

Steered rare-event estimators can miss failure regions: evidence from trained ReLU networks

Oct 25, 2026 · Matthew Ryan Davis

Summary

Rare-event estimation for neural networks is moving toward white-box methods: importance-sampling proposals steered by a model’s internals, and splitting methods run in activation space. On trained MNIST networks we tested several white-box estimators against subset simulation, a splitting method, and none delivered a robust end-to-end advantage. The sharpest result is a warning for steered proposals. An importance sampler built from the weights calibrated well on its development images (mean/reference 0.99–1.09 over 10 repeats) yet underestimated one later image by 6.5–14.5 orders of magnitude, while an agreement-only check still passed. Trained networks explain why: margin tails are heavy and skewed, and the response to noise collapses onto a few directions by mid-depth, yet failures depend on nonlinear noise effects outside those directions. Subset simulation, which needs no model of where failures lie, was the strongest estimator we tested, though 15 of 40 early panel cases stayed unresolved and its rarest references carried up to 75% relative error. Findings 1–4 concern probability estimation; Findings 5–6 are exploratory applications (backdoor detection and training health) where our label-free measures showed no convincing practical advantage over existing tools. Which choices were registered in advance is stated per experiment.

Approach	Question	Outcome
Gaussian closure (exact pairwise ReLU covariance)	Flip probability from propagated moments	Clean net $\times 23$ low (median); label-noise net off by 10^{17} – 10^{54}
Design-point method (FORM)	Flip probability from the nearest failure point	Signed median near zero, but only 12 of 25 cases within $\times 2$ and 9 of 25 off by more than $\times 10$

Approach	Question	Outcome
Curvature correction (finite-scale SORM)	Fix FORM with boundary curvature	Tested finite-scale quadratic did not validate (training R^2 0.56; held-out RMSE/SD 1.06)
Subspace estimator with smoothing	Exact in 9 key directions, smoothed elsewhere	Matched on a clean image; $\times 8$ –17 low on a label-noise image
Low-dimensional importance sampling (SRE-IS)	Unbiased sampling steered by the weights	Calibrated on development images; on one image of a later validation block 6.5–14.5 orders of magnitude low (relative error and effective sample size flagged non-convergence; an agreement-only gate still passed); slower than subset simulation end to end
White-box subset simulation	Faster MCMC steps using the key directions	Median efficiency $\times 0.48$ (split steps) and $\times 0.34$ (sensitivity-scaled) of the better standard method; one point $\times 1.9$
Backdoor detection by random probing	Flag a backdoor and its target class	Caught 2 of 3 attacks in the last round, but 3 of 5 clean nets flagged; Neural Cleanse caught the same 2 with 1 false alarm
Training-health monitor	Predict overfitting without labels	Rank correlation 0.11 for ours vs 0.78 for input-gradient size; a 0.02-point early-stopping pilot win

Setup

All networks have the contest's shape: 784 MNIST pixels padded to 1,024, 16 bias-free 1,024 $\times$ 1,024 ReLU layers and a linear readout, trained with Adam for 1,404 steps unless stated.

Network	Training	Test accuracy
clean, clean2	Full data, two seeds	96.93%, 96.92%
noise20, noise50	20% / 50% of training labels randomly corrupted	96.74%, 95.51%
data10, data2	10% / 2% of the training images	94.59%, 91.34%
noise100	All labels random	10.10% (chance)

Noise model. Additive Gaussian static on the standardized pixels, σ chosen per image so the design-point estimate targets a failure probability of $1e-3$ to $1e-6$. Perturbations are unclipped and touch only the 784 pixel coordinates (padding unchanged). A failure is any change of the predicted class. Findings 5–6 also use random patches, low-frequency noise and warps.

References. Subset simulation (preconditioned Crank–Nicolson with innovation scale 0.3, so proposal correlation ≈ 0.954 ; 2,000 samples per level, level probability 0.1, 5–10 repeats) and projected cross-entropy importance sampling. A case counted as resolved when the two agreed within $\times 1.5$ and each had under 30% relative error; this agreement gate was set after seeing the first results. 25 of 40 panel cases met it and 15 stayed unresolved, so the resolved set may favour easier cases; "resolved" is an operational gate, not a certificate of truth.

