THE STATISTICAL RESOLUTION OF NODE-CLASSIFICATION BENCHMARKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Prediction errors of a message-passing network are correlated along the edges of the graph it is evaluated on, so the m nodes of a node-classification test set do not contribute m independent observations. We study the consequence: the *statistical resolution* of a benchmark, the smallest difference it can detect, is far coarser than its test-set size suggests, and most published differences fall below it. We formalize resolution through the design effect D and the minimum resolvable gap, show that the classical plug-in estimator of D is unidentified on real error fields, where a previously undescribed degree-sampling bias alone flips the sign of its correlation estimate on 14 of 16 configurations on ogbn-arxiv. We instead estimate D with a graph-partition block bootstrap validated against known design effects, then train 22,400 models on 14 benchmarks. Resolution attaches to the (dataset, *model class*) pair: on identical Cora test nodes a graph-free MLP has $D = 1.4$ and a GCN 3.9. Depth leaves it flat, and size does not buy it, since the 48,603 test nodes of ogbn-arxiv carry the information of roughly 700. Nothing in the bootstrap needs the statistic to be a mean, so the same machinery gives the design effect of ROC-AUC; auditing all 86 published gaps on the scale each was reported on, 87% fall below their benchmark's measured resolution. A claim about an *architecture* needs one term more: training randomness is a median 28% of the variance of a paired comparison, and adding it takes the rate at which a seed change alone is called significant from 11.9% to 0.4%. Code, the cached per-node correctness vectors and the audit corpus accompany this submission.

1 INTRODUCTION

A benchmark is an instrument, and every instrument has a resolution: a smallest difference it can distinguish from its own noise. For most of machine learning it is the binomial standard error $\sqrt{p(1-p)/m}$, and test sets are sized so that the differences a community cares about sit above it. This paper is about a setting where that calculation is silently wrong. In transductive node classification, a message-passing network's predictions at two test nodes are functions of overlapping neighbourhoods of a *single shared graph*; their errors are therefore correlated, and the variance of test accuracy carries a covariance term the binomial formula omits. On Cora, we measure that a 1,000-node test set behaves, for a two-layer GCN, like 262 independent nodes; on ogbn-arxiv, 48,603 test nodes behave like roughly 700.

The community has long documented symptoms of unreliability: rankings that reverse across splits (Shchur et al., 2018), comparisons confounded by selection protocols (Errica et al., 2020), subfields whose progress dissolved with a dataset artifact (Platonov et al., 2023). Missing is the quantity that turns symptoms into a diagnosis: how large must a difference on a benchmark be before it means anything? We call this the *minimum resolvable gap* (MRG) and show it is computable, model-class-dependent, and, for the benchmarks the field reports most, larger than most of the differences the literature reports.

Three obstacles stand between the textbook theory of dependent sampling and a usable answer. First, the dependence must be located: we prove that the error-dependency graph of an L-layer message-passing network is contained in the $2L$-th power of the data graph (§3). Second, the classical plug-in estimator built on that picture (estimate hop-wise correlations $\bar{\rho}_d$, sum them against neighbourhood sizes $\bar{k}_d$; Wang et al., 2011) is unusable for three separate reasons (§4). Beyond a centring bias,

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.

we identify a previously undescribed *degree-sampling bias*: nodes enter a distance-d shell in proportion to their degree, and degree predicts accuracy, so the standard estimator reports errors as *anti*-correlated on 14 of 16 configurations on ogbn-arxiv, where the true design effect is 24–273. Fixing it still leaves a noise floor, which we bound, that grows where the design effect lives. Third, a surviving estimator must be validated against ground truth, which real data cannot provide; on synthetic error fields with known D over real topologies, a graph-partition block bootstrap (GPBB, §5) restores most of the coverage i.i.d. intervals lose, while every hop-based competitor saturates (§6).

With a validated instrument we measure the benchmarks as they are used (§7): 22,400 trained models across 14 datasets and 16 configurations in five model classes. **(i) Resolution is a property of the model class, not the benchmark alone.** On identical test nodes, graph-free MLPs have D near 1–1.7 while message-passing classes sit at roughly 2–4 on homophilous graphs (class medians): a dataset card cannot state one resolution number. **(ii) Depth is irrelevant, and our own prediction here was wrong.** The $2L$-support result suggests D should grow with depth; on Cora an SGC control that quadruples the receptive field while moving accuracy by one point leaves D flat, because measured error correlation concentrates at one hop, generated by label homophily rather than the nominal receptive field. **(iii) Size does not buy resolution.** ogbn-arxiv's design effect of 58–90 for well-fit message-passing models means its MRG is 2.6 points, not the 0.6 its size suggests; and a graph-free MLP reads $D = 165$ on the same partition, so the dependence lives in the data's regional structure, which message passing partially *averages away*. **(iv) Most published gaps are unresolvable.** Of 86 rank-1-vs-2 and rank-1-vs-5 gaps harvested from published tables, 87% fall below their benchmark's own measured MRG, scored on the metric each table reports, including ROC-AUC, whose design effect the same block bootstrap supplies; 14.9% of i.i.d.-significant pairwise comparisons do not survive GPBB, rising to 47.5% below one point. Meanwhile the expressiveness headroom a large theoretical literature studies is, with features, under 1% of the MRG on 11 of 14 datasets and exactly zero on 8. **(v) One correction is not enough.** Training randomness, invisible to any resampling of one test set, is a median 28% of the variance of a paired comparison, so a claim about an *architecture* resolves at $1.22\times$ the within-graph figure. The cost is small: where errors are essentially independent GPBB widens intervals by 3–15%.

Contributions. (1) A measurement framework for benchmark resolution: estimands, the receptive-field support proposition, model-class-indexed design effects, and the MRG. (2) A negative result with a mechanism: the hop-based plug-in family is unidentified on real error fields, with a new degree-sampling bias and a noise-floor bound. (3) GPBB, validated against known design effects on real topologies, with its coverage measured up to the range where it fails. (4) The largest controlled resolution study of these benchmarks to date (22,400 runs) and an audit of all 86 published gaps against it, each on the metric it was reported on. (5) An accounting of what a within-graph correction still misses, measured against a real-data null and against network cross-validation.

2 Related work

Evaluation pitfalls in graph learning. Shchur et al. (2018) showed that GNN rankings on the Planetoid benchmarks reverse under resampled splits; Errica et al. (2020) that published graph-classification comparisons conflate model selection with assessment; Platonov et al. (2023) that the standard heterophilous benchmarks contain duplicates and leakage; Palowitch et al. (2022) that real benchmarks occupy a sliver of graph-property space; Arnaiz-Rodríguez & Errica (2026) that several headline phenomena are weakly operationalized. Wang et al. (2021) document training "tricks" worth more than architecture changes; You et al. (2020) show best designs do not transfer. These works established that single-number comparisons mislead; none quantifies what a benchmark *can* resolve.

