Two Channels of Sequential Evidence: Why a Better Predictive Representation Did Not Detect Change Better

Anonymous authors
Paper under double-blind review

Abstract

A representation that predicts the future well ought to make a change in the world easier to notice. We set out to demonstrate exactly that, on a benchmark built so that only genuinely nonlinear structure can survive, with the detector held fixed so that every representation is judged on identical terms and with the interpretation of every outcome fixed before the numbers were seen. The result was the opposite of the one we expected. The learned predictive representation recovered the underlying state better than anything else we trained, and still detected change no better than a linear method from the subspace-identification literature that costs a single matrix factorisation and no gradient steps at all; under a second, equally legitimate detector the ordering reversed outright. Chasing that failure proved more informative than the success would have been. We show that the evidence a sequential test accumulates splits, exactly and by inspection of the test itself rather than by modelling it, into two separate channels: one that keeps paying as evidence accrues, and one that pays a bounded amount and then stops. Whether a representation helps is a question of which channel it opens, and every representation we trained left the paying channel shut while the true state opened it. From this we prove that a representation cannot manufacture evidence at all: it can only fail to destroy it, or move it into a form the test is able to read. That turns two of our most damaging measurements from embarrassments into predictions. We further find that the ordering of representations is a property of the extreme tail of the test statistic rather than of any average, so an entire family of cheap diagnostics, including one we pre-registered and one we had to retract mid-study, could never have worked. We report the negative result, the machine-checked theory that explains it, and a measurement that turns "which channel does this benchmark open" from an assumption into an answerable question.

1 Introduction

Self-supervised predictive objectives are widely believed to produce representations that expose the structure of a process rather than its surface. The Joint-Embedding Predictive Architecture (LeCun, 2022; Assran et al., 2023) makes that belief explicit: predict in representation space, not observation space, and the representation should retain what is predictable and discard what is not. If that is right, then a task whose difficulty is precisely *noticing that the process changed* ought to be a natural beneficiary.

We tested it. We built a benchmark whose regime change is invisible to any first-order statistic by construction, fixed a sequential detector from the anytime-valid testing literature (Ville, 1939; Ramdas et al., 2023), passed every representation through that same detector, and pre-registered what would count as success. The headline is negative: a JEPA-style representation recovers the latent state far better than the alternatives we trained and nonetheless fails to convert that into detection, losing to a classical linear predictive-state construction (Littman et al., 2001; Van Overschee & De Moor, 1996) that requires one singular value decomposition and no training at all.

The interesting part is why. Our central observation is that the detector's log-evidence is not a monolithic quantity to be maximised but a sum of two groups of terms that behave completely

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.

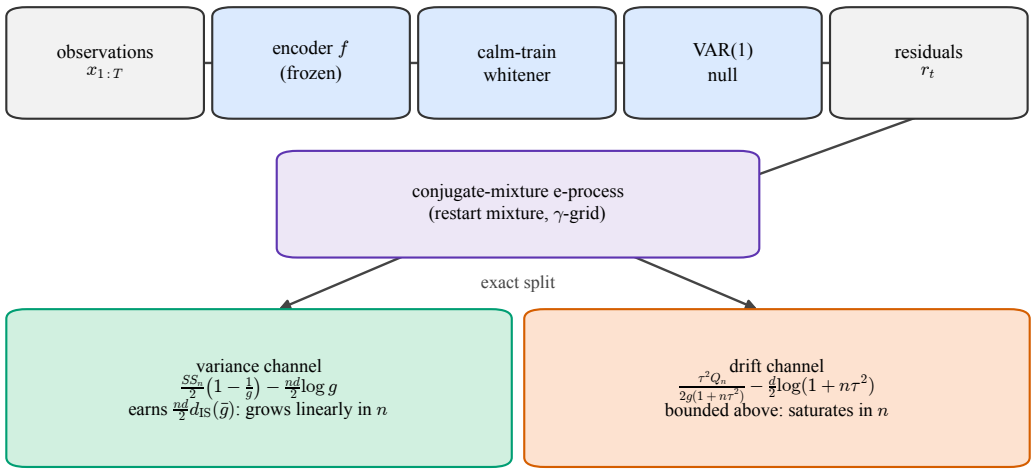


Figure 1: Every representation passes through the same frozen pipeline, and the resulting log-evidence splits *exactly* into two channels. The split is two disjoint groups of terms in the estimator, not a model of it; it is verified against the implementation to floating-point precision (Appendix A).

differently, and the split is exact - it is obtained by reading the estimator, not by approximating it. One group earns a divergence per unit of run length and therefore *accumulates*; the other is bounded above however long the run, and therefore *saturates*. Figure 1 shows where the split comes from.

Once the two channels are separated, the negative result stops being a disappointment about one architecture and becomes a statement about what sequential detectors can see. Our contributions:

1. **An exact two-channel decomposition** of a conjugate-mixture e-process (§3), with the variance channel's earnings identified as an Itakura-Saito divergence (Itakura & Saito, 1968) and the drift channel proved to saturate. The analytic content is machine-checked in Lean 4 (de Moura & Ullrich, 2021; The mathlib Community, 2020); the split itself is checked numerically against the implementation.
2. **A derivation of a previously anomalous measurement.** The decomposition implies that widening the representation bottleneck cannot help and generically hurts (Corollary 4) - which is what had been measured, counter-intuitively, and left unexplained.
3. **An alignment theorem** (§3.1): by data processing, no representation can create evidence; it can only avoid destroying it or move it into the detector's expressible class. Detectability is therefore detector-relative *by definition*, which converts our most damaging measurement - an ordering that inverts under a second detector - into a prediction.
4. **A negative result about cheap diagnostics** (§4.3). We pre-registered a detectability functional derived from the decomposition and it failed; so did two others, one of which we retracted mid-study. We then show why: an assumption-free scan over the same statistics reproduces the detector's ordering exactly, so the statistics are sufficient and what no summary captures is the upper tail of the null maximum.
5. **A measurement, not an assumption.** Which channel a benchmark opens is measurable, and we argue it belongs in a benchmark's specification.

