\times$, $25\times$ and $49\times$ at trajectory dimensions 32, 64 and 128. Cost is linear in the step count and in the trajectory dimension, and the trajectory dimension is the latent dimension times the number of lags, so doubling the horizon and doubling the latent size cost the same. RESID and SHAPE need no integration at all: they are a residual and a closed-form score at the inverse base point, and their cost is that of one encoder pass. Whether the exact trace is worth its price is settled elsewhere in this paper - with the surrogate the statistic is not a density ratio (Section K) - but at d_{traj} in the hundreds a Hutchinson estimator becomes the obvious middle term, and we have not tested one.

Discretization, and a better default. In the diagonal Gaussian case the flow density has a closed form (✓ `ode_logdensity_base`), so the integration error is exactly measurable. Euler is first order: halving the step halves the error, reaching 0.06828 nats at the 16 steps this paper uses. A midpoint (Heun) integrator, at two velocity evaluations per step, is second order and reaches 0.00183 nats with *four* steps - about 37 times more accurate than our default at half its cost - and 0.00707 nats with two. Below 10^{-5} nats the sweep stops improving, which is where float32 accumulation dominates.

This is a straightforward improvement to the method as implemented, and we report it as one: a second-order integrator with four steps dominates first-order Euler with sixteen on both axes. Table 25 asks whether it matters for the *detection*, which is a different question, since an error of a few hundredths of a nat per window enters a restart mixture that accumulates over hundreds of steps.

Does the integrator change the detection? Table 25 scores the same trained model, on the same episodes, with the density integrated three ways. It does not. Across integrators the detection rate moves by less than the spread across seeds, and the more accurate integrator sits at a *stricter* operating point rather than a worse one: midpoint with four steps reports a lower detection rate and a lower calm-test rate than Euler with sixteen. A discretization error of a few hundredths of a nat per window is simply not visible after a restart mixture has accumulated over a hundred and fifty post-change steps. We therefore recommend the second-order integrator on grounds of accuracy and cost, and report that it buys no detection.

N How far ahead should the flow predict?

The trajectory is the target code at lags $(1, 2, 4, 8)$, and that choice was made once and never tested. It is not innocuous: it fixes the dimension the flow has to model, hence the cost of the exact trace (Section M), and it decides how much of the future law the readouts can see. Table 27 retrains at five horizons and scores the readouts on identical episodes.

The horizon matters more than anything else we ablated, and our choice was not the best one. RATIO reaches 0.904 at lags $(1, 2, 4)$ - per-seed 0.847-0.947, the tightest spread in the table - against 0.391 at

the $(1, 2, 4, 8)$ we used, and it does so at lower cost (1.49 s versus 1.89 s per episode). Extending the horizon once more, to $(1, 2, 4, 8, 16)$, destroys the readout completely: 0.007, indistinguishable from its own calm test. The pattern is not "longer is better" but the opposite past three lags, and the reason is visible in the construction: each extra lag adds d_z dimensions to a density that must be estimated from the same data, and the composite VAR null it is tested against degrades at long lags, so both sides of the ratio get noisier while the change stays where it was. RESID and SHAPE are flat across the sweep, as they must be - neither reads the joint trajectory law - and the oracle is constant at 0.904 by construction, since it never sees the model.

Two caveats. The sweep is three seeds and the per-seed spread at $(1, 2, 4, 8)$ is wide, so we claim an ordering between $(1, 2, 4)$ and $(1, 2, 4, 8)$ - their per-seed ranges do not overlap - and not a precise gap. And this sweep ran on a different machine from the rest of the paper (cpu), so absolute values should be compared within Table 27 rather than against Table 1. We did not re-run the main results at the better horizon: doing so after seeing this sweep would be exactly the selection the calm test of Section H exists to catch, and the honest statement is that our headline configuration is a defensible but suboptimal point that we chose before measuring.

The full campaign at the preferred horizon. An ablation on one setting is weak evidence, so we re-ran the entire campaign at lags $(1, 2, 4)$: seven seeds, all three settings, the same protocol and the same calm test (Table 26). Three things follow. RATIO reaches 0.818 on the swirl against 0.682 for the frozen VAR e-process on the same codes, so at this horizon the flow-native readout is the strongest learned detector rather than the second strongest. The energy-free change remains undetected by everything (0.005 for RATIO, 0.093 for the best baseline), which rules out the obvious objection that our negative result was an artefact of a badly chosen horizon. And RATIO remains silent on the linear-rotation control (0.054) while the detectors whose nulls cannot express a rotation fire, so Proposition 1 continues to hold where it should.

The horizon was selected on the swirl, so the swirl column of Table 26 is partly in-sample; the shape-switch and linear-rotation columns are not, and they are the two that carry the paper's conclusions. We keep Table 1 at the pre-registered horizon as the primary result.

O SCOPE OF THE CLAIMS AND UNTESTED DETECTOR FAMILIES

Proposition 1 is an algebraic containment result: it gives $\ell_t \equiv 0$ when Φ and S are diagonal, the moment head emits the exact lag moments and $r_\phi \equiv 0$. Our benchmark satisfies none of these exactly - the latent dynamics are planar rotations with large off-diagonal entries, the lag covariances are not diagonal, and r_ϕ is active - so the near-silence we observe on the linear control is an empirical consequence of the null already explaining a rotation, not an instance of the theorem. Likewise Proposition 3 constrains statistics that are functions of residual *energy*; it says nothing about learned likelihood-ratio statistics or segmentation models, which is why BOCPD is not covered by it (Section F). Our negative result is therefore a statement about the monitors we evaluated, not a theorem about all learned monitors. Two families we did not evaluate are the natural next tests, and neither is excluded by our theory: neural likelihood-ratio CUSUM statistics (Gong et al., 2022), which learn the alternative rather than assuming an energy class, and contrastive predictive representations (Deldari et al., 2021), which are trained with an objective other than one-step prediction. The diagnosis of Section P predicts that the first family can succeed if its statistic reads a residual in which the change survives, and that the second changes the representation - which our measurements suggest was never the binding constraint.

P WHERE THE FOURTH MOMENT GOES

If no learned detector sees the energy-free change, the natural conclusion is that the encoder throws it away. We measured that assumption instead of making it, by tracking the excess kurtosis of the innovations at every stage between the observation and the statistic the readouts consume (Table 28, `moment_pipeline.py`; one trained model, pooled over episodes, time and coordinates). The assumption is wrong.

The injected change raises the excess kurtosis of the *observation* innovation by +0.233. At the *code* innovation - the one-step residual $z_t - \mu - \Phi(z_{t-1} - \mu)$ of a VAR(1) fitted on calm codes - it rises

by +0.361, which is *larger* than at the observation level. The encoder does not discard the change; it sharpens it. What loses it is the next step: the flow residual $y_t - m_t$ over the multi-lag trajectory gains only +0.052, and the inverse base point that SHAPE reads gains +0.071.

The arithmetic explains the gap. If a residual is the sum of a heavy-tailed innovation and an independent, roughly Gaussian prediction error, its excess kurtosis is that of the innovation scaled by the fourth power of the innovation's share of the amplitude. A residual that is half prediction error therefore keeps about a sixteenth of the fourth-moment signature. The multi-lag flow residual is dominated by ordinary prediction error, so it dilutes a fourth-moment change even when the change is fully present in the code. Two candidate explanations are excluded by the same table: temporal averaging, which would predict a loss by a factor of the context length between the observation and the code and is contradicted by the code innovation gaining more than the observation innovation; and absorption (Proposition 5), since the predicted scale is unchanged after the change (post/pre $= 1.0001$).

A single-seed pilot. The diagnosis suggests a readout rather than a new architecture: apply the shape core of Proposition 4 - the log ratio of a unit-variance Student-t to a Gaussian, which tests shape at matched energy - to the whitened one-step code innovation, and aggregate it with the same restart mixture as every other detector. On one seed of the shape switch, with no retraining and reading the codes of a model the campaign had already trained, this reaches detection 0.297 at a calm-test rate of 0.037, with calm drift -0.036 nats per step; every detector in Table 1 lies between 0.005 and 0.105 on the same task.