Process. One person (Matthew Ryan Davis) directed the work, Claude designed experiments and read results, and Codex ran every experiment on a laptop CPU. Some later rounds fixed their pass rules and held-out data in a plan file before running; each finding says which choices were registered and which were post hoc.

Related work

Our results bear most on recent white-box rare-event estimators, and on the older evidence that splitting scales.

- **Splitting for network robustness.** [Webb et al. \(ICLR 2019\)](#) estimate the probability that a property is violated under an input distribution with adaptive multilevel splitting, scaling to a CIFAR-100 DenseNet. Our subset simulation is the same family. We add evidence that splitting holds up where steered alternatives fail, not a new splitting method.

- **Rare outputs of language models.** [Wu & Hilton \(2024\)](#) compare importance sampling with activation extrapolation on small transformers and find importance sampling best, both beating naive sampling. [Angell et al. \(2026\)](#) build importance-sampling proposals by activation steering and report needing about 5% of the Monte Carlo samples. [Parulekar & Liu \(2026\)](#) adapt multilevel splitting to activation space with a gradient-based MCMC kernel, and [Dorman et al. \(ICML 2026\)](#) apply statistical-physics rare-event methods to language models.
- **What we add.** Finding 3 is a caution for steered proposals: one derived from a model's internals calibrated on development inputs and still missed a failure region on another input, with relative error and effective sample size as the only warnings. We study classifiers under input noise, not language-model outputs, so how far this transfers is open.
- **Reliability methods.** FORM and SORM are the standard first- and second-order structural-reliability approximations; Finding 1 shows how their smooth-boundary assumptions break on trained piecewise-linear networks. The Gaussian closure is our own contest estimator, described in the WhestBench write-up.

Backdoor detection and generalization measures (Findings 5–6) have their own literatures, cited in those sections.

Finding 1: moment closures and design-point methods fail in the tail

A Gaussian closure that propagates means and covariances is accurate for bulk statistics on trained networks but useless for rare failures. The design-point method (FORM) has a signed median near zero but is off by more than $\times 10$ on 9 of 25 cases (only 12 within $\times 2$), and our tested finite-scale curvature correction did not validate. Errors are $\log_{10}(\text{prediction} \div \text{reference})$ over resolved cases (10 images, targets $1e-4$ and $1e-6$):

Method	clean net (15 cases): median / median abs / range	noise50 net (10 cases): median / range
FORM, union over rival classes	+0.03 / 0.29 / -1.19 to +1.34	+0.02 / -1.09 to +1.11
Gaussian closure	-1.37 ($\times 23$ low) / — / -4.46 to +0.11	-24.5 / -53.5 to -16.7

- **Why the closure fails.** On the label-noise network the margin distributions of failing images are skewed (-1.1 to -2.9) with excess kurtosis up to 10.6, and few last-layer units change activation under the noise (2.5–3.8% of sample–unit pairs at layer 15 on those images). The propagated Gaussian approximation is wrong by tens of orders of magnitude; its propagated variances are also off on some images (image 1228: predicted margin SD 0.65 vs sampled 0.93), so the error is not purely distributional shape.
- **Why FORM misses.** It over-predicts where the failure set curves away from the design point (image 5183, $\times 11$ –13) and under-predicts where failure needs the noise outside the design direction (image 783, $\times 10$ –12).
- **Curvature does not help.** A symmetric quadratic fitted to the margin around each design point at the noise scale gave admissible (positive) Breitung factors for only 124 of 225 points, a median training R^2 of 0.56, and held-out error no better than predicting the mean (RMSE/SD 1.06). This was the first stage only; the Tvedt and multistart variants were not run. Our reading is that on a piecewise-linear network the boundary at noise scale is kink structure, not a smooth surface.
- **Cost.** Even for bulk statistics the exact closure took 5.2 s per image at σ 0.5, against 0.49 s for 2,048 Monte Carlo samples. A rank-5 operator version switching at layer 6 stayed within a median 0.28% of the full closure's variance (60 clean images) and cut counted matrix work by a median 34.5% (range 30–58%; the counter excludes factorizations and elementwise work), but did not cut wall time.