2 Background and Related Work

Predictive representation learning. Joint-embedding methods avoid reconstruction by predicting representations of one view from another (Chen et al., 2020; Grill et al., 2020; Bardes et al., 2022), with JEPA (LeCun, 2022; Assran et al., 2023) making the predictive formulation central; contrastive predictive coding (van den Oord et al., 2018) shares the motivation. These methods are evaluated overwhelmingly by downstream accuracy on recognition tasks. Our question is different: whether

the representation makes a *sequential decision* easier, which is not implied by, and as we show does not follow from, better state recovery. Recent work makes the latent generative structure explicit, either variationally (Gögl & Yau, 2026) or through flow matching over latent trajectories (Huo et al., 2026); §5 explains why our analysis predicts these are the interesting directions and why our results do not settle them.

Predictive state representations. A predictive state is a sufficient statistic of the past for the future (Littman et al., 2001; Boots et al., 2011); its canonical linear construction is regularised canonical correlation between a past and a future window, i.e. the subspace-identification realisation of a linear dynamical system (Van Overschee & De Moor, 1996). This is the natural non-neural answer to the question JEPA poses, and it is the baseline we find hardest to beat.

Sequential testing and change detection. Classical quickest-detection procedures (Page, 1954; Lorden, 1971; Lai, 1998) assume a known likelihood ratio. We use the modern game-theoretic formulation, in which evidence is a nonnegative martingale under the null (Ville, 1939; Ramdas et al., 2023; Grünwald et al., 2024; Howard et al., 2021; Wasserman et al., 2020), because it gives anytime validity without a stopping correction. Offline change-point detection is surveyed by Truong et al. (2020) and benchmarked by van den Burg & Williams (2020); spectral drift detection (Hinder et al., 2022) targets kernel-matrix block structure. What distinguishes our setting is that the detector is held *fixed* across representations, so that differences are attributable to the representation rather than to a jointly tuned pair.

Geometry. Our alignment result composes the transportation inequality of Talagrand (1996) (see also Villani, 2008) with the data-processing inequality (Cover & Thomas, 2006); the Gaussian W_2 formula of Gelbrich (1990) appears in the appendix.

3 The Two-Channel Decomposition

Setting. A representation is a map f applied to observations; the resulting latent is whitened with calm-training statistics, a first-order vector autoregression is fitted on calm data as the null, and the residuals $r_1, \dots, r_n$ are fed to a conjugate-mixture e-process with inverse null covariance A, mixing parameter τ^2, and a fixed grid of variance-inflation hypotheses g. Write $SS_n = \sum_t r_t^\top A r_t$ for the residual energy, $S_n = \sum_t r_t$, and $Q_n = S_n^\top A S_n$. Full definitions are in Appendix D.

Proposition 1 (Exact split)**.** *The log-evidence contributed by a run of length n at grid point g is*

$$\underbrace{\tfrac{SS_n}{2}\left(1-\tfrac{1}{g}\right) - \tfrac{nd}{2}\log g}_{\textit{variance channel } V_g(n)} + \underbrace{\frac{\tau^2 Q_n}{2g(1+n\tau^2)} - \frac{d}{2}\log(1+n\tau^2)}_{\textit{drift channel } M_g(n)}. \tag{1}$$

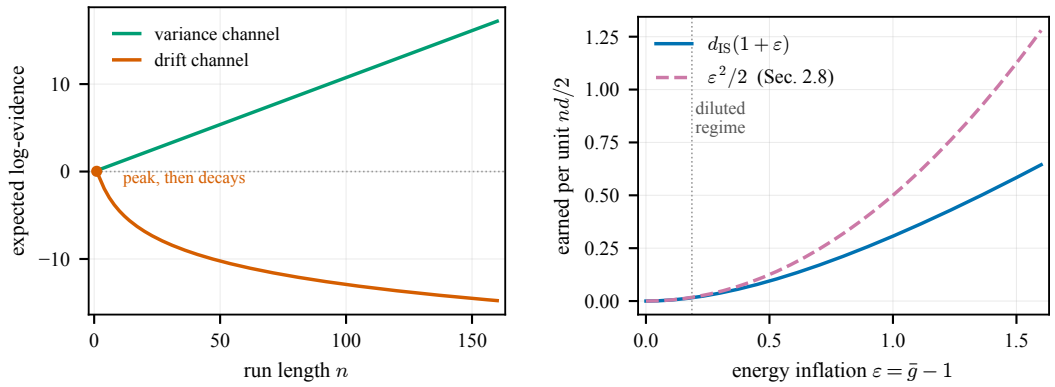
Proposition 1 is an identity of the estimator: the two brackets are disjoint groups of terms in the implemented expression. We therefore do not prove it, we *check* it, against the implementation to $|\Delta| \le 3.6 \times 10^{-15}$ (Appendix A).

Lemma 2 (Variance channel earns a divergence)**.** *If the post-change residual energy is inflated by a factor $\bar{g}$ in the detector's own metric, so that $\mathbb{E}[SS_n] = nd\,\bar{g}$, then $\mathbb{E}[V_g(n)]$ is maximised over $g > 0$ at $g = \bar{g}$, with value $\frac{nd}{2} d_{\mathrm{IS}}(\bar{g})$, where $d_{\mathrm{IS}}(x) = x - 1 - \log x \ge 0$ is the Itakura-Saito divergence, zero if and only if $x = 1$.*

Two consequences matter. First, the channel earns *nothing at all* unless the energy genuinely inflates - a representation with $\bar{g} = 1$ gets zero from it however long the run. Second, what it earns grows *linearly* in n.

Lemma 3 (Drift channel saturates)**.** *For a stationary residual process with long-run covariance Ω, the positive part of $\mathbb{E}[M_g(n)]$ is bounded above by $\mathrm{tr}(A\Omega)/(2g)$ uniformly in n, while its price $\frac{d}{2}\log(1+n\tau^2)$ increases without bound.*

So the drift channel has a finite optimal window and then decays. A detection that lives on it is a *scan* over a bounded statistic, not an accumulation. Figure 2a contrasts the two.



(a) One accumulates, one saturates.

(b) What the variance channel earns.

Figure 2: **Left:** expected contribution of each channel against run length. **Right:** the Itakura-Saito divergence of Lemma 2 against energy inflation, with the second-order truncation that a previous analysis used. The truncation is accurate only in the diluted regime, and the appendix quantifies its error.