Dependence-aware evaluation. Closest to us is the relational-learning line of Neville et al. (2009); Wang et al. (2011); Neville et al. (2012), which showed that within-network evaluation inflates type-I error and proposed a one-hop analytic correction and network cross-validation (NCV). We adopt their framing. The analytic correction is quantitatively insufficient: at radius 4 it recovers about a sixth of the excess variance on the citation graphs and 1% on ogbn-arxiv, and on dense graphs is not reliably an under-correction at all; its multi-hop extension is unidentified (§4). NCV targets a different object, the fold-to-fold spread under retraining on disjoint node folds rather than the sampling

variance of a fixed trained model, and we measure it against ours on a real-data null (§6.1): it is the one procedure here calibrated for a claim about an *architecture*, because retraining is what makes training randomness visible. Network-HAC estimators (Kojevnikov et al., 2021) weight covariances by a kernel in network distance, which is the graph-HAC baseline of §6 with an oracle bandwidth; Leung (2023) obtains cluster-robust inference from low-conductance spectral clusters, the family GPBB belongs to, with blocks sized by the receptive field and a bootstrap in place of a plug-in variance. Miller (2024) popularized clustered standard errors in LM evaluation when the clustering variable is observed; here the clusters are induced by the graph and never observed, which is what GPBB constructs. Yauney et al. (2026) ask how small an LM benchmark can be before it stops resolving differences; we ask the transductive-graph version, where the obstacle is dependence, not count. The NLP community debated related failures of split-based inference (Gorman & Bedrick, 2019; Søgaard et al., 2021; Dror et al., 2018; Dodge et al., 2019; Tang et al., 2020); block-bootstrap theory goes back to Künsch (1989); Lahiri (2003), cluster-robust practice to Cameron et al. (2008); Cameron & Miller (2015), and the design-effect identity to Kish (1965). A separate line bootstraps *network statistics* of a single observed graph: node-resampling standard errors (Snijders & Borgatti, 1999), patchwork/snowball resampling (Thompson et al., 2016; Gel et al., 2017), and latent-space network bootstraps (Levin & Levina, 2019). GPBB shares the resample-subgraph-units move but targets a different estimand: the sampling variance of a *classifier's* test accuracy, where the dependence enters through a trained model's receptive field, not the graph statistic.

Expressiveness. A large literature bounds what message passing can represent (Xu et al., 2019; Morris et al., 2019; Chen et al., 2020; Zhang et al., 2024; Xu & Zou, 2025), and a growing empirical wing measures how much of that ceiling is realized (Zopf, 2022; Wang & Zhang, 2024; Pellizzoni et al., 2025; Kemper et al., 2025; Franks et al., 2024; Subramonian et al., 2023). We add the statistical floor: on all 14 datasets, featured 1-WL headroom is 0.00–0.25 points against MRGs of 0.13–10.4 (§7.5).

3 SETUP: ESTIMANDS, DEPENDENCE, AND RESOLUTION

Let $G = (V, E)$ be a graph with $n = |V|$ nodes, features X, labels Y, and a train/validation/test partition; let f be a classifier trained on the training nodes and $T \subseteq V$ the test set, $m = |T|$. Test accuracy is $\hat{a} = \frac{1}{m}\sum_{i \in T} c_i$ with correctness indicators $c_i = \mathbf{1}[f(i) = y_i]$.

Two estimands. Before analysing variance one must fix what $\hat{a}$ estimates. Under the *finite-population* (transductive) estimand a_F, the accuracy over the non-training nodes of *this* graph, T is a without-replacement sample and a finite-population correction shrinks variance (Wang et al., 2011). Under the *superpopulation* estimand $a_S = \Pr(f(v) = y_v)$ over a distribution of graphs of which G is one draw, no such correction applies. Whenever a paper claims "method A beats method B" generally, it claims a_S, and all inflation results below concern a_S. A third distinction cuts across these two: a_S is defined for the *trained* classifier f, while a claim about an architecture adds the variance of training itself. We measure that component and report both resolutions (§7.6); the graph-to-graph component stays unidentifiable from one graph (§8, Appendix K). (Conformal transductive guarantees (Zargarbashi et al., 2023; Huang et al., 2023) are compatible; Appendix K.)

Where the dependence lives.

Proposition 1 (Receptive-field support)**.** *Let f be an L-layer message-passing network whose layer ℓ aggregates over 1-hop neighbourhoods, trained on any subset of G. Conditional on the trained parameters θ (equivalently, on the training set, initialization and optimizer randomness), c_i and c_j are functions of disjoint inputs, hence independent under node-wise exogenous noise (label and feature noise independent across nodes given the graph), whenever $d_G(i, j) > 2L$. Equivalently, the dependency graph of $(c_i)_{i \in T}$ is a subgraph of the $2L$-th graph power G^{2L}.*

The proof (Appendix A) is elementary: $f(i)$ depends on the L-ball around i, and two L-balls intersect iff $d_G(i, j) \leq 2L$. It parameterizes every estimator below by the receptive field and generates a falsifiable prediction (Prop. 2). The exogenous-noise clause is a modelling device: real label and feature noise is itself spatially structured (the MLP rows of Table 1 measure that component), and G^{2L} bounds only what message passing *adds*.

Design effect and effective sample size. Write $\sigma^2 = p(1-p)$ with $p = 1 - \mathbb{E}[\hat{a}]$. Following Kish (1965), the *design effect* is $D = \mathrm{Var}(\hat{a}) \,/\, (\sigma^2/m)$, the factor by which dependence inflates variance

over an i.i.d. sample of the same size, and $n_{\text{eff}} = m/D$ is the number of independent observations the test set is worth.

Corollary 1 (Kish, 1965; Wang et al., 2011)**.** *$D = 1 + \sum_{d \geq 1} \bar{k}_d \, \bar{\rho}_d$, where $\bar{k}_d$ is the mean number of test partners at graph distance d from a test node and $\bar{\rho}_d$ the mean error correlation at distance d.*

The identity is standard, specialized at $d = 1$ by Wang et al. (2011); its promise (estimate $\bar{\rho}_d$, plug in) and its failure (§4) drive the paper.

Resolution.