This is one seed, and we report it as a pilot rather than as a result. It is not comparable to the multi-seed entries of Table 1, it has not been run on the swirl or on the linear-rotation control, so its specificity to shape is untested, and it has not been tried on the real recording of Section R. What it does establish is that the information survives in the representation and that a readout aimed at the right residual can reach it; turning that into a detector with a measured operating characteristic is the obvious next step, and the code to do it ships with this paper (`innovation_shape.py`).

Q ANALYTIC-BASE ABLATION

The naive head outputs the velocity from an MLP on (x, τ, c) without the analytic base. Replacing base-plus-residual by that head leaves the one-step residual detector intact but destroys the linear predictability of the code, and the frozen VAR e-process with it. Three runs agree on this, at three sample sizes. On the three seeds run in both arms (aligned), the code's one-step VAR R^2 falls from 0.691 to 0.064, the frozen VAR e-process from 0.500 to 0.002, while RESID stays at 0.966 against 0.946 (Figure 13b, Table 29). Comparing the full 7-seed base runs with the 3-seed naive runs gives the same picture: R^2 0.703 against 0.064, frozen VAR 0.536 against 0.002, RESID 0.946. The stored pre-registered runs agree again: VAR R^2 0.046-0.068 without the base against 0.686-0.718 with it. The base is therefore not a detail of the density: it is what keeps a flow-trained code usable by detectors that never saw the flow.

R A REAL BACKGROUND WITH AN INJECTED ENERGY-FREE CHANGE

The synthetic benchmark settles what happens when we control the law end to end. It cannot settle what happens on a real signal, whose innovations are neither Gaussian nor stationary, and our one fully public attempt failed (Section S). This section takes the middle road that the failure leaves open: a real recording as the background, and the energy-free change of Proposition 3 injected into it, so that the change time and the absence of an energy signature are known exactly while everything else is real.

Construction. The background is the UCI individual household electric power consumption record (Hébrail & Bérard, 2006): $2{,}075{,}259$ one-minute samples over four years, seven numeric channels, of which we keep the 98.7% of rows that are complete. Episodes are disjoint windows of 300 samples, assigned to the training, validation, calibration, holdout, test and calm-test splits in rotation so that every split spans the whole four years and none can be favoured by season or trend. A VAR(1) fitted on the training split alone gives innovations e_t; a change episode is rebuilt forward from ν

as $x_t = \mu + \Phi(x_{t-1} - \mu) + s_t e_t$ with s_t an independent two-point scale of unit second moment. Before ν the episode is the recording, untouched; with $s_t \equiv 1$ the reconstruction returns the recording exactly, which is what the calm-test episodes are, so the two arms differ in the injected scale and in nothing else.

Making it exactly energy free. Drawing s_t independently makes the injection energy-free only in expectation, which is not good enough here: these innovations have a sample kurtosis of 55, so a few spikes decide the realized variance, and an i.i.d. draw moved it by more than ten percent - an energy signature that an energy detector could legitimately use. We therefore pair the innovations of a window by magnitude, deal the two scales one to each member of a pair, and rescale the whole sequence by the single constant that makes the realized post-change innovation energy equal the energy that same window would have had with no injection. The result is exact: 1.000000 (Table 30). The shape change is untouched by that constant; the innovation kurtosis rises to 111.

Configuration. The model is the same small tier as the synthetic benchmark (Section D) with the input projection resized to the seven channels: context 64, code dimension 8, lags $(1, 2, 4, 8)$, λ_{var}=1 (config A), 2000 AdamW steps, and the identical split sizes (150 training, 30 validation, 200 calibration, 200 holdout, 150 change test, 150 calm test). Three seeds are run; a seed controls the network initialization, the training batches, the assignment of episodes to splits and the permutation that pairs innovations with scales, so each seed is a fresh draw of both the model and the injection. The reproduction commands are in Section Y.

What "constant energy" means here. The invariance we enforce and verify is exact for the *innovations* of the fitted VAR(1) surrogate: the realized $\sum_t \|s_t e_t\|^2$ over the post-change window equals the $\sum_t \|e_t\|^2$ that window would have had without the injection, to 1.000000. It is not a statement about the realized variance of the observed series, which the persistence Φ propagates and which therefore moves a little; nor about the pre/post ratio within an episode, which differs from one in the recording itself and does so equally in the calm arm.

How strong must the change be? Table 30 answers a question the synthetic benchmark cannot pose. On this background the mixture used throughout the paper ($c = 1/2$) is essentially invisible *even to a detector that knows exactly what was injected*: the informed reference detects 0.046. The reason is that the recording's own innovations already have kurtosis 55, so multiplying the fourth moment by $25/16$ changes little that was not already there. Widening the mixture at constant energy raises the informed reference to 0.373 at $c = 0.25$ and 0.584 at $c = 0.2$, which is the operating point we adopt: strong enough that an informed statistic finds it reliably, so that a learned detector which misses it is missing something that is demonstrably there. *That a real heavy-tailed background hides an energy-free change far better than a Gaussian one is itself worth stating*, and it sharpens Proposition 3: blindness to such changes is a property of the statistic, but the amount of evidence available at all is a property of the background.

Result. Table 31 reports every detector under both alarm rules with its calm-test twin. The reference statistic is not an oracle - it knows the injection but inherits a Student-t fit per whitened coordinate and the independence of those coordinates - so it is a lower bound on what a fully informed detector could do, and an upper reference for the learned ones.

S PUBLIC TCPD PILOT

We adopted six series of the Turing change-point dataset (van den Burg & Williams, 2020). Each method is fitted on the longest calm segment between annotated change points (majority vote of annotators, merged within five steps) and thresholded at the 95th percentile of its calm statistic. The small-tier encoder with a one-dimensional input collapses under the non-collapse constraint (relaxed to median std 0.02, rank fraction 0.25 for this pilot); a five-channel embedding (level, difference, absolute difference, rolling mean, rolling standard deviation) with a stronger variance weight lets two series train, but the density ratio never alarms on them. A two-sided CUSUM on the z-scored series has recall one with dozens to hundreds of alarms and $F_1 \leq 0.11$ (Table 32). Under this protocol these series are weak-signal for every method we tried; a public-data evaluation needs a larger encoder, per-series protocols and longer training, which we did not attempt.

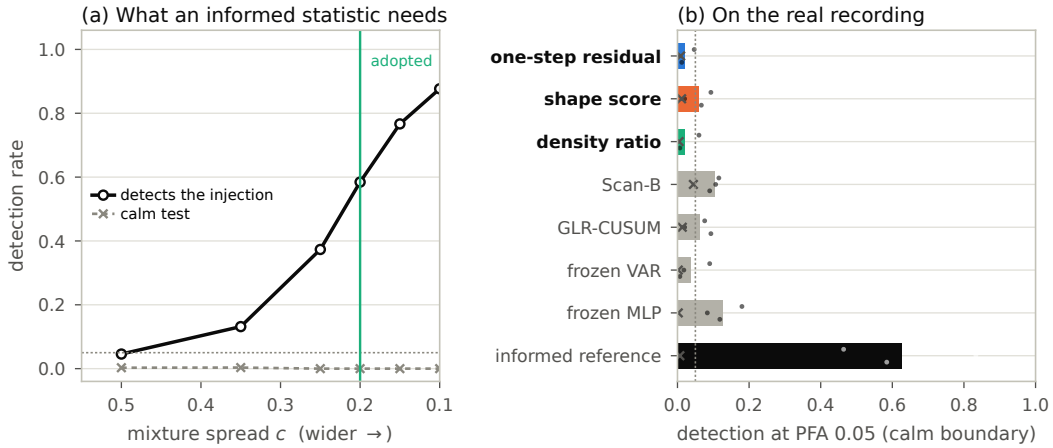


Figure 12: An energy-free change injected into a real recording. (a) Detection by the statistic that knows the injection, against the width c of the innovation scale mixture, at realized energy held exactly constant; the paper's synthetic mixture is the leftmost point and is invisible here. **(b)** Every detector at the adopted spread, with its calm-test rate as a cross. Our readouts in bold.

