

German Alfaro - independent researcher

Abstract |

Can we predict a neural network’s expected behavior by analyzing its structure rather than running it on many inputs? We address this question in the **ARC White-Box Estimation Challenge 2026**: given white-box access to random 256×32 ReLU MLPs and a fixed FLOP budget, estimate the expected final-layer activations under Gaussian inputs. Starting from plain Monte Carlo (MC), we developed a production estimator combining **input whitening**, **antithetic sampling**, and **algebraic FLOP optimizations** (folded whitening + half-covariance), achieving an official adjusted score of 3.357×10^{-7} — **1.9× better** than the MC baseline (6.5×10^{-7}) on the public split. Along the way we ran **13 diagnostic scripts** testing analytical closures (covariance propagation, K=3 cumulants, Gram-Charlier), control variates, active-subspace sampling, low-rank compression, layer skipping, and adaptive importance sampling. Our main scientific contribution is not only the estimator but a **decisive map of what fails and why** at depth 32: analytical methods hit an irreducible Gaussian-closure bias; higher-order Hermite/Gram-Charlier corrections have a 10× theoretical ceiling that collapses once moments are estimated from finite samples; and generic unbiased variance reduction (whitening + antithetic) appears near its floor. These results clarify when white-box structural analysis can compete with sampling — and when it cannot — with direct implications for compute-bounded model auditing and monitoring. |

1. Introduction

Modern AI safety workflows increasingly require **predicting model behavior under uncertainty** — e.g., estimating expected activations, tail probabilities, or downstream effects — without unlimited forward passes. White-box access to weights should, in principle, allow **mechanistic estimation**: propagating statistical moments through the architecture instead of brute-force sampling. The **ARC White-Box Estimation Challenge 2026** formalizes this: given one random MLP (width 256, depth 32, He initialization) and a budget of ~272B FLOPs, predict expected neuron activations. The grader scores **budget-adjusted final-layer MSE**:

$$\text{adjusted_score} = \text{final_layer_mse} \times \max(0.1, \text{effective_compute} / \text{flop_budget})$$

Lower is better. The organizer MC baseline scores $\sim 6.5 \times 10^{-7}$; the public leaderboard leader is $\sim 2.6 \times 10^{-7}$.

Why this matters for AI safety. Auditing and monitoring large models often requires estimating internal statistics (mean activations, rare-event probabilities) under input distributions. If structural methods could beat sampling at fixed compute, safety evaluations could be cheaper and more scalable. If they cannot — especially at depth — that boundary is itself valuable: it tells practitioners when exhaustive or MC-based evaluation remains unavoidable.

¹ Submitted as part of the **ARC White-Box Estimation Challenge 2026**, conducted at the [Global South AI Safety Hackathon](#), June 2026

Our main contributions are:

1. **A production-grade whitened antithetic MC estimator** with folded whitening and half-covariance tricks, graded at 3.357×10^{-7} (submission 311690), reclaiming $\sim 4.4\%$ more samples at zero bias.
2. **A systematic diagnostic benchmark** across 13 scripts, mapping which structural alternatives (covariance propagation, $K=3$ cumulants, Gram-Charlier, control variates, active subspaces, compression, layer skipping) remain viable at depth 32 — and where the compute–accuracy frontier sits for each.
3. **A variance-reduction ladder and score-structure analysis** establishing that whitening accounts for $\sim 2\times$ of our gain, antithetic pairs add modestly post-ReLU, and we appear near the generic unbiased variance-reduction ceiling — explaining the remaining gap to the leaderboard leader ($\sim 31\%$ lower per-sample variance).

2. Related Work

Monte Carlo sampling is the unbiased ground truth: error scales as $1/\sqrt{k}$. Variance reduction (antithetic variates, importance sampling, control variates) is standard but often breaks under deep nonlinear composition.

Mean propagation (diagonal Gaussian closure) tracks per-neuron means and variances through ReLU layers in $O(\text{depth} \cdot \text{width}^2)$ FLOPs. It works at shallow depth but ignores correlations; at depth 32 it is $\sim 155\times$ worse than MC locally.