Definition 1 (Minimum resolvable gap)**.** *For a paired comparison of systems A, B on the same test set with correctness correlation ρ_{AB}, the minimum resolvable gap at level α and power $1 - \beta$ is $\text{MRG} = (z_{1-\alpha/2} + z_{1-\beta}) \cdot \sqrt{2\sigma^2(1 - \rho_{AB})\, D_\Delta / m}$, where σ^2 is the common marginal variance under the null of equal accuracy and D_Δ is the design effect of the* difference $c_i^A - c_i^B$.

This is the classical minimum detectable effect in its large-sample form (the finite-block form is in §5); the only novelty is insisting on D_Δ, measured from the paired standard error, rather than assuming independence or borrowing the marginal D. Throughout, $\alpha = .05$, $\beta = .2$.

A falsifiable depth prediction.

Proposition 2 (Conditional depth monotonicity)**.** *Fix an error field with correlation profile $(\bar{\rho}_d)_{d \geq 1}$ and let the receptive field set only the summation horizon, $D(L) = 1 + \sum_{d \leq 2L} \bar{k}_d \bar{\rho}_d$. If $\bar{\rho}_d \geq 0$ for all d, then $D(L)$ is non-decreasing in L, and strictly increasing from L to L' if $\bar{\rho}_d > 0$ for some $d \in (2L, 2L']$.*

Our controlled simulation realizes exactly this setting and confirms the prediction (§6). On real benchmarks the strict clause fails: measured $\bar{\rho}_d$ is essentially zero beyond d=1, so a wider field adds nothing, and the profile is not even invariant to depth (§7.2). We state it before measuring, so the disagreement is a test, not a retrofit.

4 Why the natural estimator of D is unusable

Corollary 1 invites a plug-in estimator: measure $\bar{\rho}_d$ on distance-d test pairs, multiply by $\bar{k}_d$, sum. This family (Wang et al., 2011 at $d = 1$, its multi-hop extension, and a graph analogue of HAC/Newey–West weighting) fails on real error fields for three separable reasons.

(a) A centring bias (fixable). Centring each distance class on its own sample means induces $\mathbb{E}[\hat{\rho}_d] \approx \rho_d - 1/m$ even under independence; multiplied by geometrically growing $\bar{k}_d$ and summed, it drives the plug-in D below zero on exactly independent data. Referencing every class to the mean product over *all* distinct test pairs removes it; the shipped null test then holds $|\hat{\rho}_d|$ under 0.005.

(b) A degree-sampling bias (fixable, previously undescribed). A node enters a distance-d shell by being reached along edges, so shell membership is sampled roughly in proportion to degree, and degree predicts accuracy. On ogbn-arxiv the test error rate is 0.396 in the bottom degree quartile against 0.207 in the top, so partners at d=1 have mean error 0.261 against a global 0.308; centring on the global mean charges that marginal shift to the correlation and *flips its sign*. On **fourteen of the sixteen** configurations the textbook $\hat{\rho}_1$ is negative ($-.347$ to $+.021$) and its implied one-hop $\widehat{D}$ falls below 1 (-0.36 to 1.08), reporting the i.i.d. interval as *conservative* against a true design effect of 24–273; the MLP's $\widehat{D}_{\text{1hop}} = -0.36$ is outside the parameter space (Appendix E gives all sixteen). Recentring each shell on its own marginals, $\hat{\rho}_d^{\text{marg}} = (\mathbb{E}[e_a e_p \mid d] - \mathbb{E}[e_a|d]\,\mathbb{E}[e_p|d])/\sigma^2$ (a covariance standardized by the global σ^2, as Corollary 1 requires, not a correlation bounded by 1), restores a positive sign on all sixteen. A unit test isolates the mechanism: on a synthetic field with *zero* spatial correlation but degree-dependent error rates, textbook centring reports $\hat{\rho}_1 = -0.257$; corrected, -0.000.

(c) Non-stationarity and a noise floor (not fixable). Corollary 1 presumes a common marginal error rate; real fields violate it wildly (the error rate varies twofold across degree quartiles, and accuracy by 22 points across graph regions, §7.3). Once marginals vary, $\bar{\rho}_d$ is no longer a correlation and what the plug-in reports depends on a centring convention the identity does not fix: on Cora with

every test node as an anchor and a ground truth of $D = 4.1$, the textbook partial sums reach $\mathbf{-6.84}$ by d=6 and the corrected ones $\mathbf{7.15}$ at d=4 before falling back through the truth (Appendix A), so no principled stopping depth exists. Estimation noise compounds it:

Proposition 3 (Plug-in noise floor)**.** *For an error field whose pairwise error products are non-negatively correlated (independent or positively associated errors), a fixed centring constant, and n_a test anchors,* $\mathrm{se}(\bar{k}_d\hat{\rho}_d) \;\geq\; (\sigma_d/\sigma^2)\sqrt{\bar{k}_d/n_a}$*, where $\sigma_d^2 = \mathrm{Var}(e_i e_j)$ over distance-d pairs.*

The bound *grows* with d: the shells that carry the design effect are the ones the estimator cannot resolve. On ogbn-arxiv at the L=2 radius the four terms have magnitudes 0.5–17 against floors of 0.18–8.9; resolving the deepest to two decimals would need 1.1×10^8 anchors, 660× the node count. The estimator we use next never estimates $\bar{\rho}_d$; we report $\hat{\rho}_1, \hat{\rho}_2$ (both centrings) as diagnostics only.

5 GPBB: A GRAPH-PARTITION BLOCK BOOTSTRAP

The alternative to modelling correlation is to resample units that contain the dependence. Prop. 1 supplies the scale: for receptive field L, errors at distance $> 2L$ are independent, so blocks of diameter well above $2L$ interact only through their boundaries.

Estimator. Partition V into B contiguous blocks by capacity-constrained multi-source BFS, capping each at $1.15 \times \lceil n/B \rceil$ nodes (largest block under 7.8% here). Choose B from the dependence radius: $B = \mathrm{clip}(n/(4\,\overline{|\mathcal{B}_{2L}|}), 15, 200)$, where $\overline{|\mathcal{B}_{2L}|}$ is the mean $2L$-ball size. The rule targets blocks four times the dependence scale, but its floor of 15 binds at radius 4 on eleven of the fourteen, ten of which have blocks smaller than one $2L$-ball (§8); that leakage is why $\widehat{D}$ is a lower bound (§6). Resample blocks with replacement; recompute $\hat{a}^*$ as a ratio estimator over the test nodes of sampled blocks; take the standard deviation over B^*=2000 replicates, scaled by $\sqrt{B/(B-1)}$ (Cameron et al., 2008), and form a t_{B-1} interval. The paired version reuses the same block draws for both systems, preserving ρ_{AB}. Everything is seeded (Appendix B), and the partition never sees labels or errors.