Corollary 4 (Bottleneck monotonicity)**.** *Spreading a fixed amount of change over more latent dimensions lowers $\bar{g}$ and hence strictly lowers $d_{\mathrm{IS}}(\bar{g})$. Adding latent dimensions that carry no part of the change also raises the drift channel's price and the null spread of both statistics. Expected evidence at fixed change content is therefore non-increasing in the latent width.*

Corollary 4 *derives* a measurement that was already in hand and had no explanation: widening the bottleneck makes detection worse (Appendix I). Under any account in which representations help by retaining information, that result is perverse; here it is forced.

3.1 Representations Cannot Create Evidence

The detector never sees the observations, only $z = f(x)$. Applying the transportation inequality in representation space (Talagrand, 1996) gives a lower bound on expected evidence governed by the separation *after* the map and by the model's own prediction error in that space. The shared whitener neutralises the covariance factor by construction, so it is not what separates representations. The upper bound is the decisive one:

Proposition 5 (Alignment, not information)**.** *For any measurable f, $\mathrm{KL}(f_{\#}P \,\|\, f_{\#}Q) \leq \mathrm{KL}(P \,\|\, Q)$. Composed with the evidence identity, the achievable growth in representation space is capped by its raw-observation value.*

A representation can therefore only *fail to destroy* evidence, or move the change into directions the detector's alternative class can express. Representation learning for sequential detection is not an information problem; it is an alignment problem, and alignment is defined relative to a detector. This reframes rather than rescues: it predicts that an advantage measured under one detector need not survive another, which is exactly what we observe.

4 Experiments

Benchmark. The change is a state-dependent modulation of a rotation in a latent space observed through a fixed nonlinearity and padded with higher-variance nuisance channels. The construction matches the marginal law exactly and compensates the first-order autocovariance, so linear methods are blind by design and the only surviving signal is a *conditional second-moment* change (Appendix C). A linear-rotation control, in which the transition itself changes, is run throughout as a negative control.

Arms and protocol. We compare a JEPA-style predictive arm, an autoencoder, a repaired contrastive arm, PCA, raw observations, and a spectral predictive-state representation, against two

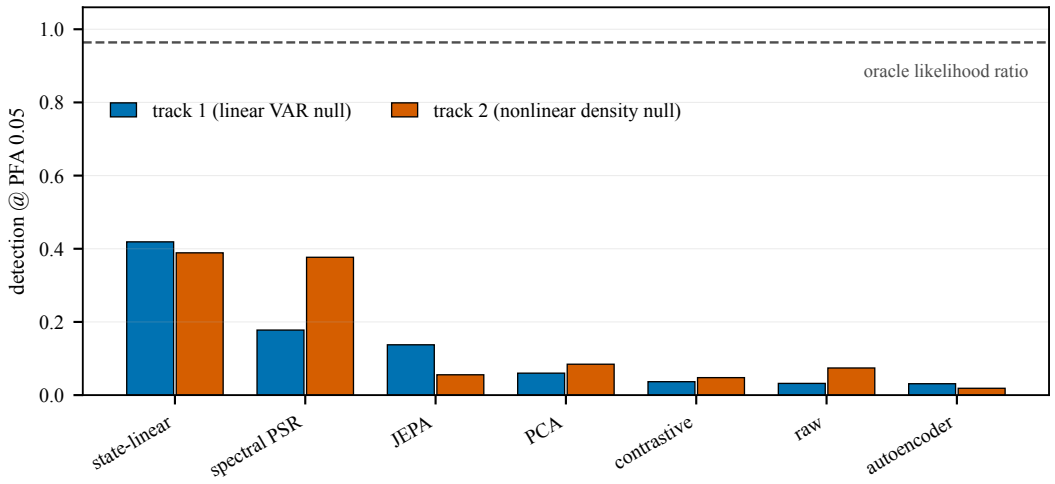


Figure 3: Detection at matched achieved false alarm, three seeds, both detector tracks. The learned predictive arm is beaten by a training-free spectral predictive-state representation, and the ordering between them *inverts* between tracks. All arms sit far below the exact likelihood ratio.

Table 1: Detection at matched achieved false alarm 0.05, three seeds, with every arm evaluated on identical episodes so that comparisons are paired. The exact conditional likelihood ratio of the generative model reaches 0.964; the gap from the best learned arm is $+0.805$, with a 95% interval of $[+0.732, +0.879]$. Per-seed spread is large (Appendix D), which is why the episodes are shared rather than merely matched in distribution.

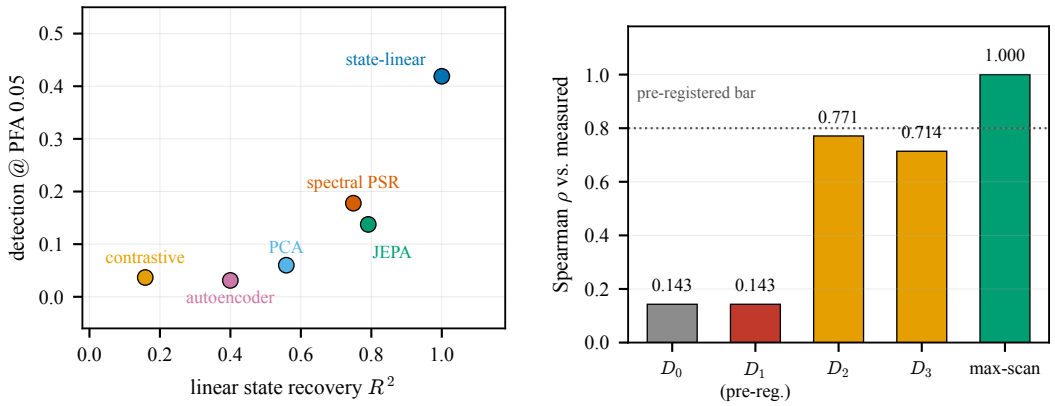
representation	track 1 (linear null)	track 2 (density null)
oracle likelihood ratio	0.964	-
true state (linear detector)	0.419	0.389
spectral PSR *(no training)*	**0.178**	**0.377**
JEPA	0.138	0.056
PCA	0.060	0.084
contrastive	0.037	0.048
raw observations	0.032	0.074
autoencoder	0.031	0.019

references: the true state fed to the same detector, and the exact conditional likelihood ratio of the generative model. All arms share the encoder budget, the whitener, the detector, the calibration episodes and the achieved-false-alarm estimand; detection is reported at a matched achieved false alarm rate of 0.05. Two detector tracks are used: the frozen linear null, and a matched nonlinear conditional-density null. Protocol, capacity matching and checkpoint selection are in Appendix D.