T Pre-registration and deviations

The design and six predictions were fixed before the F-JEPA-RD arm was trained. Table 33 states each and its outcome, under the *pre-registered* alarm rule in which they were written; Tables 7 and 8 give the same runs under the drift-robust rules. **The largest deviation is the alarm rule itself.** We pre-registered the rule of Section G and discovered, after all runs were complete, that it credits calm drift as detection (Section H); we therefore added a calm test set and two drift-robust rules, and report the main text under one of them. This is a post-hoc change of the primary metric, made for a stated and demonstrable reason, and it moves our own headline numbers down, not up: P2 survives it (RATIO 0.626 against the 0.419/0.389 bracket) while the shape-switch result that motivated the paper does not. Other deviations: the aligned protocol (post hoc, conservative), the divergence and oracle corrections, the second training configuration, the shape switch and the shape score (added after the pre-registered swirl experiments), and the diffusion baseline (added when the concurrent work appeared). The JEPA and state-fed references in Table 33 were measured on the same benchmark under the original protocol before this project (3 seeds).

U Extended related work

Flows and densities. Flow matching, rectified flows and stochastic interpolants learn velocity fields that transport a source to data (Lipman et al., 2023; Liu et al., 2023; Albergo & Vanden-Eijnden, 2023; Tong et al., 2024); the probability-flow ODE and the instantaneous change of variables turn such fields into exact densities up to discretization (Chen et al., 2018; Grathwohl et al., 2019; Song et al., 2021), with stochastic trace estimators as a cheaper alternative (Hutchinson, 1989). Normalizing flows give exact densities by construction (Rezende & Mohamed, 2015; Papamakarios et al., 2021); raw model likelihoods are known to be unreliable out-of-distribution scores (Nalisnick et al., 2019), which is one reason we use *ratios* against a calm null and a source-shape score rather than raw likelihood. Diffusion and flow models are established for probabilistic time-series forecasting and imputation (Ho et al., 2020; Rasul et al., 2021; Tashiro et al., 2021; Kollovieh et al., 2025) and are benchmarked on stochastic dynamics (Pfister et al., 2026).

V Controls and ablations: figure

Figure 13 accompanies Section 4.5 and Sections K and Q.

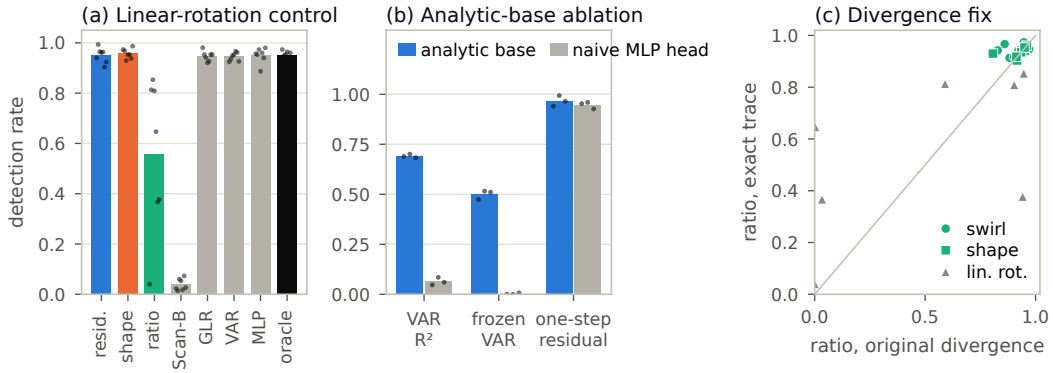


Figure 13: Controls. (a) Linear-rotation control (aligned, config A): no readout gains over the frozen VAR e-process. **(b)** Analytic-base ablation on the swirl (seeds 0-2): the naive head keeps the residual detector but destroys the code's VAR R^2 and the frozen detector. **(c)** Density ratio with the original divergence ($\mathbf{1}^\top J\mathbf{1}$) vs. the exact trace, per seed.

W What did not work: full discussion

Summary (moved from Section 4.5). Four negative findings are detailed in Section W. The first is the one above: three of our headline numbers, and one baseline's, were artefacts of the alarm rule, and we found it only because a statistic that drifts upward under the null contradicts Proposition 2. The learned shape channel mixes shape with scale: after calm standardization the shape score collapses to 0.073-0.167 on the shape switch (Section L). On public TCPD series our small encoder collapses or never alarms, while a calibrated CUSUM reaches $F_1 \leq 0.11$ (Section S).

The learned shape channel mixes shape and scale. Standardizing the inverse base point by its calm per-coordinate mean and variance before scoring removes what Proposition 4 calls the energy term. The standardized score collapses on the shape switch (0.073-0.167) and weakens on the swirl (0.298-0.559), but still fires on the linear control (0.932). In the learned latent the shape switch appears mostly as inflated variance of the inverse base point rather than as pure excess kurtosis, so SHAPE is a general variance-and-shape discrepancy score, not a kurtosis test (Section L).

Public change-point data. On six series of the TCPD benchmark (van den Burg & Williams, 2020), our small encoder collapses on short univariate inputs or trains but never alarms, even with a five-channel input embedding; a two-sided CUSUM calibrated the same way reaches $F_1 \leq 0.11$ with many false alarms. We make no claim on public benchmarks (Section S).

X Anticipated questions

Is the shape score shape-specific? No (Section L), and under a drift-robust alarm rule it has no power on the shape switch at all (Section 4.3). Proposition 4 is a statement about an exact density under the true law; the oracle realizes it, our learned score does not - not because the encoder discards the change (it does not; Section P) but because the multi-lag residual the score is computed on dilutes it.

Could a learned whitening at the inverse base point restore shape specificity? It cannot recover what is not there. Standardizing u_t by its calm per-coordinate mean and scale removes the energy term of the score exactly (Proposition 4) and is what Table 23 measures; but on the shape switch the post-change u_t differs from the calm one by $+0.6\%$ in standard deviation and -0.2% in kurtosis, so no invertible reweighting of u_t has material evidence to act on. A whitening - learned or analytic - helps only where the change survives into u_t, which on our benchmark means the swirl. The repair is not in the objective that trains the embedding - the embedding keeps the change - but in the residual the score is applied to (Section P).

Can the analytic base go beyond a diagonal Gaussian? Partly, and the cost is specific. A full or low-rank-plus-diagonal covariance keeps everything: the optimal-transport base velocity of

Equation (2) generalizes with $s \mapsto S^{1/2}$, the inverse base point becomes $u_t = S^{-1/2}(y_t - m_t)$, and $\int_0^1 \operatorname{div} v^\star = \log\det S^{1/2}$, so ✓ `integral_div_baseVel` carries over with a Cholesky (or Woodbury) factor in place of $\log s$; the cost is a factorization per window, $O(d^3)$ dense or $O(dr^2)$ low-rank, against $O(d)$ now. A *mixture* base is where it breaks: the map from source to data is no longer a single monotone affine map, so there is no closed-form inverse base point and no exact base density, and both the shape score of Section 3.2 and Proposition 1 lose their closed forms. Our reading is that a low-rank-plus-diagonal base is the useful extension and a mixture base is a different method.

Can absorption be limited in deployment? Yes, and the paper already contains the cheap version: RESID scores the residual with a *calm-fixed* metric and therefore cannot absorb, which is why Proposition 5 predicts and Section I observes its delay advantage. Two further options follow from the proof rather than from experiment: freeze the moment head after calibration so that m_t, s_t cannot follow the new regime (which costs pre-change fit whenever the calm law drifts), or condition on delayed context $X_{t-\Delta}$ so that post-change observations enter the conditioning set only after Δ steps (which converts absorption into a fixed extra delay of Δ). We have not measured either.