Two implementation choices carry most of the weight (Appendix B): *balance*, since unconstrained BFS partitions put up to 84% of a graph in one block and inflate $\widehat{D}$ on exactly independent data by up to 3.40×; and *scale-factor bookkeeping*, since $\widehat{D}$ must be the ratio of the *unscaled* GPBB variance to the i.i.d. one, or the $\sqrt{B/(B-1)}$ factor manufactures a B-dependent trend. Across depths we hold the partition fixed at the coarsest radius (D_{com}), since B varies with L.

Any statistic, not just a mean. Corollary 1 is an identity about a mean of indicators, but resampling blocks and recomputing does not care what is recomputed. We therefore use the design effect in its general form, $D(T) = \mathrm{Var}(T)/\,\mathrm{Var}_{i.i.d.}(T)$, the denominator estimated by the i.i.d. bootstrap; for an accuracy this returns Kish's D up to Monte-Carlo error, and for ROC-AUC, the metric three of these benchmarks are reported on, it gives a resolution on the AUC scale (§7.4).

What GPBB does not claim. Blocks truncate dependence beyond the block diameter, so GPBB estimates a *lower bound* on D: every resolution number is optimistic for the field, and every "unresolvable" conclusion conservative. Coverage degrades gracefully but measurably at large dependence ranges (§6). Two bookkeeping points, both of the same sign (Appendix B): the tabulated MRG uses normal quantiles and the unscaled standard error, so the t_{B-1} interval at B=15 needs a 1.11× larger gap; and stranded nodes leave some blocks disconnected (under 5.5% of nodes on twelve benchmarks at radius 4, 9–11% on two), which is leakage.

6 CALIBRATION WHERE GROUND TRUTH EXISTS

Real data cannot validate a variance estimator, because the true D is unobservable. We therefore build error fields with known dependence on *real graph topologies*: latent Gaussian fields smoothed by K rounds of neighbourhood averaging, thresholded to error rate p=0.2, so the dependence radius is exactly $2K$ by construction and D_{true} is known by Monte Carlo.

Figure 1 shows the Cora topology; three more are in Appendix C. At K=0 every method is nominal: none pays a spurious price under independence. As K grows, i.i.d. coverage collapses (0.754

Table 1: Design effect by dataset and model class (class medians; GPBB, radius-matched partitions), n_{eff} and MRG at GCN-$L2$; MRG is the median over architecture pairs at radius 4 (large-sample form, Appendix B). Right: published gaps below the measured MRG. †: ROC-AUC source tables, not scored. ‡: the receptive-field-0 partition cannot see region-scale dependence; on the common partition the ogbn-arxiv MLP reads 165, above every message-passing class (§7.3).

dataset	m	MLP	GCN	SGC	SAGE	GAT	n_{eff}	$\text{MRG}_{i.i.d.}$	MRG	$<$MRG
ogbn-arxiv	48,603	24.8‡	59.7	185.7	65.0	86.9	721	0.57	2.64	2/2
tolokers	2,940	9.1	9.6	10.7	9.1	10.7	285	0.99	0.88	†
cora	1,000	1.40	3.89	4.02	3.70	4.02	262	3.23	3.64	8/8
chameleon	456	2.11	3.84	2.78	3.36	3.28	127	6.62	8.89	1/2
amazon-ratings	6,123	1.65	3.77	3.54	3.40	3.85	2,173	1.86	2.51	9/10
questions	12,231	1.56	2.96	3.38	2.54	2.87	4,328	0.10	0.13	†
citeseer	1,000	1.28	2.38	2.56	2.21	2.96	483	3.58	4.29	9/10
pubmed	1,000	1.27	2.19	2.36	1.83	2.49	545	3.47	4.16	10/10
squirrel	1,041	1.39	1.98	2.34	1.51	2.71	526	4.87	5.70	1/2
squirrel-filt.	416–483	1.37	1.43	1.60	1.37	1.67	350	6.47	6.10	4/4
chameleon-filt.	159–194	1.29	1.35	1.46	1.25	1.43	147	10.34	10.38	4/4
minesweeper	2,500	1.06	1.18	1.18	1.14	1.18	2,282	0.55	0.50	†
roman-empire	5,666	0.99	1.01	0.91	1.11	0.98	5,958	1.96	1.98	5/10
actor	1,520	0.97	0.92	0.91	0.77	0.88	1,713	3.85	3.51	–

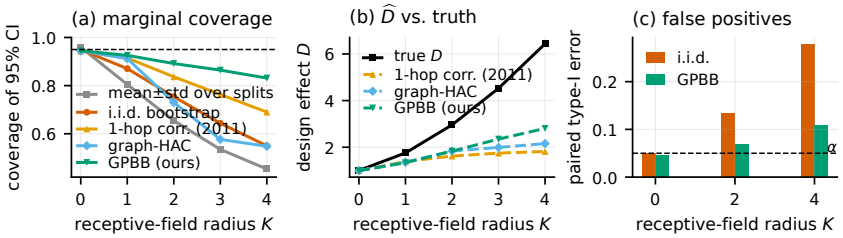

Figure 1: Ground-truth calibration on the Cora topology; error fields with smoothing radius K (dependence radius $2K$) and known D. (a) i.i.d. intervals and split-resampling collapse; the 1-hop analytic correction saturates; GPBB degrades slowly. (b) All estimators underestimate D; GPBB least from $K{=}3$ on, with the best coverage of the five at every $K \geq 1$. (c) Type-I error for a paired test of two identical systems.

at $K{=}2$, 0.550 at $K{=}4$). The popular mean±std-over-splits heuristic is *worse* (0.454 at $K{=}4$), since resampling splits of the same graph resamples the same dependence. The one-hop correction (Wang et al., 2011) helps and then saturates (0.690), and graph-HAC is worse still, falling to 0.548, back at the uncorrected level. GPBB holds 0.892 and 0.831. Its $\widehat{D}$ at $K{=}2$ is 1.83 against a truth of 2.96 ± 0.09, recovering 42% of the *excess* variance and 62% of the total: the lower bound in numbers. Panel (c): on two systems of *equal true accuracy* sharing 90% of their noise, the i.i.d. paired bootstrap false-positives 13.3% of the time at $K{=}2$ and 27.9% at $K{=}4$; GPBB cuts these to 6.9% and 10.8%.