Covariance propagation (full matrix, “gain method”) captures correlations at $O(\text{depth} \cdot \text{width}^3)$ cost ($\sim 1.6\text{B}$ FLOPs). Example implementations ship with the official challenge starter kit. It still assumes Gaussian closure at each ReLU, which accumulates bias: final-layer MSE $\sim 9.5 \times 10^{-5}$ vs MC $\sim 2.9 \times 10^{-6}$ ($\sim 33\times$ worse).

Cumulant propagation ($K=3$) from Schöpfer et al. (ARC, 2025) uses Hermite polynomials and diagram summation to propagate skewness through ReLU networks. The official `mlp_cumulant_propagation` code beats MC at depth 4 but loses $\sim 4\times$ at depth 32 — depth, not implementation quality, is the bottleneck.

Competing with sampling (ARC blog, 2025) frames beating MC as a milestone for mechanistic estimation. Our work differs by (a) pushing optimized MC with whitening rather than replacing it, and (b) measuring **ceilings** (what is theoretically possible with exact moments) separately from **estimators** (what works with $\sim 10^4$ samples), closing several apparent gaps.

When to use our method vs. prior art: Use our whitened antithetic MC when you need unbiased, compute-aware estimation of deep random MLPs at the challenge scale (256×32). Use analytical propagation only at shallow depth (≤ 4 layers) or when bias is acceptable. Our diagnostics show no current analytical method beats optimized MC at depth 32 with $\sim 10^4$ samples.

New insight: The failure mode is not “bad moment propagation” alone but **Gaussian closure at ReLU** combined with **high-variance estimation of higher-order moments**. Gram-Charlier with *true* skew/kurtosis would be $10\times$ better than MC; with *sample* moments it ties MC exactly at $N \approx 9800$.

3. Methods

3.1 Problem and scoring

- **Input:** MLP with weights $W_0 \dots W_{L-1}$, each 256×256 ; inputs $x \sim N(0, I)$.
- **Target:** $E[\text{ReLU}(z_L)]$ where z_L is the final pre-activation vector.
- **Output:** (depth, width) array; grader uses **only the final row**.
- **Budget:** $\sim 272 \times 10^9$ FLOPs tracked by `flopscope`; effective compute includes Python/residual overhead R_m .

3.2 Production estimator (final pipeline)

Our best graded estimator (`estimator.py`) implements:

1. **Antithetic Gaussian sampling:** draw $u \sim N(0, I)$ for $k/2$ rows, stack $[u; -u]$. Exact zero batch mean; halves covariance computation cost.
2. **Input whitening (layer 0 only):** compute empirical cov $\Sigma = u^T u / (k/2)$, eigendecompose, build $\Sigma^{-1/2}$, transform inputs before the first matmul. Matches batch to mean 0, cov I — aligning samples with the target distribution $N(0, I)$.
3. **Fold whitening into W_0 :** precompute $W_0' = \Sigma^{-1/2} W_0$ so $\text{ReLU}(x_{\text{white}} @ W_0) = \text{ReLU}(x @ W_0')$, eliminating one k -dependent matmul ($\sim 1.23B$ FLOPs saved).
4. **Half-covariance:** with antithetic pairs, $x^T x = 2 \cdot u^T u$; use only the first half for Σ .
5. **Final-layer-only output:** return zeros for layers 0–30, estimate only layer 31. Saves negligible FLOPs but matches leaderboard leaders and simplifies the path.
6. **Budget fraction:** use $\sim 15.5\%$ of FLOP budget for forward passes (`_BUDGET_FRACTION = 0.155`), yielding $k \approx 9826$ samples at width 256, depth 32.

Key design choices: - Whitening only at **layer 0** because the target distribution is known; internal activations are non-Gaussian and post-ReLU whitening would change the computed function (biased). - `flopscope.numpy` only (no raw NumPy) for grader compatibility. - Float32 RNG draws to reduce residual overhead. - 100MB array cap on (k, width) to avoid grader memory limits.