What about representation drift during calm periods? This is the failure we did measure, and it is the subject of Section H. Our readouts' statistics climb on calm holdout data by up to $+22.6$ nats per step, which is exactly what a misspecified calm null produces, and under the pre-registered alarm rule that climb was being read as detection. Online recalibration of the base moments or of the VAR null would reduce the drift at its source and is the right fix; the calm boundary of Section H is the cheaper one, since it absorbs whatever drift remains and costs nothing at inference. Neither repairs a null that is wrong in a way that correlates with the change.

Is RESID just a residual detector? Yes, by design. What is new is that a flow-trained model yields it together with a shape score and a density ratio, that the analytic base keeps the code usable by standard detectors (Section Q), and that Proposition 5 explains when the residual beats the density.

Why a synthetic benchmark? Because it provides an exact oracle and a change with provably no energy. Our one public-data attempt is reported as a failure.

Is the diffusion reimplementation faithful? It follows the pipeline of Kraevskiy & Prokhorov (2026) with the differences listed in Section F, is trained on the same latents as every other detector, and its variability across trainings is reported.

How can a learned detector exceed the oracle? At PFA 0.05 strong detectors saturate near $1-\mathrm{PFA}$; per-seed rates then reflect false-alarm sampling, not power. The oracle is several times faster.

Is the aligned protocol cherry-picked? It was introduced after pre-registration, but it raises every baseline and changes our readouts by at most 0.012; both protocols are reported on the same models.

Does Lean verify the code? No. It verifies the mathematics. The code is checked by unit tests of the same identities (base velocity, ODE vs. closed form, flow vs. VAR in the diagonal case), by exact reproduction of stored runs, and by the generated arithmetic checks.

Y REPRODUCIBILITY

Environment. Python 3.11, PyTorch 2.x, NumPy, SciPy, CUDA. Lean 4 `v4.33.1` with Mathlib `v4.33.1`.

Commands. From `supplementary/code`:

```
python verify_tj1.py; python test_fjepa_gaussian.py   # algebra + validity
python shape_gate.py
python shape_oracle.py -out ../results/oracle_shape7.json
S="0 1 2 3 4 5 6"; R=../results; D=../paths
for X in nonlinear_swirl shape_switch; do             # Tables 1, 7-10; Figs 1-4
  python bench_protocol.py -setting $X -lam-var 1 -flow-lr \
         -seeds $S -out $R/protocol_${X}_l1.json
  python bench_protocol.py -setting $X -b2608 \
         -seeds $S -out $R/protocol_${X}_l25.json
done
python bench_protocol.py -setting linear_rotation -lam-var 1 \
       -flow-lr -seeds $S -out $R/protocol_linrot_l1.json
python bench_protocol.py -setting nonlinear_swirl -lam-var 1 \
       -no-base -seeds 0 1 2 -out $R/protocol_nobase_l1.json
python paths_demo.py -setting shape_switch -seed 0 \
       -out $R/paths_shape.json   # Figs 12-13
python paths_demo.py -setting nonlinear_swirl -seed 0 -out $R/paths_swirl.json
python reeval_paths.py -dump-dir $D -all -out-dir $R   # re-score any alarm rule
python metric_artifact.py -out $R/metric_artifact.json      # Fig 10, Table 6
python bocpd_baseline.py -setting shape_switch -seeds $S \
       -out $R/bocpd_shape.json                             # Table 4 (also swirl, linrot)
python cost_and_discretization.py -out $R/cost.json         # Table 23
python ode_ablation.py -setting nonlinear_swirl -seeds 0 1 \
       -out $R/ode_ablation.json                            # Table 24
python horizon_ablation.py -setting nonlinear_swirl -seeds 0 1 2 \
       -horizons 1 1,2 1,2,4 1,2,4,8 1,2,4,8,16 \
       -out $R/horizon.json                                 # Table 26
FJEPA_LAGS=1,2,4 python bench_protocol.py -setting shape_switch -lam-var 1 \
       -flow-lr -seeds $S -out $R/protocol_shape_h124.json  # Table 25
python moment_pipeline.py -setting shape_switch -seed 0 \
       -out $R/moments.json                                 # Table 27
python innovation_shape.py -setting shape_switch -seeds 0 \
       -out $R/inno_shape_pilot.json                        # pilot, Section O
python semisynthetic.py -prepare -raw <UCI household_power_consumption.txt>
python semisynthetic.py -calibrate -out $R/semisynthetic_calib.json   # Table 29
python semisynthetic.py -seeds 0 1 2 -flow-lr -out $R/semisynthetic.json  # Table 30
TCPD_ROOT=<clone of alan-turing-institute/TCPD> python public_cpd.py \
       -out $R/fjepa_public_v2.json                         # Table 31
python make_numbers.py          # numbers.tex, tables, Numbers.lean
cd ../../paper/figures
python fig_theory.py; python fig_results.py
python fig_teaser.py; python fig_paths.py
cd ../../lean && lake exe cache get && lake build
```

The stored pre-registered runs are regenerated by `fjepa_detect.py`, `bench_recent.py` and `baseline_2608.py` (commands in the supplementary `README`).

Table 13: Paired comparisons over seeds, *computed under the pre-registered alarm rule* (aligned protocol): mean difference in detection rate, seeds won, paired t-test and Wilcoxon signed-rank p-values. **These significances do not support the paper's claims and are not used for them.** They quantify differences between rates that Section H shows to be inflated by calm drift; under the calm boundary the same contrasts on the shape switch are near zero or negative (Table 1). The table is retained because the pre-registered analysis plan specified it. With n=7 the smallest attainable two-sided Wilcoxon p is 0.016.