Finding 2: failures depend on nonlinear noise effects outside the tested subspaces

Trained networks compress their response to noise onto about five directions by mid-depth, yet failures still depend nonlinearly on noise outside the low-dimensional subspaces we tested. That combination defeated every method we tried that is exact in a few directions and approximate elsewhere; it does not rule out other adaptive or nonlinear representations.

The collapse is real. On the clean net at σ 0.5 (10 images; zero-based layers), the top-5 directions of the pre-activation covariance hold a median 0.46 of the variance at layer 4, 0.79 at layer 6, 0.94 at layer 8 and ≥ 0.98 from layer 10; an untrained net rises only from 0.09 at layer 4 to 0.22 at layer 15. The full anatomy panel covered 124 cases across nets, images and σ . Projections onto those directions keep the heavy margin tails (skew to -2.9 , kurtosis to 10.6 on noise50 outliers).

But a 9-direction estimator misses the failures. We built an estimator that is exact in a 9-direction input subspace (design-point directions plus closure directions pulled back through the Jacobian) and linearizes the margins in the remaining noise. It matched the reference on clean image 295 ($1.12\text{e-}4$; all variants within $\times 0.79\text{--}1.14$, the pullback-only variant lowest) and was $\times 8\text{--}17$ low on noise50 image 783 (reference $2.90\text{e-}3$) in its design-point and combined configurations, the same miss as FORM; the pullback-only and no-smoothing variants were worse ($\times 0.003\text{--}0.004$ and $\times 0.04$).

The miss is not a missing failure region. Clustering subset-simulation failure samples found no clearly separated clusters (silhouette $0.04\text{--}0.12$), and boundary searches from cluster centres mostly did not converge (0, 0 and 3 of 6 per case), so this is diagnostic evidence, not proof that no other region exists. What we did find:

Image	Failures that still fail with the outside noise removed	Failures whose linearized winning-rival margin stays negative at full noise	Outside-noise SD, sampled vs linearized	Skew of outside-noise effect
noise50 783	2.6%	3–4%	0.71–0.83 vs 0.11–0.13	–1.4 to –1.5
clean 5321	7–8%	11–12%	1.70–1.99 vs 1.43–1.61	–0.4 to –0.5
noise50 5183	65–74%	70–76%	—	—

The fractions are conditional on dependent subset-simulation failure samples, and the SD and skew come from 32 fresh outside-noise draws per failure anchor, so the skew estimates are noisy. On image 783 the remaining noise (norm about 28σ) widens the margin's spread $6\text{--}7\times$ beyond the linear estimate and skews it toward failure. With 16 outside-noise draws per sample, a sampled-moments Gaussian correction moved the estimate from $\times 0.09$ to $\times 0.38$ of the reference; with 64 draws the Gaussian correction gave $\times 0.52$ and counting failures directly gave $\times 0.88$ (the reported error excluded residual-draw uncertainty). Closure-based conditional moments missed conditional tails at the anchors we checked (one example: 0.086 vs 0.178 sampled) and a quadratic surrogate over the subspace failed outright.

Finding 3: steered low-dimensional importance sampling underestimates severely on a later validation block

Importance sampling over the 9-direction subspace, with fresh outside-noise draws for every sample, is unbiased for a frozen proposal and a fixed sample size (adaptive fitting kept separate; precision-based stopping can add bias), and it calibrated well on the images it was developed on. On one image of a later validation block it missed the failure region by 6.5–14.5 orders of magnitude. Its relative error and effective sample size both flagged non-convergence, but the error bars did not quantify the missing mass, and an agreement-only gate still passed.

Development images (pass). With 16 outside-noise draws per sample and a 10% plain-Gaussian safety share, 10 independent repeats per case gave:

Case	Mean / reference	Repeat SD / reported SE
clean 295	1.09	0.68
noise50 783	0.99	1.01
clean 5321	0.99	1.00
noise50 5183	1.00	1.18

On the 25 resolved panel cases (selected from the resolved set; one replication each, four for five cases), 21 means were within $\times 1.25$ of the reference and 24 within $\times 2$. In a rarity sweep down to $p \approx 1e-8$, 17 of 18 new subset-simulation references agreed within two combined standard errors.