4.1 The Negative Result

Figure 3 and Table 1 are the result. Three things follow. The spectral PSR, which costs one SVD, matches or beats the learned arm on both tracks. The gap to the exact likelihood ratio is enormous for every arm, learned or not. And the ordering between the two best learned-or-classical arms *reverses* between detector tracks - under a pre-registered rule that required the benefit to survive a detector swap, this is a failure.

Figure 4a shows why it is not a story about representation quality. The learned predictive arm recovers the latent state better than every other trained arm and still detects poorly: state recovery and detectability are close to unrelated across our arms.



(a) Recovery does not imply detectability.

(b) Every moment-based surrogate fails.

Figure 4: **Left:** linear state recovery against detection. **Right:** rank correlation between each candidate detectability predictor and measured detection; only the assumption-free scan is exact (§4.3).

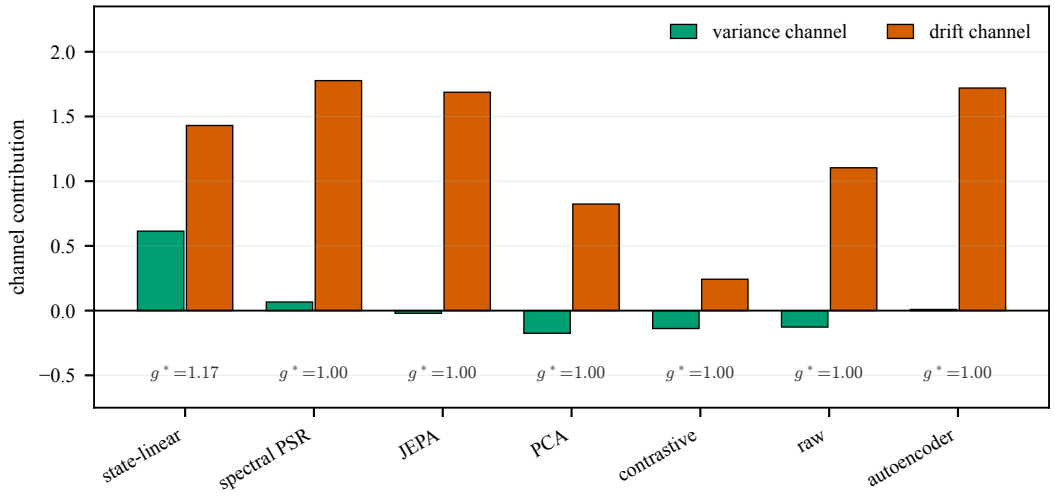


Figure 5: The decomposition of §3, measured per arm. Every learned and projected representation selects the null grid point, at which the variance channel is identically zero by Lemma 2, and lives entirely on the saturating channel. Only the true state opens the accumulating one.

4.2 Which Channel Is Open

Figure 5 is the mechanism. Every trained or projected representation selects the null inflation hypothesis, where the variance channel contributes exactly nothing, and therefore detects using a channel that Lemma 3 shows cannot accumulate. The true state does not: it opens the variance channel and consequently uses windows several times longer. The benchmark closes the variance channel in *observation* coordinates by construction, but the change is a genuine variance modulation in *state* coordinates, so the state re-opens it and no representation we trained recovered that. The gap to the state reference is therefore a difference of channel, not of degree.

4.3 Why Cheap Diagnostics Cannot Work Here

If the decomposition is right, it should yield a cheap predictor of detectability. We pre-registered one, with its success criteria fixed and committed before the scoring code existed, and **it failed**. So

did a dimension-corrected version of an earlier scalar, and so did two exploratory variants - one of which we published internally, then retracted after an ablation contradicted it. Figure 4b summarises.

The reason is instructive. An assumption-free scan over the *same* window statistics, run inside the correct calibration protocol, reproduces the detector's ordering exactly. The statistics are therefore *sufficient*; what no low-order summary captures is the upper tail of the null maximum, which we measure to be far from its Gaussian prediction and to differ substantially between arms (Appendix G). The ordering is a tail property, not a moment property. That is a general caution: when a decision threshold is calibrated on a *maximum*, no summary built from means and variances can rank its inputs, however exactly that summary is derived from the detector's own algebra.

A second ablation isolates the aggregation. Replacing the restart mixture with a plain maximum, changing nothing else, leaves the ordering exactly intact and detection almost unchanged (Appendix H). *The mixture buys anytime-validity, not power*, at least here - worth knowing before attributing detection gains to the aggregation rule.

5 Limitations

Our benchmark is synthetic. It was designed so that the question could be asked cleanly, at the cost of the transfer question being unresolved; the appendix records what would be required. A second setting we built turned out to be unable to discriminate *any* pair of representations, for reasons we can derive rather than guess (Appendix J); we report it as a negative about benchmark design rather than quietly dropping it.

We study one detector family in two instantiations. Proposition 5 says conclusions are detector-relative by construction, so this is a real limit, not a formality. Seed variance on this benchmark is large enough that early small-sample readings were misleading; our numbers use identical episodes across arms so that comparisons are paired, and we report where sample size is the binding constraint.

Finally, our analysis suggests where to look next but does not test it. Both the variational and flow-matching JEPA variants (Gögl & Yau, 2026; Huo et al., 2026) make a predictive *distribution* explicit, which is exactly what the dead variance channel lacks; and a trajectory-level likelihood would change what the residual is rather than how it is scored. Neither has been run here, and Proposition 5 constrains what either could claim: any gain must be argued as alignment, never as retained information.