3.3 Diagnostic methodology

We built 13 scripts under `scripts/` to test hypotheses before grader submission:

Script	Tests
<code>diag_finite_width.py</code>	Analytical ceilings with <i>true</i> final-layer moments
<code>diag_gc_estimator.py</code>	Gram-Charlier with <i>estimated</i> moments at real budget
<code>diag_variance_reduction.py</code>	Variance ladder: plain $\rightarrow$ anti $\rightarrow$ whiten $\rightarrow$ orth
<code>diag_mc_minus_covprop.py</code>	CovProp bias stability, control-variate ρ^2
<code>diag_active_subspace_*.py</code>	Low-rank input subspace sampling (pilot +

Script	Tests
diag_matrix_compression.py	oracle)
diag_layer_skip.py	SVD weight truncation for FLOP savings
diag_variance_ab.py	MLMC / layer skipping
	Shrinkage whitening α sweep; GC control variate
bench_frac_sweep.py, bench_pipeline_audit.py	Budget tuning and pipeline correctness
profile_estimator.py	FLOP and wall-time profiling

Each diagnostic uses paired seeds, fixed k , and high- N MC truth (80k–1.5M samples) where applicable. Full log: `EXPERIMENTS.md`.

3.4 What we tried that did not work

Approach	Outcome
Covariance / mean propagation	4–33 $\times$ worse; closure bias
$K=3$ cumulant extrap (final layer only)	$\beta > 0$ always worse locally
Full $K=3$ cumulant propagation (vendor code)	$\sim 4\times$ worse than MC @ depth 32
Shrinkage blend MC + cov-prop	Graded 5.008×10^{-7} vs 4.803×10^{-7} whitening
Control variates (linear, cov-prop, GC)	$\rho^2 \approx 0$ or biased control
Active subspace sampling (oracle U)	3.6 $\times$ worse than production
Weight SVD compression (rank 128)	$\sim 171,000\times$ MSE inflation
Layer skip / MLMC	$\text{corr} \leq 0.13$ skipping 1 layer; $\text{Var}(f-g)/\text{Var}(f) \geq 1.35$
Adaptive IS, radial stratified, internal CV	ESS collapse or insufficient p
Orthogonal Gaussian blocks post-whiten	0 $\times$ extra reduction
LHS / Sobol QMC	Worse on grader; whitening destroys QMC structure
frac=0.5 (more samples)	Graded 3.611×10^{-7} , worse than 3.357×10^{-7}

4. Results

4.1 Official grader results (public split, 50 MLPs)

Table 1. Graded submissions (selected).

Rank	Method	Adjusted score	vs MC baseline	Util
★ Ours	Fold whiten + half-cov, anti, frac=0.155, zeros L0–30	3.357×10^{-7}	1.9$\times$ better	$\sim 15.5\%$

Rank	Method	Adjusted score	vs MC baseline	Util
	White + anti, frac=0.155, zeros L0–30	3.494×10^{-7}	1.9× better	16.2%
	White + anti, frac=0.155	3.520×10^{-7}	1.8× better	~15.5%
	Whitening, frac=0.092	4.803×10^{-7}	1.4× better	~9.2%
Ref	MC baseline (organizer)	6.5×10^{-7}	1.0×	—
Ref	Leaderboard top	$\sim 2.6 \times 10^{-7}$	$\sim 2.5 \times$ better	$\sim 14.5\%$

Submission **311690** (fold whiten + half-cov) improved **−3.9%** vs zeros-only and **−4.6%** vs the prior frac=0.155 baseline, with k increasing 9409 → 9826 (+4.4% samples) at zero measurable bias ($<10^{-6}$ vs unfused path).

4.2 Variance-reduction ladder (local, fixed k=1024)

Table 2. Per-sample MSE at fixed k (3 seeds, 160 batches).

Sampler	MSE	Reduction vs plain
Plain iid	7.69×10^{-5}	1.00×
Antithetic	6.36×10^{-5}	1.21×
Whiten	3.73×10^{-5}	2.06×
Whiten + anti (production)	3.41×10^{-5}	2.26×
Orth + whiten + anti	3.46×10^{-5}	2.23× (no gain)

Observation: Whitening drives most of the gain. Antithetic pairs help less after ReLU breaks input symmetry. Orthogonal Gaussian sampling adds nothing once whitening covers the dominant subspace.

4.3 Analytical ceiling vs. reachable estimation

Table 3. Final-layer MSE with exact vs. estimated moments (6 seeds, N_truth=1.5M for ceilings).