6.1 A SECOND NULL, WITH NO SYNTHETIC FIELD

The study above knows D because the field is artificial; the benchmarks supply a complementary null in which the data is real instead. Two runs of one configuration on one split differ only in the training seed, so any claim that *the architecture* differs is false by construction. Over 8,326 such comparisons across 14 datasets and all 16 configurations, the i.i.d. paired bootstrap calls a seed change significant 17.0% of the time and GPBB 11.9% (49% and 29% on ogbn-arxiv). Neither errs about the two *trained models*, whose accuracies do differ; both err about the architecture, which is what the literature reports. Network cross-validation (Neville et al., 2012), which retrains on k disjoint subnetworks and tests across them, targets the architecture directly and holds at 4.8% against a nominal 5% on this null (792 comparisons, Appendix I) where the within-graph intervals are at

14.1% and 11.8%. It pays for that with k retrainings per comparison, and cannot be applied to a model someone else trained. §7.6 gets the same accounting from the grid we already have.

7 THE RESOLUTION OF REAL BENCHMARKS

Design. 14 datasets (Planetoid, the Platonov heterophilous suite in filtered and original versions, geom-gcn variants, ogbn-arxiv) × 16 configurations (MLP, GCN, SAGE, GAT at 1–8 layers, SGC at $K \leq 8$ hops; Appendix D) × 10 splits × 10 seeds = 22,400 trained models (Kipf & Welling, 2017; Veličković et al., 2018; Hamilton et al., 2017; Wu et al., 2019; Yang et al., 2016; Pei et al., 2020; Hu et al., 2020; Platonov et al., 2023). Training uses standard recipes (GCN reaches 80.5% on Cora, matching published values); per-node correctness vectors are cached, so every number below is recomputable without a GPU. Full protocol in Appendix D.

7.1 RESOLUTION IS A PROPERTY OF THE MODEL CLASS

Table 1 gives D by dataset and model class as class *medians* (Figure 3 plots them), robust to degenerate fits (the collapsed 8-layer GCN on Cora, the underfit 1-layer GCN on ogbn-arxiv). On homophilous citation graphs, message-passing classes carry $D \approx 2$–4 while the graph-free MLP sits at 1.3–1.4 on its own radius-matched partition: *on identical test nodes*, choosing a GNN over an MLP multiplies variance by roughly 2–3×. On the heterophilous suite as cleaned by Platonov et al. (2023) (filtered sets, roman-empire, actor, minesweeper), $D \approx 1$ for every class, so the i.i.d. interval is already right. Two datasets break the pattern: tolokers, nominally heterophilous, has $D \approx 10$ for *every* class including the MLP, and ogbn-arxiv reaches 25–186 (§7.3). GAT's high figure there is not an artefact of its self-loop-free neighbourhood: attending over $\mathcal{N}_i \cup \{i\}$ buys 0.7–2.6 accuracy points and moves D by 4–8% (Appendix E). Since D varies by 2.9× across model classes on Cora's nodes, and on ogbn-arxiv even the *ordering* of classes depends on the partition scale (the MLP's 24.8 is 165 on the common partition, §7.3), **a dataset card cannot state a single resolution number**.

7.2 DEPTH IS IRRELEVANT: A REFUTED PREDICTION

Prop. 2 predicts D grows with receptive field, and in simulation, where dependence really is generated at radius $2K$, it does (Fig. 1b). On real benchmarks it does not (Fig. 3b, Appendix E). The designed control is SGC, which widens the receptive field with no learned message passing and none of the collapse deep GCNs suffer: on Cora, from K=2 to K=8 the receptive field quadruples while accuracy moves one point (0.792/0.804/0.802, below Cora's MRG) and D_{com} stays at 4.18/4.13/3.97. Plain GCN matches until it degenerates (L=8: 27.6% accuracy, D_{com}=10.6), a broken model rather than a wider field. The mechanism is visible in the correlation profile: measured $\hat{\rho}_1 \approx 0.3$–0.4 but $\hat{\rho}_2 \approx 0.02$ regardless of depth. **The dependency graph G^{2L} of Prop. 1 bounds only what message passing adds, and even that bound is far from tight: real error dependence is generated by one-hop label homophily in the data, not by the nominal receptive field.** We predicted otherwise: Prop. 2 is a conditional whose strict clause the data refutes, and whose fixed-profile premise real models violate too (Appendix A). Resolution depends on the model *class* (graph-free vs. message-passing), not on depth within the class.

7.3 OGBN-ARXIV: SIZE DOES NOT BUY RESOLUTION, AND THE DEPENDENCE IS IN THE DATA

ogbn-arxiv was included as the large-test-set control: with m=48,603 (49× Cora), naive theory promises $\mathrm{MRG}_{i.i.d.} = 0.57$ points. Measured, well-fit message-passing models ($L \geq 2$; GCN, SAGE, GAT) carry $D = 58$–90, so $n_{\mathrm{eff}} \approx 700$, comparable to Cora despite 49× the nodes, and $\mathrm{MRG} = 2.64$ points. (The underfit one-layer GCN reaches D=149 and linear SGC 174–273: badly-fit models inflate D, §8.) Because the number is extreme we verified it with a decomposition sharing no code with GPBB: between-block accuracy variance over 15 balanced blocks gives D_{anova} within 10% of the same-partition bootstrap on all 16 configurations (median 7.5%), against 0.97 on independent Bernoulli errors. Per-block accuracy for GCN-$L2$ spans 0.601–0.825: contiguous regions differ by 22 points. On the official temporal split alone the numbers are milder and the conclusion the same: GCN-$L2$ has D=45, $n_{\mathrm{eff}} \approx$ 1,100 and $\mathrm{MRG} = 2.2$ points (Appendix E).

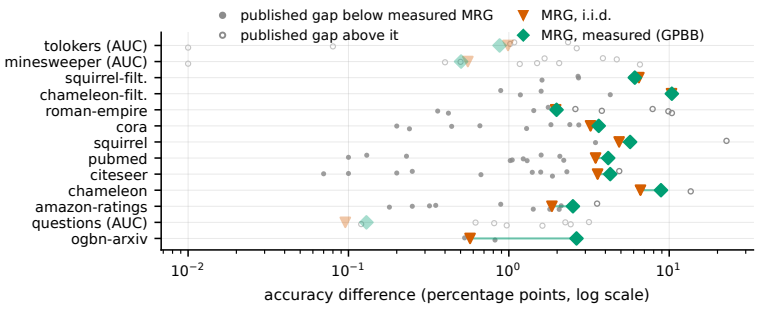

Figure 2: Published within-table gaps (dots) against each benchmark's minimum resolvable gap, assumed-independent ($\triangledown$) and measured ($\diamond$). Filled dots fall below the measured MRG (80% power): 75 of 86 gaps (87%), 53 of 62 on accuracy and 22 of 24 on AUC. Rows marked (AUC) are scored on the AUC scale against their own measured AUC design effect (Appendix H).