Validation block (fail). On three images per network held out from the directional-sweep selection (all six image-network pairs had appeared in earlier pilots, and noise50 5183 was one of the calibration images, so these are not previously unseen images), the pre-registered agreement gate formally passed (21 of 24 points), but one image was missed almost entirely:

noise50 image 995	$\sim 1e-3$	$\sim 1e-4$	$\sim 1e-5$	$\sim 1e-6$
Importance sampling	$1.3e-4$	$4e-12$	$3e-17$	$5e-22$
Subset-simulation reference	$2.5e-4 \pm 3e-5$	$1.2e-5 \pm 3e-6$	$1.5e-6 \pm 7e-7$	$1.7e-7 \pm 1.3e-7$
Effective sample size	8.2	1.6	1.8	1.2

The rarest point counted as agreeing only because the reference itself had 75% error, so an agreement-only gate is weak. The weights-derived proposal never reached the failure region. Relative errors of 0.35–0.92 and all four runs hitting the 65,536-sample cap without meeting the 20% precision stop signalled trouble; the effective sample size, reconstructed from the same mean and error, says the same thing rather than adding independent evidence.

Cost. Counting every network evaluation, a cold-start estimate took 355–403 s per point (330–374 s of it the boundary search), 6–55× slower than subset simulation (median 30×) on the 18 points where both reached 20% error. A cheap, unconverged 3-rival search cut setup to 7–8.5 s and kept 18 of 20 points within two combined standard errors of the reference (4,096-sample floor). With a 1,024-sample floor (17 of 20), sharing one setup across four noise levels beat subset simulation on five development pairs (40–77 s against 65–263 s; noise50 5321 excluded, and some subset curves missed their precision stop). On the six validation pairs it was slower on every pair (102–1,208 s against 81–110 s). These timings ran under CPU contention and compare fixed-repeat references with variable-precision runs, so they are not matched-precision speed ratios. A variant that reads the failure probability for every noise level off one set of rays was 7.5–15% accurate on two pilot curves but failed a post-hoc replacement gate (16 of 20 points; error bars matching the repeat spread at 4 of 8, or 3 of 8 counting fresh fits only).

Finding 4: subset simulation is robust, and white-box tuning does not speed it up

Using the white-box subspace to tune subset simulation's MCMC steps made it less efficient at five of six development points. Efficiency is cold work-normalized variance, forward passes including setup × relative variance, from 20 independent repeats; the table gives the better standard method's value divided by each white-box variant's (above 1 favours white-box):

Point	Split step sizes (S and complement)	Plus sensitivity-scaled steps
clean 295, $\sim 1e-6$	0.28	0.02
clean 295, $\sim 1e-4$	0.24	0.03
noise50 995, $\sim 1e-6$	0.48	0.20
noise50 995, $\sim 1e-4$	0.48	0.51
noise50 7076, $\sim 1e-6$	1.91	1.38
noise50 7076, $\sim 1e-4$	0.94	0.48

- **Baselines.** Standard preconditioned Crank–Nicolson (innovation scale 0.3) and our own between-level step-size adaptation targeting 44% acceptance, in the spirit of adaptive conditional sampling (Papaioannou et al. 2015) but not an exact reproduction of it. Neither was uniformly better (ratio 0.72–3.52).
- **Accuracy.** Within two standard errors of a high-effort reference (adaptive, 10,000 samples per level, 10 repeats) at 5–6 of 6 points for every variant. Even subset simulation spread $\times 0.3$ –3.8 at $\sim 1e-6$ on the label-noise images.
- **Why no gain.** Separate updates of the subspace and its complement cost two forward passes per step, and the one point that gained ($\times 1.9$) was also where the white-box variant was closer to the reference. Even an optimistic halving of the sampling cost (setup cannot be halved) would leave the medians (now $\times 0.48$ and $\times 0.34$) near or below $\times 1$, short of the pre-set $\times 2$ target, so validation on new images was not run.