Method	MSE	vs best graded
Infinite-width constant $\sqrt{(1/\pi)}$	1.00	$3 \times 10^6 \times$ worse
Gaussian closure (true μ, σ)	1.50×10^{-6}	4.5× worse

Method	MSE	vs best graded
Gram-Charlier (true μ , σ , skew, kurt)	3.31×10^{-8}	0.10× (10× better ceiling)
Direct MC @ $N \approx 9800$	6.85×10^{-6}	1.0×
Gram-Charlier @ $N \approx 9800$ (sample moments)	6.88×10^{-6}	1.005× (no gain)

Interpretation: Non-Gaussianity at the final layer (mean $|\text{skew}| \approx 0.41$, $|\text{excess kurtosis}| \approx 0.39$) creates headroom for higher-order closures — but estimating skew/kurtosis from the same $\sim 10^4$ samples injects variance that exactly cancels the benefit. The crossover occurs near our operating budget.

4.4 Active subspace and compression (local)

- **Oracle active input subspace** (true Jacobian directions, rank 4): **3.6× worse** than production whitening — even perfect subspace knowledge does not beat global 256D whitening.
- **Weight effective rank:** $\sim 155/256$ per layer; rank-128 SVD truncation yields **$\sim 171,000\times$** MSE inflation.
- **Layer skip:** skipping even **one** hidden layer drops coarse/full correlation to **0.13**; MLMC variance ratio ≥ 1.35 at every level.

4.5 Robustness

- **Pipeline audit** at $k=9826$: fold-whiten, half-cov, full-cov, and no-fold paths produce **identical** per-sample variance — no implementation bug.
- **Fine frac sweep** (0.140–0.220): local optimum near $\text{frac}=0.150$ (-1.4% vs 0.155); $\text{frac}=0.5$ graded worse (3.611×10^{-7}) confirming compute-multiplier tradeoff.
- **20-seed paired benchmarks** consistently rank methods in the same order; grader public split (50 MLPs) occasionally diverges from local mini-splits (documented in EXPERIMENTS.md).

Figure 1 (conceptual). Estimator pipeline:

```

u ~ N(0, I), k/2 rows
→ x = [u; -u]                (antithetic)
→ Σ = uTu / (k/2)           (half-cov)
→ W0' = Σ-1/2 W0          (fold whiten)
→ ReLU(x @ W0') → ... → ReLU → zL
→ mean(ReLU(zL))           (final layer only)

```

5. Discussion and Limitations

Implications for AI safety

Our results sharpen the **compute-bounded auditing** question: white-box weight access does not automatically yield better estimates than careful sampling at depth 32. For safety applications that require expected activation statistics under known input distributions:

- **Optimized MC with variance reduction** (whitening, antithetic pairs, FLOP-aware implementation) is the reliable workhorse when unbiased estimates matter.
- **Analytical / cumulant / closure methods** can win at shallow depth but carry **systematic bias** that grows with depth — dangerous if mistaken for ground truth in monitoring pipelines.
- The **$\sim 1.4\times$ gap to the public leader** appears driven by lower per-sample variance ($\sim 31\%$), not architectural tricks (zeros L0–30, library choice, or matrix compression). Closing it likely requires a novel variance-reduction idea not captured by generic methods we tested.

This maps directly to safety evaluation design: before deploying a structural estimator for internal statistics, verify at the target depth and sample budget whether closure bias dominates — our diagnostics provide a template for that verification.

Limitations

- **Scope:** Random MLPs with He init, ReLU, fixed 256×32 shape, Gaussian inputs. Results may not transfer to trained networks, other activations, or structured architectures (attention, sparsity).
- **Leader gap:** We did not fully explain the leaderboard leader’s $\sim 31\%$ variance advantage; shrinkage whitening ($\alpha=0.01$) gives only $\sim 3.5\%$ locally.
- **Single metric:** Grader uses adjusted final-layer MSE; intermediate layers and tail probabilities (where cumulant methods shine in the ARC paper) are not scored here.
- **Compute model:** Scoring depends on flopscope FLOP accounting and residual overhead; micro-optimizations (float32 RNG, zeros L0–30) have small but nonzero effects.
- **Hackathon timeframe:** We did not implement full $K=3$ diagram summation from scratch; we benchmarked the official vendor implementation instead.

Future Work

1. Grader A/B on shrinkage whitening ($\alpha=0.01$) and fine frac=0.150 — marginal but low-risk.
2. Investigate the leader’s sampling scheme via public submission forensics (all-layers MSE signature, utilization patterns).
3. Extend ceiling/estimator separation framework to **tail probabilities** (threshold exceedance), where cumulant methods may retain advantage.
4. Test whether findings hold on AlgZoo / smaller networks where analytical methods first beat MC.
5. Package diagnostics as a reusable “can structural methods beat MC?” toolkit for safety researchers.