The MLP row settles it. On the *same* partition, the graph-free MLP has $D_{\text{anova}} = 177$ (block accuracy 0.313–0.725), 2.5× the GCN's, and an over-smoothed SGC-$K8$ reaches 280. Message passing is therefore not the source: subject areas differ in intrinsic difficulty, the features carry that structure, any model inherits it, and neighbourhood averaging partially *removes* it, which is why the better models have the smaller design effects here. The hop-based estimators of §4 fail here for the same reason: region-level heterogeneity at the scale of 10^4 nodes is invisible to a truncated sum over hop shells, and visible to resampled blocks.

7.4 The audit: published gaps against measured resolution

We harvested 86 published gaps, each a rank-1-vs-2 or rank-1-vs-5 difference computed *within* a single table, never across papers (Appendix F). Each is compared to its benchmark's measured MRG at dependence radius 4, on the scale its source table reports: for three binary datasets (24 gaps) that is ROC-AUC, whose design effect the block bootstrap gives directly (§5). It is substantial and again model-class-indexed (on tolokers $D(\text{AUC})$ runs from 2.9 for the MLP to 12.8 for GAT), putting those MRGs at 4.3, 4.7 and 5.9 AUC points against gaps of 0.01–6.6. **75 of the 86 published gaps (87%) fall below the resolution limit of the benchmark they were reported on**: 53 of 62 on accuracy and 22 of 24 on AUC (Figure 2, Appendix H). The MRG is an 80%-power threshold, so a gap below it can still be nominally significant; against the 5% rejection threshold itself, $1.96\,\widehat{\text{se}}_\Delta$ (70% of the MRG), 44 of the 62 accuracy gaps (71%) still fall short, and 48 fall below GPBB's finite-block threshold (Appendix B). Re-scoring with pair-class-matched MRGs, up to 23% smaller for GNN pairs, leaves the tally at 53 of 62. On Cora, CiteSeer, and PubMed, the field's most-reported benchmarks, the median rank-1-vs-2 gap is 0.20–0.34 points against MRGs of 3.6–4.3: an order of magnitude below what those test sets can distinguish. The two scales disagree about the three AUC benchmarks: near-ceiling accuracies shrink σ^2 and region-level dependence cancels in differences, while the AUC of the same predictions is far coarser (minesweeper, 0.50 accuracy points against 4.3 AUC points).

The same picture in paired form: across all $\binom{16}{2}$ architecture comparisons on 14 datasets at 20 (split, seed) replicates per pair (33,600 paired tests), 14.9% of i.i.d.-significant calls do not survive GPBB; restricted to gaps under 5 points the loss is 30.4%, under 2 points 38.0%, and under 1 point, the regime where modern leaderboard claims live, **47.5%**. GPBB is not nested inside the i.i.d. test: 331 comparisons go the other way, so the *net* fall is 13.3%.

7.5 The statistical floor vs. the expressiveness ceiling

Extending node-level 1-WL headroom measurement (Zopf, 2022; Subramonian et al., 2023; Kemper et al., 2025) to all 14 datasets: with node features, the accuracy an oracle measurable with respect to the $L{=}2$ WL colour partition leaves on the table is 0.000 median (max 0.251, actor) against a median MRG of 3.57 points, a ratio below 0.01 on 11 of 14. Feature uniqueness does part of this: on the four datasets where $\geq 99\%$ of feature vectors are distinct the headroom is already ≤ 0.07 at $L{=}0$,

but on the other ten it still falls from 0.3–20 points at $L{=}0$ to at most 0.25 by $L{=}2$ (Appendix D). **What limits a message-passing model on these benchmarks is not what it can represent but what the benchmark can measure**; sub-point gains attributed to expressiveness are unattributable.

7.6 What one training run cannot see

GPBB corrects the within-graph term; a claim about an architecture needs one more. Across 14 datasets the variance of a paired architecture difference over training seeds at a fixed split is a median 28% of the total paired variance. It is 3% on ogbn-arxiv, where the within-graph term is large, and 38–43% on Cora, CiteSeer, minesweeper and squirrel-filtered, the datasets on which §7.7 finds GPBB nearly free. The terms are complementary, so a method-level claim resolves at

$$\mathrm{MRG}_{\mathrm{method}} = (z_{1-\alpha/2} + z_{1-\beta})\sqrt{\widehat{\mathrm{se}}^2_{\mathrm{within}} + \hat{s}^2_{\mathrm{train}}},$$

both terms measured: $1.22\times$ the within-graph MRG at the median (1.11–1.77), leaving 54 of the 62 accuracy-metric gaps unresolvable instead of 53. Adding the seed term takes the seed-change rejection rate of §6.1 from 11.9% to 0.4%. That is conservative, since the null varies only the seed while the interval also covers test-set resampling, but ten training runs buy it where NCV needs ten retrainings *per fold*. **Report the within-graph MRG beside the seed spread**: both come from runs a benchmark paper already does (Appendix J).

7.7 What the correction costs where it is not needed

On the five $D \approx 1$ datasets, GPBB's paired intervals are only 3–15% wider than i.i.d. (median ratio 1.03 on roman-empire, 1.15 on chameleon-filtered), the excess fully accounted for by finite-partition inflation (Appendix B), and a third of individual intervals come out *narrower*. On actor, the topology where blocks leak most (Appendix C), the block-free hop diagnostic agrees: $|\hat{\rho}_d^{\mathrm{marg}}| \le 0.02$ at $d{=}1$ and < 0.005 at $d{=}2..5$ on every actor configuration. Separately, the GPU-nondeterminism floor from 30 re-trained cells is 0.13 points of median accuracy, 3.6% of a typical MRG (Appendix G).

8 Limitations

GPBB's D is a lower bound, and its coverage degrades with the dependence radius: 0.83 at $K{=}4$ on the Cora topology and as low as 0.49 on actor, whose dense core lets $2K$-balls swallow most of the graph. That regime is the norm here: at radius 4 a block holds fewer nodes than one dependence ball on 10 of the 14, from $0.62\times$ on PubMed to $0.07\times$ on tolokers. It is a property of the benchmark, not of our partitioner: no low-conductance partition into 15 blocks exists on actor, and neither boundary refinement nor spectral clustering recovers more than 0.05 of coverage there (Appendix B). Every such failure understates dependence, so our resolution figures are optimistic and the audit conservative. Two components stay outside: the graph-to-graph variance, invisible with a single graph, and metrics whose block resampling we have not worked out beyond the ROC-AUC of §7.4, such as macro-F1 and ranking metrics on edges. D is a joint property of data and model, and badly-fit models inflate it, so every figure is indexed by configuration. The gap corpus is hand-assembled, not a systematic survey, and search-budget and trick confounds (Dodge et al., 2019; Tang et al., 2020; Wang et al., 2021) compound ours.