6 Conclusion

We asked whether a predictive objective buys detectability and found that it does not, losing to a training-free classical construction and reversing under a change of detector. Pursuing the failure produced an exact decomposition of what a sequential detector can see, a proof that representations can only realign evidence rather than create it, the derivation of a previously anomalous measurement, and the finding that the ordering of representations lives in a tail rather than in any moment. The practical recommendation is narrow and we think durable: report which channel your benchmark opens, because a benchmark can be alive in one and dead in the other, and "we held the detector fixed" does not by itself make two settings comparable.

Impact Statement

This work is methodological and uses synthetic data only. Its likely effect is on evaluation practice: we argue that a representation should not be credited with improving a sequential decision unless the channel carrying that improvement is identified, and we give a negative result showing how easily the opposite conclusion is reached. Sequential change detection is deployed in monitoring settings where false alarms and missed detections carry real cost; we would regard uncritical transfer of a synthetic benchmark result to such a setting as a misuse of this work, and §5 states plainly that the transfer question is open. We identify no dual-use concern.

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

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

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

Byron Boots, Sajid M. Siddiqi, and Geoffrey J. Gordon. Closing the learning-planning loop with predictive state representations. *International Journal of Robotics Research*, 30(7):954–966, 2011.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, 2020.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2nd edition, 2006.

Leonardo de Moura and Sebastian Ullrich. The Lean 4 theorem prover and programming language. In *International Conference on Automated Deduction (CADE)*, 2021.

Matthias Gelbrich. On a formula for the L^2 Wasserstein metric between measures on Euclidean and Hilbert spaces. *Mathematische Nachrichten*, 147(1):185–203, 1990.

Moritz Gögl and Christopher Yau. Var-JEPA: A variational formulation of the joint-embedding predictive architecture – bridging predictive and generative self-supervised learning. *arXiv preprint arXiv:2603.20111*, 2026.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, et al. Bootstrap your own latent: A new approach to self-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Peter Grünwald, Rianne de Heide, and Wouter M. Koolen. Safe testing. *Journal of the Royal Statistical Society Series B*, 86(5):1091–1128, 2024.

Fabian Hinder, André Artelt, Valerie Vaquet, and Barbara Hammer. Precise change point detection using spectral drift detection. *arXiv preprint arXiv:2205.06507*, 2022.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021.

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.

Fumitada Itakura and Shuzo Saito. Analysis synthesis telephony based on the maximum likelihood method. In *Proceedings of the 6th International Congress on Acoustics*, pp. C17–C20, 1968.

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

Yann LeCun. A path towards autonomous machine intelligence. *OpenReview preprint*, 2022. Version 0.9.2.

Michael L. Littman, Richard S. Sutton, and Satinder Singh. Predictive representations of state. In *Advances in Neural Information Processing Systems (NIPS)*, 2001.

Gary Lorden. Procedures for reacting to a change in distribution. *The Annals of Mathematical Statistics*, 42(6):1897–1908, 1971.

E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954.

Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4):576–601, 2023.

Michel Talagrand. Transportation cost for Gaussian and other product measures. *Geometric and Functional Analysis*, 6(3):587–600, 1996.

The mathlib Community. The Lean mathematical library. In *ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP)*, 2020.

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

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

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Peter Van Overschee and Bart De Moor. *Subspace Identification for Linear Systems: Theory, Implementation, Applications*. Kluwer Academic Publishers, 1996.

Cédric Villani. *Optimal Transport: Old and New*. Springer, 2008.

Jean Ville. *Étude critique de la notion de collectif*. PhD thesis, Gauthier-Villars, Paris, 1939.

Larry Wasserman, Aaditya Ramdas, and Sivaraman Balakrishnan. Universal inference. *Proceedings of the National Academy of Sciences*, 117(29):16880–16890, 2020.

A Machine-Checked Verification

What is proved, and what is checked. Proposition 1 is an *identity of the estimator*: the two brackets in Eq. 1 are two disjoint groups of terms in the implemented log-evidence. It is therefore not a theorem to prove but a claim about code, and we check it numerically: for random residuals, a single restart and a single grid point, the sum of the two brackets agrees with the implementation to $|\Delta| \le 3.6 \times 10^{-15}$ at every grid point.

The analytic content built on top of the split is machine-checked in Lean 4 (de Moura & Ullrich, 2021) against `mathlib` (The mathlib Community, 2020). The module `CJEPATheory.ChannelDecomposition` proves:

Lean name	statement
`dIS_nonneg`	$d_{\mathrm{IS}}(x) \ge 0$ for $x > 0$
`dIS_eq_zero_iff`	$d_{\mathrm{IS}}(x) = 0 \iff x = 1$
`variance_channel_le`	$\bar{g}(1 - 1/g) - \log g \le d_{\mathrm{IS}}(\bar{g})$ for all $g > 0$
`variance_channel_at_gbar`	equality at $g = \bar{g}$
`variance_channel_dead`	$\bar{g} = 1 \Rightarrow$ channel ≤ 0 at every g
`drift_positive_part_lt`	drift positive part $\le \Omega/(2g)$, uniformly in n
`drift_price_mono`	the price is monotone in n
`dIS_mono`	$1 \le x \le y \Rightarrow d_{\mathrm{IS}}(x) \le d_{\mathrm{IS}}(y)$
`dilution_decreases_divergence`	$d_{\mathrm{IS}}(1 + c/d_2) \le d_{\mathrm{IS}}(1 + c/d_1)$ for $d_1 \le d_2$

The first two give Lemma 2's divergence and the "dead channel" statement; the next three give its maximisation; the two drift results give Lemma 3; the last two give the direction of Corollary 4. An axiom audit (`Verify.lean`) prints the dependencies of each: all of the above use only Lean's three standard axioms, with no `sorry`.

The audit output we obtained is shipped as `lean/AXIOM_AUDIT.txt`, with the toolchain and the pinned Mathlib revision recorded in its header and that revision pinned in `lake-manifest.json`. A reviewer can rerun `check.sh` and diff against it directly; it lists 36 theorems, of which the 9 above belong to this paper. We ship the sources, the pin and the audit rather than the build cache, which is several gigabytes and is fetched by `check.sh` in minutes.