6. Conclusion

We entered the **ARC White-Box Estimation Challenge 2026** asking whether mechanistic, white-box methods can beat Monte Carlo for predicting deep MLP behavior under a fixed FLOP budget. Our best submission — **whitened antithetic MC with folded whitening and half-covariance** — achieves 3.357×10^{-7} , nearly $2\times$ **better** than the organizer MC baseline, by

lowering per-sample variance without introducing closure bias. Equally important, we systematically tested and ruled out a wide class of seemingly natural improvements: covariance and cumulant propagation, Gram-Charlier/Hermite closures, control variates, active-subspace sampling, weight compression, and layer skipping all fail at depth 32 for identifiable reasons. The central takeaway for AI safety is dual: **(1)** invest in unbiased, variance-reduced sampling when depth and nonlinearity break Gaussian structure; **(2)** treat analytical propagation as bias-prone unless validated at the target depth and budget — the 10× Gram-Charlier ceiling with exact moments is a cautionary example of optimism that finite-sample moment noise destroys.

Code and Data

- **Code repository:** <https://github.com/galfaroi/Can-You-Predict-a-Network-Without-Running-It->; production estimator in `estimator.py`, experiment log in `EXPERIMENTS.md`, diagnostics in `scripts/`.
- **Challenge:** [ARC White-Box Estimation Challenge 2026](#) — starter kit and grader on AICrowd
- **Vendor benchmark:** `_vendor/mlp_cumulant_propagation/` (ARC official K=3 code)
- **Data:** Challenge MLPs generated deterministically from seeds; no external datasets.

Author Contributions (optional)

[Fill in: e.g., “A.B. led estimator design and grader submissions. C.D. implemented diagnostics. All authors contributed to experiment design and writing.”]

References

1. Schöpfer et al. (2025). *Estimating the expected output of wide random MLPs with cumulant propagation*. arXiv:2605.05179. <https://arxiv.org/pdf/2605.05179>
2. Alignment Research Center (2025). *Competing with sampling*. <https://www.alignment.org/blog/competing-with-sampling/>
3. Alignment Research Center (2025). *AlgZoo: uninterpreted models with fewer than 1,500 parameters*. <https://www.alignment.org/blog/algzoo-uninterpreted-models-with-fewer-than-1-500-parameters/>
4. ARC / AICrowd. *ARC White-Box Estimation Challenge 2026 — starter kit and documentation*. <https://github.com/AICrowd/whest-starterkit>
5. ARC (2022). *Formalizing the Presumption of Independence* (Appendix D — ReLU cumulants). arXiv:2211.06738. <https://arxiv.org/abs/2211.06738>
6. `mlp_cumulant_propagation` (official implementation). https://github.com/alignment-research-center/mlp_cumulant_propagation

Appendix

A. Diagnostic scripts inventory

`diag_finite_width.py`, `diag_gc_estimator.py`, `diag_variance_reduction.py`,
`diag_mc_minus_covprop.py`, `diag_active_subspace_stability.py`,
`diag_active_subspace_oracle.py`, `diag_matrix_compression.py`,

diag_layer_skip.py, diag_variance_ab.py, bench_frac_sweep.py,
bench_pipeline_audit.py, profile_estimator.py

B. Score formula reminder

$$\text{adjusted_final_layer_score} = \text{final_layer_mse} \times \max(0.1, \text{effective_compute} / \text{flop_budget})$$

Below 10% budget utilization the multiplier floors at 0.10; above 10%, extra compute increases the penalty roughly proportionally while MC variance scales as $1/k$.

C. Production hyperparameters

`_BUDGET_FRACTION = 0.155`, `_ANTITHETIC = True`, $k \approx 9826$, whitening at layer 0 only, zeros for layers 0–30.

LLM Usage Statement

We used Claude (Cursor AI assistant) to implement diagnostics, run benchmarks, draft sections of this report, and iterate on `estimator.py`. All grader scores, numerical results, and promotion/rejection decisions were verified against local scripts and `EXPERIMENTS.md`. The team reviewed and owns the final claims.