9 Conclusion

Measuring design effects across 22,400 trained models on 14 benchmarks answers how large a difference on a node-classification benchmark must be before it means anything. Resolution belongs to the (benchmark, model class) pair, not to the benchmark alone. Size is a poor proxy for it: the 48,603 test nodes of ogbn-arxiv carry the information of roughly 700. Most published gaps we sampled fall below their own benchmark's resolution. Where dependence is absent the correction costs 3–15% of interval width, so reporting it by default is nearly free, and the released caches let a benchmark publish its resolution beside its size. Two questions stay open: inference at dependence radii too large for blocks to contain, and the graph-to-graph variance no single-graph method can see.

Ethics statement

This work analyses public benchmark datasets and published aggregate results; no human subjects, personal data, or deployment risks are involved. Its findings bear on scientific practice: some published comparisons are, on our evidence, statistically unresolvable, which we present as a property of the instruments rather than a criticism of individual works.

Reproducibility statement

The supplementary material contains all code, the cached per-node correctness vectors for all 22,400 runs, the published-gap corpus with per-gap provenance, unit tests covering every estimator invariant used in the paper (51 tests, including the sign-flip regression test of §4), and a script that rebuilds all twelve tables in the paper from the result files and checks them against the LaTeX source. Every design effect, resolution figure, plug-in diagnostic, ablation and audit number in the paper regenerates from those caches on a single CPU core, with no retraining and no dataset download: the released commands reproduce the shipped result files, and a full pass over all 14 datasets takes 33 minutes on one core. Four measurements sit outside that and say so where they are reported: the reproducibility floor of Appendix G and the self-loop GAT of Appendix E, which need training runs; the ROC-AUC resolution of Appendix H, which needs the three binary benchmarks retrained with their predicted scores cached, since a correctness vector cannot produce a rank statistic (those score caches *are* shipped, so the analysis itself re-runs on CPU); network cross-validation (Appendix I), which retrains ten folds by construction; and the WL headroom of §7.5, which reads node features and so needs the two documented commands that restore them, three for ogbn-arxiv. Node features are omitted from the shipped graph caches, since only retraining and the featured WL-headroom measurement read them, and one documented command restores them from the public sources. Datasets are public; splits and seeds are fixed and enumerated in Appendix D.

AI use statement

In this work, we used generative AI tools to assist in writing, editing, and debugging code, including code used to implement our methods and experiments. We did not use generative AI tools for any of the other tasks that require disclosure: in particular, we did not use them to generate synthetic data, to develop theoretical models or conceptual frameworks, to formulate mathematical claims or assist in proofs, to propose or refine hypotheses, to design the research methodology or experiments, to assist with translation, to clean or reformat datasets, or to support data analysis or the interpretation of results. Additionally, we used generative AI tools to edit the manuscript to improve readability and clarity, and to obtain simulated peer-review-style feedback on drafts, which informed revisions to the writing and presentation. All AI-assisted code was reviewed and tested for correctness by the authors, and all AI-suggested edits and feedback were reviewed and adopted at the authors' discretion; the scientific content, claims, and results are entirely the authors' own. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

References

Adrián Arnaiz-Rodríguez and Federico Errica. Oversmoothing, "oversquashing", heterophily, long-range, and more: Demystifying common beliefs in graph machine learning. In *International Conference on Learning Representations (ICLR)*, 2026.

A. Colin Cameron and Douglas L. Miller. A practitioner's guide to cluster-robust inference. *Journal of Human Resources*, 50(2):317–372, 2015.

A. Colin Cameron, Jonah B. Gelbach, and Douglas L. Miller. Bootstrap-based improvements for inference with clustered errors. *The Review of Economics and Statistics*, 90(3):414–427, 2008.

Zhengdao Chen, Lei Chen, Soledad Villar, and Joan Bruna. Can graph neural networks count substructures? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Jesse Dodge, Suchin Gururangan, Dallas Card, Roy Schwartz, and Noah A. Smith. Show your work: Improved reporting of experimental results. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. The hitchhiker's guide to testing statistical significance in natural language processing. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

Federico Errica, Marco Podda, Davide Bacciu, and Alessio Micheli. A fair comparison of graph neural networks for graph classification. In *International Conference on Learning Representations (ICLR)*, 2020.

Billy J. Franks, Christopher Morris, Ameya Velingker, and Floris Geerts. Weisfeiler-leman at the margin: When more expressivity matters. *arXiv preprint arXiv:2402.07568*, 2024.

Yulia R. Gel, Vyacheslav Lyubchich, and Lilia L. Ramirez-Ramirez. Bootstrap quantification of estimation uncertainties in network degree distributions. *Scientific Reports*, 7(1):5807, 2017.

Kyle Gorman and Steven Bedrick. We need to talk about standard splits. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Kexin Huang, Ying Jin, Emmanuel Candès, and Jure Leskovec. Uncertainty quantification over graph with conformalized graph neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Niklas Kemper, Tom Wollschläger, and Stephan Günnemann. What expressivity theory misses: Message passing complexity for GNNs. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

Leslie Kish. *Survey Sampling*. John Wiley & Sons, 1965.

Denis Kojevnikov, Vadim Marmer, and Kyungchul Song. Limit theorems for network dependent random variables. *Journal of Econometrics*, 222(2):882–908, 2021.

Hans R. Künsch. The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics*, 17(3):1217–1241, 1989.

Soumendra N. Lahiri. *Resampling Methods for Dependent Data*. Springer, 2003.

Michael P. Leung. Network cluster-robust inference. *Econometrica*, 91(2):641–667, 2023.

Keith Levin and Elizaveta Levina. Bootstrapping networks with latent space structure. *arXiv preprint arXiv:1907.10821*, 2019.

Evan Miller. Adding error bars to evals: A statistical approach to language model evaluations. *arXiv preprint arXiv:2411.00640*, 2024.

Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *AAAI Conference on Artificial Intelligence*, 2019.

Jennifer Neville, Brian Gallagher, and Tina Eliassi-Rad. Evaluating statistical tests for within-network classifiers of relational data. In *IEEE International Conference on Data Mining (ICDM)*, 2009.

Jennifer Neville, Brian Gallagher, Tina Eliassi-Rad, and Tao Wang. Correcting evaluation bias of relational classifiers with network cross validation. *Knowledge and Information Systems*, 30: 31–55, 2012.