Finding 5: backdoor detection by rare-output probing shows no advantage over Neural Cleanse

Counting how often random perturbations of clean images flip predictions to each class detects simple backdoors, but clean networks produce false alarms. Neural Cleanse matched both single-target detections in each later round with fewer clean flags; our one extra hit (the confirmation round’s source-specific 4→9) came with five of five clean false alarms. This and Finding 6 are exploratory applications; their false alarms are not evidence about probability estimation. We ran three rounds on contest-shaped MNIST networks (1% poisoning unless noted; trigger success above 90% for every attack counted):

Round	Attacks (target)	Ours: caught / clean flagged	Neural Cleanse (our NumPy adaptation): caught / clean flagged	HTell-style head probing (our adaptation): caught / clean flagged
Pilot	4×4 patch (7), blend α 0.1 (3)	Registered mean score: patch never flagged, no common setting passed. A median rule chosen afterwards on this pilot caught both, at separate settings / 0 of 3	Both / 1 of 3	Per-unit-range variants: both / thresholds set on these clean nets
Confirmation (registered)	Faint blend α 0.05, 5% poison (8); low-frequency pattern (5); source-specific patch 4→9. Warp and all-to-all attacks failed to train	3 of 3 (primary or secondary rule) / 5 of 5	Blend and pattern targets found, each with extra classes flagged; source-specific unflagged / 1 of 5	Per-unit-range variants: blend and pattern caught, source-specific missed / 1 of 8 (pilot + confirmation clean nets, used for calibration)
Head-adaptive (registered)	Head-regularized patch (2); frozen-head patch (6); frozen-head source-specific 7→1	2 of 3 / 3 of 5 (primary rule alone: 2 of 3 / 1 of 5)	2 of 3 correct; source-specific net flagged with the wrong class (3) / 1 of 5	Last-block range variant: head-regularized only / 1 of 5 (frozen 8-clean thresholds)

- **What breaks our detector.** Each clean network has its own favourite flip classes (one leaned to class 2, another to 3, 6 and 8), driven by a few fragile images near a boundary; a per-source-class rule fires on ordinary confusions such as 9→4. Smooth low-frequency noise produced false alarms on all 8 clean networks of the first two rounds.
- **What works.** At light pixel noise (σ 0.5) and random 4×4 patches, the median per-image flip rate gave index 0 on every clean network and 8–17 on single-target backdoors. That exploratory rule, written from the pilot alone, caught both single-target attacks of the confirmation round with no false alarms: an out-of-sample check of a superseded rule, not a registered pass, and not to be pooled with the registered four-setting detector.
- **Where head probing is blind.** With the same probe distribution and random draws, networks sharing a frozen clean output layer get identical scores from head-only probing (range-derived probes can still differ with the backbone), so it cannot separate frozen-head backdoors from frozen-head clean controls. Our whole-network probing caught the frozen-head patch; so did Neural Cleanse.
- **Cost.** Our fixed-budget screen averaged 74, 34 and 32 min per network in the three rounds; our Neural Cleanse adaptation (NumPy gradients, 20 optimization images, mask-L1 anomaly rule, masks checked on the optimization images only) 6.7–7.7 min; head probing under 0.25 s of probe time, excluding loading and range estimation. Subsampling the saved counts (not independent runs) put the first budget with 95% correct-target detection at 500–1,000 draws per image, though the pilot patch held it stably only from 10,000, and clean false-flag rates stayed high at every budget.

Prior art: random latent probing of the classification head (HTell, arXiv 2605.18908) reports 99.03% detection at 2.11% false alarms over 21 attacks on its own benchmark, so the idea of a class that stands out under random probing is not new. Our adaptation used 10,000 probes (our choice) and thresholds from our own clean nets, not the paper's 570-model calibration.

Finding 6: a label-free training-health monitor loses to input-gradient size

On a held-out seed, input-gradient size ranked networks by overfitting far better than any of our measurements. We trained 14 contest-shaped networks (label noise 0/20/40%, training data 100/10/2%, input-noise augmentation σ 0/0.25/0.5; two seeds; 4,212 steps; 10 checkpoints each), fixed every measure's sign and variant on seed 1, and scored the final

checkpoints of seed 2. The gap is training accuracy on the labels actually used (corrupted for label-noise runs) minus clean test accuracy. Granulated τ averages three factor-wise triples that share the baseline run, not a full crossed design:

Measure	Rank agreement with the generalization gap (granulated Kendall τ , seed 2)
Input-gradient size (baseline)	0.78
Our collapse profile	0.11
Our flip-rate curve	0.11
Our sink-class concentration	-0.11
Our random-patch flip rate	-0.56