Numerical suite. A separate script re-checks the same statements against the implementation and against the constants actually used, and **passes 23 of 23**. It exits non-zero on any failure and is safe to run in continuous integration. Two checks failed on a first run for a reason worth recording: they simulated $\mathbb{E}[SS_n]$ instead of using it exactly, and because the maximum of the variance channel is a small difference of large numbers, a 0.5% Monte-Carlo error moved it by several percent. Checking algebra against a simulated expectation tests the simulation, not the algebra.

A correction the verification forced. An earlier draft asserted that Lemma 2 reproduces a previously published growth law that had been "validated to 0.4%". That conflated two things. The 0.4% figure validates the *effective inflation* formula $\bar{g} = 1 + k(v-1)/d$ against measurement. The growth law $k^2(v-1)^2/(4d)$ is the *second-order truncation* of Lemma 2, with relative error $2\varepsilon/3 + O(\varepsilon^2)$ where $\varepsilon = \bar{g} - 1$. The numerical suite measures that error at 12.2% in the diluted regime ($\varepsilon = 0.186$) and 92% undiluted ($\varepsilon = 1.489$), both matching the predicted $2\varepsilon/3$ to within 0.002. Lemma 2 therefore *contains* the earlier law and delimits where it may be used, which is a stronger claim than reproducing it.

B Proofs

Proof of Lemma 2. With $\mathbb{E}[SS_n] = nd\bar{g}$, $\mathbb{E}[V_g(n)] = \frac{nd}{2}\big[\bar{g}(1 - 1/g) - \log g\big]$. Put $t = \bar{g}/g > 0$. Then $\bar{g}(1 - 1/g) - \log g = \bar{g} - t - \log \bar{g} + \log t$, so the claim $\bar{g}(1 - 1/g) - \log g \le \bar{g} - 1 - \log \bar{g}$ is exactly $\log t \le t - 1$, which holds for all $t > 0$ with equality iff $t = 1$, i.e. $g = \bar{g}$. Substituting $g = \bar{g}$ gives the value $\frac{nd}{2}(\bar{g} - 1 - \log \bar{g})$. Nonnegativity and the equality case are the same inequality at $t = x$. □

Proof of Lemma 3. For a stationary residual process, $\mathbb{E}[Q_n] = \mathrm{tr}(A\Omega_n)$ with $\Omega_n = \sum_{|h|<n}(n - |h|)\Gamma(h) \to n\Omega$. Hence the positive part of $\mathbb{E}[M_g(n)]$ tends to $\tau^2 n\,\mathrm{tr}(A\Omega)/\big(2g(1+n\tau^2)\big)$, which is increasing in n and bounded above by $\mathrm{tr}(A\Omega)/(2g)$ since $\tau^2 n \le 1 + n\tau^2$. The price $\frac{d}{2}\log(1+n\tau^2)$ is increasing and unbounded. □

Proof of Corollary 4. Let the total changed energy $c = k(v-1)$ be fixed and $\bar{g} = 1 + c/d$. For $d_1 \le d_2$, $c/d_2 \le c/d_1$, and d_{IS} is nondecreasing on $[1,\infty)$ because for $1 \le x \le y$, $d_{\mathrm{IS}}(y) - d_{\mathrm{IS}}(x) = (y-x) - \log(y/x) \ge (y-x) - \big((y-x)/x\big) \ge 0$, using $\log u \le u - 1$ and $x \ge 1$. Hence $d_{\mathrm{IS}}(1 + c/d_2) \le d_{\mathrm{IS}}(1 + c/d_1)$. The added dimensions contribute nothing to $\mathbb{E}[Q_n]$, raise the drift price by $\frac{\Delta d}{2}\log(1+n\tau^2)$, and raise the null standard deviation of both statistics as $\sqrt{d}$. □

Proof of Proposition 5. The data-processing inequality for f-divergences gives $\mathrm{KL}(f_{\#}P \| f_{\#}Q) \le \mathrm{KL}(P\|Q)$ for any measurable f (Cover & Thomas, 2006). The evidence identity expresses expected log-evidence as $\mathrm{KL}(P_t\|Q)$ minus the model's own predictive divergence; the first term is thus capped by its raw-observation value. □

C Benchmark Construction

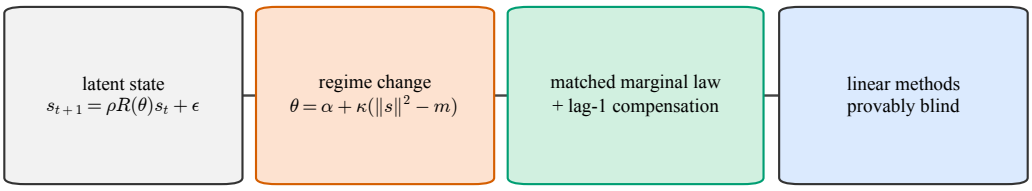


Figure 6: What the construction closes and what it deliberately leaves open.

A latent state of dimension 6 evolves as three planar rotations with decay $\rho = 0.9$ and innovation scale 0.4. Under the alternative the rotation angle becomes state dependent, $\theta = \alpha + \kappa(\|s\|^2 - m)$ with $\kappa = 0.17$, and the decay and innovation scale are adjusted so that the marginal law is matched *exactly* and the first-order autocovariance is compensated: the measured relative change in the fitted transition matrix is 0.8% against 96% for the linear-rotation control, and the autocorrelation gap is 0.004. The state is observed through a fixed random two-layer $\tanh$ map into 24 channels with observation noise 0.15, then padded with 32 pure-noise channels of *higher* marginal variance, giving $d_x = 56$. Episodes are 300 steps with the change at step 150; the null window for the false-alarm estimand starts at step 12. The bottleneck is $d_z = 8$ throughout, above the intrinsic state dimension and far below d_x, so every arm must choose what to keep.

D Protocol

Detector. Latents are whitened with calm-*training* mean and covariance (with shrinkage), a VAR(1) null is fitted on calm training latents, and residuals are scored by a conjugate-mixture e-process with $\tau^2 = 1$, geometric restart weights with ratio 0.95, and inflation grid $\{1.0, 1.5, 2.5, 4.0\}$. Detection is read at achieved false alarm 0.05 by interpolating the achieved-PFA curve, and we log whether that operating point is inside the achievable span, because "never detects" and "never measured here" are different failures and were once confused. Track 2 replaces the linear null with a matched nonlinear conditional-density null (a two-hidden-layer network predicting mean and log-scale), fitted on calm data only with early stopping; the evidence machinery, restart weights, grid and estimand are unchanged.