Setting	Ours	Baseline	Δ	wins	p_t	p_W
Swirl (A)	One-step residual	Scan-B	+0.402	7/7	3e-4	0.016
Swirl (A)	One-step residual	GLR-CUSUM	+0.314	7/7	3e-4	0.016
Swirl (A)	One-step residual	Frozen VAR e-process	+0.414	7/7	9e-7	0.016
Swirl (A)	One-step residual	Frozen MLP e-process	+0.685	7/7	2e-7	0.016
Swirl (A)	One-step residual	Exact-LR oracle	+0.002	4/7	0.877	0.812
Swirl (A)	Shape score	Scan-B	+0.409	7/7	1e-4	0.016
Swirl (A)	Shape score	GLR-CUSUM	+0.321	7/7	3e-4	0.016
Swirl (A)	Shape score	Frozen VAR e-process	+0.422	7/7	3e-7	0.016
Swirl (A)	Shape score	Frozen MLP e-process	+0.692	7/7	3e-8	0.016
Swirl (A)	Shape score	Exact-LR oracle	+0.010	3/7	0.472	0.688
Swirl (A)	Trajectory density ratio	Scan-B	+0.395	7/7	9e-5	0.016
Swirl (A)	Trajectory density ratio	GLR-CUSUM	+0.307	7/7	2e-4	0.016
Swirl (A)	Trajectory density ratio	Frozen VAR e-process	+0.407	7/7	2e-7	0.016
Swirl (A)	Trajectory density ratio	Frozen MLP e-process	+0.678	7/7	2e-7	0.016
Swirl (A)	Trajectory density ratio	Exact-LR oracle	-0.005	3/7	0.671	0.719
Swirl (B)	One-step residual	Diffusion latent MMD	+0.059	7/7	0.010	0.016
Swirl (B)	One-step residual	Scan-B	+0.180	7/7	0.003	0.016
Swirl (B)	One-step residual	GLR-CUSUM	+0.315	7/7	9e-5	0.016
Swirl (B)	One-step residual	Frozen VAR e-process	+0.392	7/7	6e-6	0.016
Swirl (B)	One-step residual	Frozen MLP e-process	+0.689	7/7	3e-6	0.016
Swirl (B)	One-step residual	Exact-LR oracle	+0.002	2/7	0.903	0.578
Swirl (B)	Shape score	Diffusion latent MMD	+0.041	4/7	0.205	0.297
Swirl (B)	Shape score	Scan-B	+0.161	6/7	0.008	0.031
Swirl (B)	Shape score	GLR-CUSUM	+0.297	7/7	1e-4	0.016
Swirl (B)	Shape score	Frozen VAR e-process	+0.374	7/7	1e-5	0.016
Swirl (B)	Shape score	Frozen MLP e-process	+0.671	7/7	7e-6	0.016
Swirl (B)	Shape score	Exact-LR oracle	-0.016	3/7	0.470	0.688
Shape (A)	One-step residual	Scan-B	+0.839	7/7	2e-9	0.016
Shape (A)	One-step residual	GLR-CUSUM	+0.799	7/7	6e-8	0.016
Shape (A)	One-step residual	Frozen VAR e-process	+0.895	7/7	2e-9	0.016
Shape (A)	One-step residual	Frozen MLP e-process	+0.888	7/7	2e-9	0.016
Shape (A)	One-step residual	Exact-LR oracle	+0.002	4/7	0.935	0.812
Shape (A)	Shape score	Scan-B	+0.860	7/7	3e-9	0.016
Shape (A)	Shape score	GLR-CUSUM	+0.819	7/7	1e-7	0.016
Shape (A)	Shape score	Frozen VAR e-process	+0.915	7/7	6e-10	0.016
Shape (A)	Shape score	Frozen MLP e-process	+0.908	7/7	3e-10	0.016
Shape (A)	Shape score	Exact-LR oracle	+0.022	4/7	0.180	0.219
Shape (A)	Trajectory density ratio	Scan-B	+0.828	7/7	4e-9	0.016
Shape (A)	Trajectory density ratio	GLR-CUSUM	+0.787	7/7	1e-7	0.016
Shape (A)	Trajectory density ratio	Frozen VAR e-process	+0.884	7/7	2e-10	0.016
Shape (A)	Trajectory density ratio	Frozen MLP e-process	+0.876	7/7	6e-10	0.016
Shape (A)	Trajectory density ratio	Exact-LR oracle	-0.010	3/7	0.437	0.781
Shape (B)	One-step residual	Diffusion latent MMD	+0.180	6/7	0.076	0.031
Shape (B)	One-step residual	Scan-B	+0.838	7/7	2e-9	0.016
Shape (B)	One-step residual	GLR-CUSUM	+0.822	7/7	5e-9	0.016
Shape (B)	One-step residual	Frozen VAR e-process	+0.900	7/7	7e-11	0.016
Shape (B)	One-step residual	Frozen MLP e-process	+0.893	7/7	4e-11	0.016
Shape (B)	One-step residual	Exact-LR oracle	+0.003	4/7	0.850	0.938
Shape (B)	Shape score	Diffusion latent MMD	+0.185	7/7	0.065	0.016
Shape (B)	Shape score	Scan-B	+0.843	7/7	9e-10	0.016
Shape (B)	Shape score	GLR-CUSUM	+0.827	7/7	1e-8	0.016
Shape (B)	Shape score	Frozen VAR e-process	+0.905	7/7	4e-12	0.016
Shape (B)	Shape score	Frozen MLP e-process	+0.898	7/7	7e-11	0.016
Shape (B)	Shape score	Exact-LR oracle	+0.008	4/7	0.533	0.688
Lin. rot. (A)	One-step residual	Scan-B	+0.912	7/7	3e-9	0.016
Lin. rot. (A)	One-step residual	GLR-CUSUM	+0.003	4/7	0.862	0.938
Lin. rot. (A)	One-step residual	Frozen VAR e-process	+0.005	4/7	0.731	0.578
Lin. rot. (A)	One-step residual	Frozen MLP e-process	-0.000	3/7	1.000	0.938
Lin. rot. (A)	One-step residual	Exact-LR oracle	-0.000	3/7	0.986	0.938
Lin. rot. (A)	Shape score	Scan-B	+0.919	7/7	2e-9	0.016
Lin. rot. (A)	Shape score	GLR-CUSUM	+0.010	4/7	0.258	0.297
Lin. rot. (A)	Shape score	Frozen VAR e-process	+0.012	3/7	0.392	0.438
Lin. rot. (A)	Shape score	Frozen MLP e-process	+0.007	4/7	0.576	0.469
Lin. rot. (A)	Shape score	Exact-LR oracle	+0.007	3/7	0.537	0.688
Lin. rot. (A)	Trajectory density ratio	Scan-B	+0.520	7/7	0.004	0.016
Lin. rot. (A)	Trajectory density ratio	GLR-CUSUM	-0.389	0/7	0.019	0.016
Lin. rot. (A)	Trajectory density ratio	Frozen VAR e-process	-0.387	0/7	0.014	0.016
Lin. rot. (A)	Trajectory density ratio	Frozen MLP e-process	-0.392	0/7	0.017	0.016
Lin. rot. (A)	Trajectory density ratio	Exact-LR oracle	-0.392	0/7	0.016	0.016

Table 14: Stored runs of the pre-registered protocol (separate trainings, the numbers of the internal report). Means over the seeds listed; (A)/(B) = config, device in brackets. These are replicates of Table 11, not the same models.

Setting	Detector (config, device)	seeds	mean	range
Swirl	native2 (A, MPS)	7	0.945	0.899–0.990
Swirl	flow LR, original divergence (A, MPS)	7	0.899	0.773–0.967
Swirl	frozen VAR (A, MPS)	7	0.120	0.037–0.221
Swirl	frozen MLP (A, MPS)	7	0.094	0.033–0.198
Swirl	shape score (A, CUDA)	7	0.945	0.880–0.977
Swirl	native2 (A, CUDA)	7	0.944	0.873–0.975
Swirl	native2 (B, CUDA)	7	0.952	0.922–0.973
Swirl	shape score (B, CUDA)	7	0.939	0.869–0.993
Swirl	GLR-CUSUM (B, CUDA)	7	0.454	0.292–0.617
Swirl	Scan-B (B, CUDA)	7	0.294	0.000–0.713
Swirl	frozen VAR (B, CUDA)	7	0.128	0.042–0.240
Swirl	frozen MLP (B, CUDA)	7	0.041	0.000–0.117
Swirl	diffusion MMD refl. (B, CUDA, separate training)	7	0.611	0.187–0.930
Shape	shape score (A, MPS)	7	0.963	0.920–0.978
Shape	native2 (A, MPS)	7	0.946	0.908–0.965
Shape	frozen VAR (A, MPS)	7	0.001	0.000–0.007
Shape	frozen MLP (A, MPS)	7	0.008	0.000–0.020
Shape	GLR-CUSUM (B, CUDA)	7	0.028	0.018–0.048
Shape	Scan-B (B, CUDA)	7	0.001	0.000–0.007
Shape	native2 (B, CUDA)	7	0.943	0.933–0.957
Shape	diffusion MMD refl. (B, CUDA, separate training)	3	0.463	0.340–0.593

Table 15: Per-seed detection at achieved PFA 0.05: nonlinear swirl, config A (file `metrics_swirl_l1_v2.json`).