Other baselines were our own WeightWatcher-style power-law tail fit to the weight spectra (continuous MLE with a KS-chosen cut-off, not the official package) and the RankMe effective rank of the uncentered last-hidden-layer activations. For early stopping, the registered criterion technically passed: our flip rate at σ 0.5 lost a median 0.06 percentage points of test accuracy against 0.08 for the best baseline, on 7 runs that barely overfit. The median hides 1.47 points of regret on one augmentation run and 0.73 on the 40% label-noise run, so we read it as a tiny pilot win, not a practical advantage. The study took 56.7M forward passes for the label-free measurements (training excluded) and 10.5 hours of summed run time.

Limitations and open directions

The conclusions hold for one dataset, one architecture, and Gaussian pixel noise for Findings 1–4 (Findings 5–6 add random patches, low-frequency noise and warps); each could change them.

- **MNIST.** Accuracies sit near 95–97% and most runs barely overfit, which weakens the early-stopping test and may favour simple measures such as input-gradient size. Harder tasks were not tested.
- **Architecture.** All networks are deep, wide, bias-free MLPs of the contest's shape. Convolutional and residual networks may not collapse in the same way.

- **Post-hoc choices.** The $\times 1.5$ reference agreement gate, the 50-sample effective-size bar, the directional-curve replacement gate and the scoring rules of the backdoor pilot were set after seeing data; other rounds were registered in advance, and the text marks which is which.
- **Small samples.** Most comparisons rest on 2–10 images per network and 3–5 clean networks per round.

Open directions.

1. Repeat the importance-sampling versus subset-simulation comparison on a CIFAR-10 convolutional network with 20 or more images per network, adding adaptive multilevel splitting (Webb et al.) and activation-steered importance sampling (Angell et al.) as baselines.
2. Test the label-free measures on the NeurIPS 2020 Predicting Generalization in Deep Learning benchmark, which supplies trained convolutional networks with known generalization gaps and published baselines. A noise-resilience entry already exists there, so only the collapse profile and sink-class measures would be new.
3. Require both relative-error and effective-sample-size convergence, plus coverage and mixing checks, before trusting any steered rare-event sampler; a high effective sample size cannot certify that no failure mass was missed.
4. Probe whole networks, not just the head, when an attacker may freeze or regularize the output layer, and calibrate per-network sink classes before reading flip rates as evidence.

Sources

- [WhestBench Phase 2 write-up](#): the contest entry, the closure and its learned corrections.
- [Webb, Rainforth, Teh & Kumar, A Statistical Approach to Assessing Neural Network Robustness \(ICLR 2019\)](#); [Wu & Hilton, Estimating the Probabilities of Rare Outputs in Language Models \(arXiv 2410.13211\)](#); [Angell et al., Estimating Tail Risks in Language Model Output Distributions \(arXiv 2604.22167\)](#); [Parulekar & Liu, Estimating Rare Events in Language Models with Proper Evaluation \(arXiv 2607.18454\)](#); [Dorman et al., Rare Event Analysis of Large Language Models \(ICML 2026\)](#). Abstracts and summaries read; full texts not reviewed.
- [HTell: Fast and Lightweight Backdoor Detection via Head Random Probing \(arXiv 2605.18908\)](#).

- [NeurIPS 2020 Competition: Predicting Generalization in Deep Learning \(arXiv 2012.07976\)](#).
- [Using noise resilience for ranking generalization of deep neural networks \(arXiv 2012.08854\)](#) (seen in search results only).
- [Wang et al., Neural Cleanse \(IEEE S&P 2019\)](#); [Papaioannou et al., MCMC algorithms for Subset Simulation \(2015\)](#); [Martin & Mahoney, implicit self-regularization \(JMLR 2021\)](#); [Garrido et al., RankMe \(ICML 2023\)](#). Codex verified these links; they identify the methods, not the equivalence of our adaptations. Jiang et al. 2020, Fantastic Generalization Measures, is cited from memory.
- Raw results: the outputs/rare folder of the Codex workspace (not published). A read-only numeric audit of 28 result files, with source hashes, is in `outputs/writeup/review_20261025/paper_numeric_audit.json`.