Arms and capacity. All neural arms share depth, width, optimiser, schedule and step budget (three layers, width 64, 2000 steps, batch 256); the set-structured arm matches the flat arm's parameter count to 1.13%. Checkpoint selection is a *feasibility constraint* followed by normalised future-prediction error: selecting on the training loss selects collapse, because collapse drives the prediction term to zero while the variance term is bounded. A run with no feasible checkpoint is recorded as an optimisation failure rather than fed to the detector.

Splits and pairing. Fit, validation, calibration, holdout and test episodes are drawn with fixed, distinct seeds, and *every arm sees identical episodes*, so all comparisons are paired. Detection differences on this benchmark have large between-seed spread - a single arm ranges from 0.020 to 0.300 across seeds - so paired comparison is not a refinement but a requirement.

E Full Results

The main-text table is Table 1; the channel decomposition follows.

Table 2: Channel decomposition per arm (track 1, three seeds). g^* is the selected grid point and n^* the selected window. Only the true state selects an inflation hypothesis above the null, and it is the only arm with a positive variance channel of any size.

representation	$\bar{g}$	per-seed $\bar{g}$	g^*	n^*	variance / drift
true state	1.034	$1.112/1.005/0.986$	1.17	53	$+0.61$ / $+1.43$
spectral PSR	0.986	$1.022/0.983/0.954$	1.00	11	$+0.07$ / $+1.78$
JEPA	0.988	$0.996/0.993/0.977$	1.00	12	-0.02 / $+1.69$
autoencoder	0.974	$1.017/0.955/0.951$	1.00	10	$+0.01$ / $+1.72$
PCA	0.911	$0.929/0.899/0.904$	1.00	11	-0.18 / $+0.82$
raw	0.975	$0.978/0.973/0.973$	1.00	103	-0.13 / $+1.10$
contrastive	0.920	$0.948/0.882/0.931$	1.00	51	-0.14 / $+0.24$

The per-seed column is the honest one. Even the true state is not uniformly above unity - one of its three seeds reads 0.986 - so $\bar{g}$ at three seeds does not cleanly separate the arms. What does separate them is (g^*, n^*), which is stable.

F The Surrogate Ladder in Full

We considered four moment-based predictors of detectability and one assumption-free scan.

- D_0: a dimension-corrected version of an earlier scalar, using the post/calm ratio of the drift statistic over one long window.
- D_1 (**pre-registered**): the maximum over grid points and window lengths of the expected post-minus-calm channel contribution divided by its null standard deviation. Success criteria - rank correlation at least 0.8 with the true state ranked first, plus reproduction of the detector-swap inversion - were committed before the scoring code existed.
- D_2 (exploratory): as D_1 but comparing against a Gaussian approximation to the null *maximum*, and dividing by the *post-change* spread.
- D_3 (exploratory): as D_2 but with the threshold taken *empirically* from calm calibration episodes.
- Max-scan: the exact per-window statistic, maximised, inside the real calibration protocol; no distributional assumption at all.

D_1 failed its first criterion under both targets and is recorded as failed. D_2 appeared to predict detection *rates*, not merely ranks, to within 0.03 on five of six arms - but §G shows that agreement is two large errors partly cancelling, and D_3 demonstrates it: with a correct empirical threshold, every arm except the true state is now *under* predicted (PSR 0.015 against 0.160 measured; JEPA 0.018 against 0.071), where D_2 over-predicted them. Both the null maximum and the post-change statistic are tail dominated.

Table 3: Rank correlation with measured detection. "Same-seed" scores the predictor against detection measured on the very same latents; "banked" uses independently collected multi-seed means. The verdict is identical under both.

	assumes	t1 same	t1 banked	t2 same	top arm
D_0	Gaussian, 1 window	$+0.143$	$+0.029$	$+0.371$	PSR (fails)
D_1 **(pre-reg.)**	Gaussian null sd	$+0.143$	$+0.200$	$+0.429$	PSR (fails)
D_2	Gaussian, both	$+0.771$	$+0.771$	$+0.600$	true state
D_3	Gaussian post	$+0.714$	-	-	true state
max-scan	**nothing**	$\mathbf{+1.000}$	-	-	true state

G The Null Maximum Is Not Gaussian

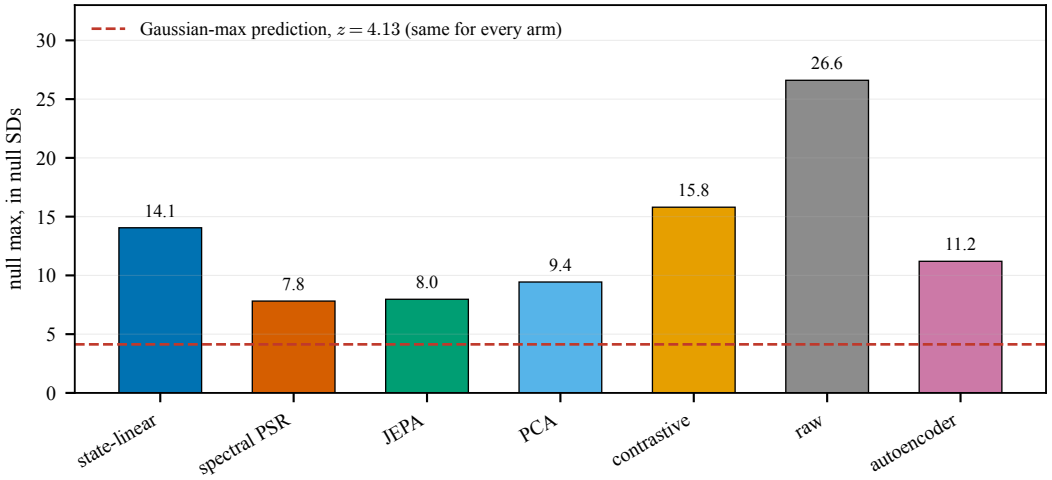


Figure 7: Where the null maximum actually sits, in units of the null standard deviation, against the value a Gaussian-maximum argument predicts for every arm alike. The residual energy carries excess kurtosis between $+1.7$ and $+3.7$ for all arms except raw observations.