Protocol	Detector	s0	s1	s2	s3	s4	s5	s6	mean
original	One-step residual, RESID (ours)	0.940	0.990	0.957	0.944	0.953	0.899	0.930	0.945
	Shape score (ours)	0.977	0.965	0.907	0.958	0.953	0.927	0.983	0.953
	Trajectory density ratio (ours)	0.927	0.953	0.978	0.920	0.971	0.967	0.964	0.954
	Scan-B (kernel MMD)	0.000	0.000	0.000	0.000	0.007	0.000	0.293	0.043
	GLR-CUSUM	0.387	0.487	0.447	0.567	0.380	0.510	0.440	0.460
	Frozen VAR e-process	0.097	0.221	0.077	0.104	0.173	0.130	0.037	0.120
	Frozen MLP e-process	0.104	0.080	0.050	0.053	0.198	0.138	0.033	0.094
	Exact-LR oracle (true latent)	0.967	0.962	0.970	0.953	0.967	0.955	0.983	0.965
	Density ratio, original divergence	0.887	0.950	0.953	0.915	0.847	0.773	0.967	0.899
aligned	One-step residual, RESID (ours)	0.940	0.993	0.965	0.962	0.963	0.903	0.923	0.950
	Shape score (ours)	0.973	0.968	0.929	0.953	0.953	0.938	0.987	0.957
	Trajectory density ratio (ours)	0.913	0.942	0.973	0.918	0.967	0.942	0.947	0.943
	Scan-B (kernel MMD)	0.377	0.430	0.593	0.520	0.538	0.673	0.707	0.548
	GLR-CUSUM	0.410	0.687	0.675	0.702	0.593	0.720	0.667	0.636
	Frozen VAR e-process	0.473	0.516	0.511	0.553	0.600	0.512	0.585	0.536
	Frozen MLP e-process	0.297	0.233	0.173	0.260	0.360	0.237	0.297	0.265
	Exact-LR oracle (true latent)	0.898	0.956	0.940	0.967	0.953	0.960	0.960	0.948
	Density ratio, original divergence	0.882	0.922	0.947	0.903	0.860	0.829	0.970	0.902

Table 16: Per-seed detection at achieved PFA 0.05: nonlinear swirl, config B (file `metrics_swirl_l25_v2.json`).

Protocol	Detector	s0	s1	s2	s3	s4	s5	s6	mean
original	One-step residual, RESID (ours)	0.993	0.952	0.956	0.943	0.927	0.979	0.927	0.954
	Shape score (ours)	0.955	0.959	0.877	0.844	0.940	0.957	0.928	0.923
	Diffusion latent MMD (reflected)	0.948	0.808	0.769	0.890	0.541	0.843	0.787	0.798
	Scan-B (kernel MMD)	0.396	0.507	0.007	0.787	0.387	0.521	0.504	0.444
	GLR-CUSUM	0.308	0.513	0.453	0.520	0.536	0.453	0.533	0.474
	Frozen VAR e-process	0.067	0.182	0.069	0.330	0.063	0.101	0.278	0.156
	Frozen MLP e-process	0.053	0.047	0.033	0.140	0.018	0.004	0.028	0.046
	Exact-LR oracle (true latent)	0.967	0.962	0.970	0.953	0.967	0.955	0.983	0.965
aligned	One-step residual, RESID (ours)	0.993	0.947	0.936	0.936	0.927	0.980	0.930	0.950
	Shape score (ours)	0.958	0.961	0.907	0.847	0.967	0.957	0.923	0.931
	Diffusion latent MMD (reflected)	0.973	0.889	0.917	0.902	0.796	0.880	0.880	0.891
	Scan-B (kernel MMD)	0.740	0.840	0.584	0.873	0.760	0.840	0.753	0.770
	GLR-CUSUM	0.524	0.693	0.727	0.615	0.641	0.583	0.660	0.635
	Frozen VAR e-process	0.494	0.532	0.524	0.561	0.575	0.558	0.660	0.558
	Frozen MLP e-process	0.280	0.210	0.187	0.393	0.334	0.113	0.308	0.261
	Exact-LR oracle (true latent)	0.898	0.956	0.940	0.967	0.953	0.960	0.960	0.948

Table 17: Per-seed detection at achieved PFA 0.05: shape switch, config A (file `metrics_shape_l1_v2.json`).

Protocol	Detector	s0	s1	s2	s3	s4	s5	s6	mean
original	One-step residual, RESID (ours)	0.958	0.922	0.957	0.959	0.953	0.965	0.908	0.946
	Shape score (ours)	0.973	0.970	0.920	0.956	0.978	0.973	0.970	0.963
	Trajectory density ratio (ours)	0.907	0.962	0.947	0.933	0.945	0.951	0.900	0.935
	Scan-B (kernel MMD)	0.000	0.000	0.000	0.007	0.000	0.000	0.003	0.001
	GLR-CUSUM	0.053	0.082	0.060	0.053	0.090	0.007	0.097	0.063
	Frozen VAR e-process	0.000	0.000	0.007	0.000	0.000	0.000	0.000	0.001
	Frozen MLP e-process	0.000	0.007	0.012	0.000	0.020	0.000	0.020	0.008
	Exact-LR oracle (true latent)	0.880	0.960	0.953	0.950	0.980	0.967	0.945	0.948
	Density ratio, original divergence	0.893	0.960	0.960	0.937	0.915	0.945	0.680	0.899
aligned	One-step residual, RESID (ours)	0.963	0.923	0.960	0.951	0.955	0.962	0.907	0.946
	Shape score (ours)	0.969	0.964	0.933	0.967	0.978	0.980	0.971	0.966
	Trajectory density ratio (ours)	0.903	0.960	0.940	0.935	0.918	0.953	0.930	0.934
	Scan-B (kernel MMD)	0.080	0.058	0.150	0.109	0.133	0.083	0.131	0.106
	GLR-CUSUM	0.220	0.159	0.227	0.093	0.121	0.057	0.150	0.147
	Frozen VAR e-process	0.047	0.053	0.068	0.029	0.007	0.060	0.091	0.051
	Frozen MLP e-process	0.035	0.087	0.049	0.045	0.107	0.033	0.050	0.058
	Exact-LR oracle (true latent)	0.913	0.940	0.950	0.919	0.907	0.980	1.000	0.944
	Density ratio, original divergence	0.917	0.960	0.950	0.935	0.913	0.950	0.807	0.919

Table 18: Per-seed detection at achieved PFA 0.05: shape switch, config B (file `metrics_shape_l25_v2.json`).

Protocol	Detector	s0	s1	s2	s3	s4	s5	s6	mean
original	One-step residual, RESID (ours)	0.940	0.936	0.952	0.960	0.973	0.970	0.933	0.952
	Shape score (ours)	0.973	0.967	0.953	0.960	0.960	0.960	0.973	0.964
	Diffusion latent MMD (reflected)	0.600	0.351	0.547	0.695	0.139	0.457	0.447	0.462
	Scan-B (kernel MMD)	0.000	0.000	0.000	0.007	0.000	0.000	0.007	0.002
	GLR-CUSUM	0.002	0.047	0.050	0.032	0.000	0.011	0.060	0.029
	Frozen VAR e-process	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	Frozen MLP e-process	0.008	0.006	0.000	0.000	0.000	0.000	0.000	0.002
	Exact-LR oracle (true latent)	0.880	0.960	0.953	0.950	0.980	0.967	0.945	0.948
aligned	One-step residual, RESID (ours)	0.908	0.944	0.956	0.952	0.963	0.972	0.933	0.947
	Shape score (ours)	0.953	0.953	0.936	0.940	0.953	0.958	0.968	0.952
	Diffusion latent MMD (reflected)	0.927	0.633	0.905	0.915	0.340	0.847	0.803	0.767
	Scan-B (kernel MMD)	0.127	0.054	0.129	0.112	0.073	0.142	0.127	0.109
	GLR-CUSUM	0.067	0.113	0.143	0.203	0.080	0.120	0.147	0.125
	Frozen VAR e-process	0.047	0.067	0.040	0.047	0.033	0.053	0.040	0.047
	Frozen MLP e-process	0.047	0.063	0.067	0.031	0.085	0.052	0.033	0.054
	Exact-LR oracle (true latent)	0.913	0.940	0.950	0.919	0.907	0.980	1.000	0.944

Table 19: Per-seed detection at achieved PFA 0.05: linear-rotation control, config A (file `metrics_linrot_l1_v2.json`).