The empirical null maximum sits between 7.8 and 26.6 null standard deviations above the mean, where the Gaussian-maximum argument predicts 4.13 identically for every arm. This is the single reason every moment-based surrogate had to fail, and it is why we present the failure as a structural result rather than as a search that has not yet succeeded.

H Aggregation Ablation

We ran the real pipeline - same whitener, null, calibration and holdout episodes, same achieved-PFA estimand - with exactly one line changed: the log-sum-exp over restarts and grid points became a maximum. Rank correlation between the two is $+1.000$ over six arms, every absolute difference is at most 0.016, and the best arm is unchanged. The restart mixture buys anytime validity, not power. This also refutes an intermediate hypothesis of ours - that the aggregation was where the true state's advantage lived - and inverts its conclusion: because the max variant *is* a scan over the same two statistics inside the correct protocol, and it reproduces the ordering exactly, the two channel statistics are *sufficient*.

I Bottleneck Sweep

Measured detection is $0.180, 0.174, 0.031, 0.017$ on track 1 and $0.225, 0.367, 0.112, 0.052$ on track 2 for widths $4, 8, 16, 32$. The decrease above the peak is what Corollary 4 predicts and is counter-intuitive under any account in which representations help by retaining information. We report the

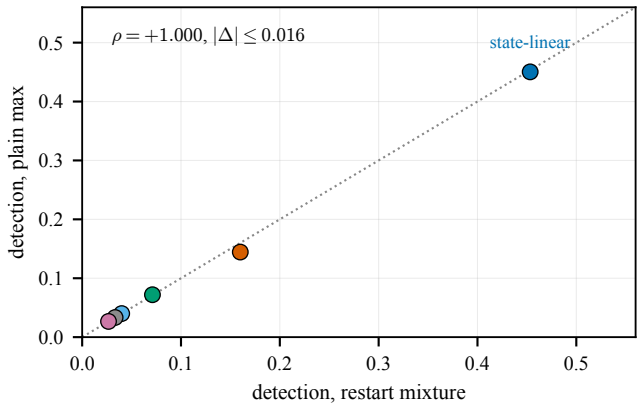


Figure 8: Replacing the restart mixture with a plain maximum, changing nothing else in the pipeline, moves no arm by more than 0.016 and leaves the ordering exactly intact.

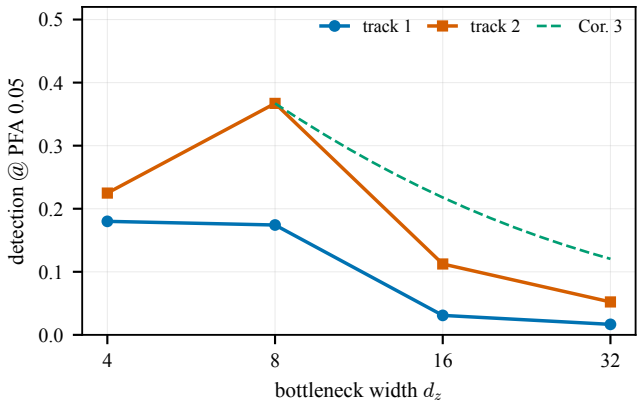


Figure 9: Detection against bottleneck width for the spectral PSR, three seeds. Above the peak, a wider bottleneck detects strictly worse, as Corollary 4 requires. The rise from the narrowest width on track 2 is outside the corollary's scope, which concerns adding dimensions that carry no part of the change.

non-monotonicity at the narrowest width rather than cropping the axis: at width 4 the bottleneck is below the intrinsic state dimension, so the comparison is no longer "the same change spread over more dimensions".

J A Second Benchmark That Cannot Discriminate

We built a second, object-structured setting and every arm sat at the detection floor. That is not an architecture failure, and it can be derived. The change inflates the variance of k latent dimensions out of d, so the effective inflation is $\bar{g} = 1 + k(v-1)/d$ and, by Lemma 2, growth per step is $\frac{d}{2} d_{\mathrm{IS}}(\bar{g})$. Once $\bar{g}$ falls below the smallest non-null grid point, no grid point beats the null and evidence stops accumulating entirely. The measured collapse occurs exactly there.

Crucially, the identity of the group that will change is *not identifiable from calm data*: in the calm regime all groups are exchangeable. Any representation fitted on calm data therefore allocates capacity independently of which group will later change, so in expectation the changed group occupies a $1/m$ share of the latent, which is precisely the diluted regime. We verified the one loophole - that a tight bottleneck might force the encoder to keep only a couple of groups, so that some seeds get lucky - and closed it: every arm spreads over 5.6 to 7.7 of 8 groups. The permutation-invariant arm is the *most* uniform of all, which is exactly the wrong thing to be on a dilution benchmark: an

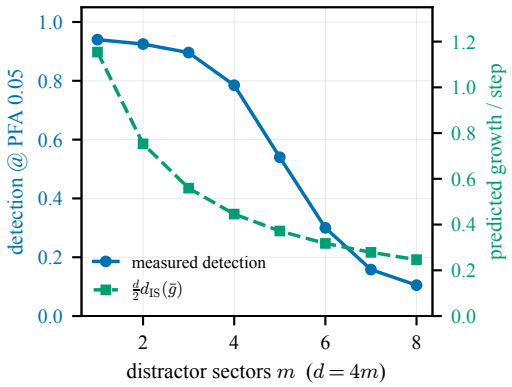


Figure 10: A benchmark whose difficulty is dilution: identical information about the change in every row, with only distractors added. Detection tracks the predicted growth and collapses where the inflation falls below the grid.

object-centric bias helps when the task is to *bind* per-object information, and cannot help when the task is to *select* which object matters.

The consequence is a benchmark-design result: a dilution mechanism is informative about representations only if the affected group is identifiable from the training distribution, or if the detector's alternative class can express a sparse change. We report the pre-registered verdict - no arm beat any other - together with the reason, rather than reframing the setting as one in which the method "was not really tested".