Protocol	Detector	s0	s1	s2	s3	s4	s5	s6	mean
original	One-step residual, RESID (ours)	0.933	0.993	0.957	0.942	0.967	0.896	0.940	0.947
	Shape score (ours)	0.973	0.964	0.900	0.953	0.955	0.933	0.980	0.951
	Trajectory density ratio (ours)	0.000	0.602	0.609	0.720	0.350	0.213	0.052	0.364
	Scan-B (kernel MMD)	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	GLR-CUSUM	0.933	0.967	0.933	0.927	0.956	0.967	0.947	0.947
	Frozen VAR e-process	0.973	0.939	0.943	0.975	0.924	0.960	0.957	0.953
	Frozen MLP e-process	0.969	0.930	0.953	0.980	0.950	0.960	0.927	0.953
	Exact-LR oracle (true latent)	0.953	0.918	0.960	0.964	0.967	0.957	0.997	0.959
	Density ratio, original divergence	0.000	0.460	1.000	0.987	0.000	0.017	0.869	0.476
aligned	One-step residual, RESID (ours)	0.940	0.993	0.965	0.962	0.963	0.903	0.923	0.950
	Shape score (ours)	0.973	0.968	0.929	0.953	0.953	0.938	0.987	0.957
	Trajectory density ratio (ours)	0.040	0.813	0.853	0.809	0.647	0.367	0.377	0.558
	Scan-B (kernel MMD)	0.023	0.013	0.060	0.053	0.018	0.073	0.027	0.038
	GLR-CUSUM	0.981	0.953	0.942	0.921	0.927	0.953	0.953	0.947
	Frozen VAR e-process	0.924	0.936	0.949	0.953	0.967	0.962	0.926	0.945
	Frozen MLP e-process	0.957	0.953	0.973	0.887	0.960	0.940	0.980	0.950
	Exact-LR oracle (true latent)	0.973	0.949	0.953	0.963	0.913	0.960	0.940	0.950
	Density ratio, original divergence	0.000	0.590	0.947	0.903	0.002	0.033	0.942	0.488

Table 20: Per-seed detection at achieved PFA 0.05: swirl, naive MLP head without analytic base, config A (file `metrics_nobase_l1_v2.json`).

Protocol	Detector	s0	s1	s2	mean
original	One-step residual, RESID (ours)	0.947	0.952	0.927	0.942
	Shape score (ours)	0.953	0.973	0.957	0.961
	Scan-B (kernel MMD)	0.000	0.013	0.028	0.014
	GLR-CUSUM	0.020	0.060	0.050	0.043
	Frozen VAR e-process	0.000	0.000	0.000	0.000
	Frozen MLP e-process	0.000	0.000	0.000	0.000
	Exact-LR oracle (true latent)	0.967	0.962	0.970	0.966
aligned	One-step residual, RESID (ours)	0.953	0.959	0.927	0.946
	Shape score (ours)	0.957	0.973	0.968	0.966
	Scan-B (kernel MMD)	0.187	0.347	0.187	0.240
	GLR-CUSUM	0.067	0.153	0.107	0.109
	Frozen VAR e-process	0.000	0.000	0.007	0.002
	Frozen MLP e-process	0.004	0.000	0.007	0.004
	Exact-LR oracle (true latent)	0.898	0.956	0.940	0.931

Table 21: Exact-LR oracle per seed. *Shape*: corrected mixture-vs-Gaussian oracle on the true latent (detection, mean delay, mean per-step increment on calm and post-change data). *Window*: the same oracles with the restart mixture starting at $t = 0$ (full) or at the first full-context index (native), both settings.

	s0	s1	s2	s3	s4	s5	s6
Shape oracle: detection	0.880	0.960	0.953	0.950	0.980	0.967	0.945
Shape oracle: mean delay	32.9	36.3	34.2	34.2	32.9	33.4	35.5
Calm increment (nats/step)	-0.278	-0.273	-0.278	-0.280	-0.276	-0.277	-0.280
Post-change increment	+0.298	+0.287	+0.300	+0.294	+0.300	+0.301	+0.298
Post/calm energy ratio	1.004	0.997	1.002	0.996	0.990	1.010	0.996
Post-change kurtosis	4.68	4.65	4.72	4.64	4.72	4.69	4.71
Swirl oracle, full window	0.967	0.962	0.970	0.953	0.967	0.955	0.983
Swirl oracle, native window	0.838	0.920	0.873	0.913	0.893	0.880	0.920
Shape oracle, full window	0.880	0.960	0.953	0.950	0.980	0.967	0.945
Shape oracle, native window	0.913	0.940	0.936	0.919	0.907	0.980	1.000

Table 22: Trajectory density ratio with the original divergence ($\mathbf{1}^\top J\mathbf{1}$) and with the exact trace, config A, both protocols. The original-divergence rows reproduce the stored pre-registered runs exactly.

Setting	Variant	s0	s1	s2	s3	s4	s5	s6	mean
Swirl (original)	$\mathbf{1}^\top J\mathbf{1}$	0.887	0.950	0.953	0.915	0.847	0.773	0.967	0.899
Swirl (original)	exact trace	0.927	0.953	0.978	0.920	0.971	0.967	0.964	0.954
Swirl (aligned)	$\mathbf{1}^\top J\mathbf{1}$	0.882	0.922	0.947	0.903	0.860	0.829	0.970	0.902
Swirl (aligned)	exact trace	0.913	0.942	0.973	0.918	0.967	0.942	0.947	0.943
Shape (original)	$\mathbf{1}^\top J\mathbf{1}$	0.893	0.960	0.960	0.937	0.915	0.945	0.680	0.899
Shape (original)	exact trace	0.907	0.962	0.947	0.933	0.945	0.951	0.900	0.935
Shape (aligned)	$\mathbf{1}^\top J\mathbf{1}$	0.917	0.960	0.950	0.935	0.913	0.950	0.807	0.919
Shape (aligned)	exact trace	0.903	0.960	0.940	0.935	0.918	0.953	0.930	0.934
Lin. rot. (original)	$\mathbf{1}^\top J\mathbf{1}$	0.000	0.460	1.000	0.987	0.000	0.017	0.869	0.476
Lin. rot. (original)	exact trace	0.000	0.602	0.609	0.720	0.350	0.213	0.052	0.364
Lin. rot. (aligned)	$\mathbf{1}^\top J\mathbf{1}$	0.000	0.590	0.947	0.903	0.002	0.033	0.942	0.488
Lin. rot. (aligned)	exact trace	0.040	0.813	0.853	0.809	0.647	0.367	0.377	0.558

Table 23: Shape score before and after standardizing the inverse base point by its calm per-coordinate mean and standard deviation (stored runs of the pre-registered protocol).

Setting (seeds)	raw score	standardized score
Shape switch (0–6)	0.963	0.073–0.167
Nonlinear swirl (0–2)	0.964	0.298–0.559
Linear rotation (0)	0.973	0.932

Table 24: Cost of the trajectory density and error of the probability-flow ODE (`cost_and_discretization.py`, mps, 32 windows). *Left*: mean absolute error of the ODE log-density against the closed form available in the diagonal Gaussian case (✓ `ode_logdensity_base`), in nats. Euler is first order, midpoint second order at two velocity evaluations per step; the paper uses Euler with 16 steps. *Right*: wall-clock per monitored window with the exact Jacobian trace and with the $\mathbf{1}^\top J\mathbf{1}$ surrogate, by trajectory dimension (latent dimension $\times$ lags).

ODE error (nats)			ms per window, 16 steps			
steps	Euler	midpoint	traj. dim	exact	$\mathbf{1}^\top J\mathbf{1}$	ratio
2	0.54375	0.00707	32	7.38	0.58	13×
4	0.27256	0.00183	64	14.26	0.57	25×
8	0.13646	0.00046	128	28.25	0.58	49×
16	0.06828	0.00012				
32	0.03415	0.00003				
64	0.01708	0.00001				
128	0.00854	0.00002				

Table 25: Does the integrator change the *detection*? The same trained model per seed, the same episodes, scored with the trajectory density integrated five ways (`ode_ablation.py`, nonlinear swirl, config A). *calm* is the calm-test rate under the same rule.

Integrator	pre-registered	calm	calm boundary	calm
Euler, 8 steps	0.910	0.930	0.615	0.018
Euler, 16 steps (this paper)	0.927	0.935	0.533	0.015
midpoint, 4 steps	0.920	0.937	0.460	0.008

Table 26: The full campaign re-run at the horizon an ablation preferred, lags $(1, 2, 4)$ instead of the pre-registered $(1, 2, 4, 8)$ (7 seeds, config A, calm boundary; *calm* is the calm-test rate). The horizon was chosen on the swirl (Table 27), so the swirl column is partly in-sample and the other two are not. Table 1 remains the pre-registered configuration.

	Nonlinear swirl		Shape switch		Linear rotation	
Detector	det.	calm	det.	calm	det.	calm
One-step residual, RESID (ours)	0.031	0.011	0.007	0.007	0.953	0.011
Shape score (ours)	0.055	0.008	0.005	0.010	0.969	0.008
Trajectory density ratio (ours)	0.818	0.013	0.005	0.002	0.054	0.013
Scan-B (kernel MMD)	0.479	0.038	0.053	0.033	0.006	0.038
GLR-CUSUM	0.449	0.034	0.054	0.036	0.993	0.034
Frozen VAR e-process	0.682	0.019	0.093	0.021	0.976	0.019
Frozen MLP e-process	0.331	0.026	0.071	0.021	0.975	0.026
Exact-LR oracle (true latent)	0.987	0.042	0.972	0.025	0.985	0.028

Table 27: The prediction horizon, swept (`horizon_ablation.py`, nonlinear swirl, config A, retrained at every horizon). Detection at achieved PFA 0.05 under the calm boundary, mean over seeds with the per-seed range; *calm* is the calm-test rate. *s/ep* is the wall-clock of the trajectory density per episode. The oracle does not depend on the model and is shown as a constant reference. † marks the pre-registered default horizon.

Lags	d_{traj}	RATIO	range	calm	RESID	SHAPE	s/ep
(1)	8	0.710	0.513–0.867	0.026	0.034	0.083	0.88
(1, 2)	16	0.558	0.300–0.720	0.005	0.033	0.052	1.16
(1, 2, 4)	24	0.904	0.847–0.947	0.011	0.038	0.092	1.49
(1, 2, 4, 8)†	32	0.391	0.113–0.672	0.011	0.023	0.079	1.89
(1, 2, 4, 8, 16)	40	0.007	0.000–0.013	0.012	0.043	0.091	2.14
Exact-LR oracle	–	0.992	0.987–0.997	–	–	–	–

Table 28: Excess kurtosis at each stage of the pipeline, calm window against post-change window (`moment_pipeline.py`; shape switch, one trained model, pooled over episodes, time and coordinates). The change survives into the one-step code innovation and is diluted afterwards.

Stage	calm	post-change	change
Observation x_t	-0.015	0.020	+0.035
Observation innovation	-0.010	0.223	+0.233
Target code z_t	0.177	0.283	+0.105
Code innovation	0.131	0.492	+0.361
Trajectory y_t	0.198	0.270	+0.072
Flow residual $y_t - m_t$	0.103	0.155	+0.052
Inverse base point u_t	0.093	0.165	+0.071

Table 29: Analytic-base ablation on the swirl, config A, seeds 0–2, both protocols. VAR R^2: one-step linear predictability of the calm code.

Head	Protocol	VAR R^2	frozen VAR	RESID
base + residual	original	0.696	0.132	0.962
base + residual	aligned	0.691	0.500	0.966
naive MLP	original	0.066	0.000	0.942
naive MLP	aligned	0.064	0.002	0.946

Table 30: How wide must an energy-free change be before a detector that knows it can see it, on a real recording? Two-point innovation scale mixture $s \in \{c, \sqrt{2-c^2}\}$ injected into the UCI household power record; $c = 1/2$ is the mixture of the synthetic benchmark. *energy* is the realized post-change innovation energy divided by the energy the same window would have had without the injection — one by construction. The reference is the log-likelihood ratio that knows the injection (Section R). * marks the spread adopted for the experiment of Table 31.

c	$E[s^4]$	energy	kurtosis	reference	calm
0.50	1.56	1.000000	55→90	0.046	0.003
0.35	1.77	1.000000	55→102	0.132	0.003
0.25	1.88	1.000000	55→108	0.373	0.000
0.20*	1.92	1.000000	55→111	0.584	0.000
0.15	1.96	1.000000	55→113	0.767	0.000
0.10	1.98	1.000000	55→114	0.877	0.000

Table 31: An energy-free change injected into a real recording (Section R): every detector under the pre-registered rule and under the calm boundary, each beside its calm-test twin, means over seeds. *drift* is the mean per-step increment on calm holdout episodes.

	pre-registered		calm boundary		
Detector	det.	calm	det.	calm	drift
One-step residual, RESID (ours)	0.781	0.790	0.020	0.009	7.39
Shape score (ours)	0.738	0.722	0.060	0.011	71.57
Trajectory density ratio (ours)	0.908	0.930	0.022	0.004	86.94
Scan-B (kernel MMD)	0.167	0.082	0.104	0.044	0.00
GLR-CUSUM	0.202	0.087	0.063	0.013	0.79
Frozen VAR e-process	0.180	0.068	0.038	0.002	0.24
Frozen MLP e-process	0.306	0.059	0.127	0.002	0.24
Injection-aware reference	0.855	0.039	0.627	0.007	-0.01

Table 32: Public TCPD pilot. Every method is fitted on the longest segment between annotated change points and thresholded at the 95th percentile of its calm statistic; alarms within 5 steps of an annotation count as hits. "collapsed": no checkpoint passed the non-collapse constraint.

Series	CUSUM F_1	CUSUM alarms	F-JEPA-RD status	F-JEPA-RD F_1
`brent_spot`	0.009	223	trained	0.000
`well_log`	0.051	269	collapsed	–
`usd_isk`	0.016	121	collapsed	–
`co2_canada`	0.108	70	collapsed	–
`nile`	0.041	48	trained	0.000
`bank`	n/a	15	trained	–

Table 33: Pre-registered predictions and outcomes.

	Prediction	Outcome
P1	Frozen detector: F-JEPA-RD at least as good as a deterministic JEPA and no collapse of the nonlinear track.	**Not met as stated.** Under the original protocol and the pre-registered rule the frozen VAR reaches 0.120 on our codes against 0.138 on the deterministic JEPA's, and the nonlinear track does not collapse; but that comparison rests on the rule Section H retires. Re-run under the aligned protocol and the calm boundary it does *not* survive: the two representations are statistically indistinguishable for this detector (0.719 vs. 0.736, paired $p = 0.393$ over 7 seeds), though our codes remain more linearly predictable (VAR R^2 0.701 vs. 0.675, $p = 0.008$).
P2	Native likelihood-ratio detector beats the state-fed linear bracket (0.419/0.389).	Held, and it survives the correction: density ratio 0.954 (original protocol) and 0.943 (aligned) under the pre-registered rule, 0.626 under the calm boundary - still clear of the bracket.
P3	Selectivity: no gain on the linear-rotation control.	Held (Figure 13a).
P4	Sanity ceiling: nothing exceeds the exact-LR oracle.	Not a valid criterion for this metric: per-seed rates exceed the oracle within the saturation band; the oracle stays fastest (Section J).
P5	Public benchmarks: beat or tie baselines on $\geq$ 2 public suites.	Not met (Section S).
P6	$\geq$ 7 seeds with paired t and Wilcoxon tests.	Held as a procedure: every main result uses $\geq$ 7 seeds. The tests themselves (Table 13) were computed under the pre-registered rule and are not evidence for the paper's claims; see that table's caption.