John Palowitch, Anton Tsitsulin, Brandon Mayer, and Bryan Perozzi. GraphWorld: Fake graphs bring real insights for GNNs. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2022.

Hongbin Pei, Bingzhe Wei, Kevin Chen-Chuan Chang, Yu Lei, and Bo Yang. Geom-GCN: Geometric graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2020.

Paolo Pellizzoni, Till Hendrik Schulz, and Karsten Borgwardt. Graph neural networks can (often) count substructures. In *International Conference on Learning Representations (ICLR)*, 2025.

Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. A critical look at the evaluation of GNNs under heterophily: Are we really making progress? In *International Conference on Learning Representations (ICLR)*, 2023.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *Relational Representation Learning Workshop, NeurIPS*, 2018.

Tom A. B. Snijders and Stephen P. Borgatti. Non-parametric standard errors and tests for network statistics. *Connections*, 22(2):161–170, 1999.

Anders Søgaard, Sebastian Ebert, Jasmijn Bastings, and Katja Filippova. We need to talk about random splits. In *Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, 2021.

Arjun Subramonian, Adina Williams, Maximilian Nickel, Yizhou Sun, and Levent Sagun. Weisfeiler and leman go measurement modeling: Probing the validity of the WL test. *arXiv preprint arXiv:2307.05775*, 2023.

Raphael Tang, Jaejun Lee, Ji Xin, Xinyu Liu, Yaoliang Yu, and Jimmy Lin. Showing your work doesn't always work. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

Madeleine E. Thompson, Lilia L. Ramirez-Ramirez, Vyacheslav Lyubchich, and Yulia R. Gel. Using the bootstrap for statistical inference on random graphs. *Canadian Journal of Statistics*, 44(1): 3–24, 2016.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Tao Wang, Jennifer Neville, Brian Gallagher, and Tina Eliassi-Rad. Correcting bias in statistical tests for network classifier evaluation. In *Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, 2011.

Yanbo Wang and Muhan Zhang. An empirical study of realized GNN expressiveness. In *International Conference on Machine Learning (ICML)*, 2024.

Yangkun Wang, Jiarui Jin, Weinan Zhang, Yong Yu, Zheng Zhang, and David Wipf. Bag of tricks for node classification with graph neural networks. *arXiv preprint arXiv:2103.13355*, 2021.

Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *International Conference on Machine Learning (ICML)*, 2019.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations (ICLR)*, 2019.

Tuo Xu and Lei Zou. Towards a complete logical framework for GNN expressiveness. In *International Conference on Learning Representations (ICLR)*, 2025.

Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. Revisiting semi-supervised learning with graph embeddings. In *International Conference on Machine Learning (ICML)*, 2016.

Gregory Yauney, Shahzaib Saqib Warraich, and Swabha Swayamdipta. How reliable is language model micro-benchmarking? In *International Conference on Learning Representations (ICLR)*, 2026.

Jiaxuan You, Rex Ying, and Jure Leskovec. Design space for graph neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Soroush H. Zargarbashi, Simone Antonelli, and Aleksandar Bojchevski. Conformal prediction sets for graph neural networks. In *International Conference on Machine Learning (ICML)*, 2023.

Bohang Zhang, Jingchu Gai, Yiheng Du, Qiwei Ye, Di He, and Liwei Wang. Beyond Weisfeiler-Lehman: A quantitative framework for GNN expressiveness. In *International Conference on Learning Representations (ICLR)*, 2024.

Markus Zopf. 1-WL expressiveness is (almost) all you need. In *International Joint Conference on Neural Networks (IJCNN)*, 2022.

A PROOFS

A.1 PROPOSITION 1 (RECEPTIVE-FIELD SUPPORT)

An L-layer message-passing network computes its logit at node i as a deterministic function of the multiset of features in the closed L-ball $\mathcal{B}_L(i)$ and of the trained parameters θ: $f(i) = g_\theta\big(X_{\mathcal{B}_L(i)}, A_{\mathcal{B}_L(i)}\big)$, by induction on layers (each layer's representation at a node depends on the previous layer's representations in its 1-hop neighbourhood). Condition on θ (equivalently, on the training set, initialization, and optimizer randomness). If $d_G(i,j) > 2L$ then $\mathcal{B}_L(i) \cap \mathcal{B}_L(j) = \emptyset$: a node u in both balls would give $d_G(i,j) \le d_G(i,u) + d_G(u,j) \le 2L$. Under a node-wise exogenous noise model (label noise and feature noise independent across nodes given the graph), $c_i = \mathbf{1}[f(i) = y_i]$ and $c_j = \mathbf{1}[f(j) = y_j]$ are then measurable functions of disjoint independent inputs, hence independent. The set of dependent pairs is therefore contained in $\{(i,j) : d_G(i,j) \le 2L\}$, i.e. the edge set of G^{2L}. □

Two remarks. First, conditioning on θ is the right coupling for evaluation: at test time the parameters are fixed, and the randomness that inference must average over is the realization of errors on the test set. Unconditionally, θ is a function of the whole training graph and induces additional *global* dependence; that component is exactly what block resampling within one graph cannot see, and it is part of the superpopulation gap discussed in §8 and Appendix K. Second, the bound is a support statement, not a magnitude statement, and it covers only the dependence that message passing induces under the exogenous-noise clause: when label or feature noise is itself spatially structured, as on the real benchmarks here (the graph-free MLP, L=0, has $D > 1$ on twelve of the fourteen, Table 1), that data-intrinsic component lies outside G^{2L} at every L. §7.2 shows empirically that even the message-passing component is far from tight.

A.2 COROLLARY 1 (DESIGN-EFFECT IDENTITY)

With $e_i = 1 - c_i$, $\mathrm{Var}(\hat{a}) = \frac{1}{m^2}\sum_{i,j\in T}\mathrm{Cov}(e_i,e_j) = \frac{\sigma^2}{m} + \frac{1}{m^2}\sum_{i\neq j}\mathrm{Cov}(e_i,e_j)$ under a common marginal error rate p (so $\mathrm{Var}(e_i) = \sigma^2 = p(1-p)$). Grouping ordered pairs by graph distance d and writing $\mathrm{Cov}(e_i,e_j) = \sigma^2\rho_{ij}$, $\sum_{i\neq j}\mathrm{Cov} = \sigma^2\sum_d m\,\bar{k}_d\,\bar{\rho}_d$ where $\bar{k}_d$ is the mean number of distance-d test partners per test node and $\bar{\rho}_d$ the k-weighted mean correlation at distance d. Dividing by σ^2/m gives $D = 1 + \sum_d \bar{k}_d\bar{\rho}_d$. □

The common-marginal assumption is essential and is precisely what real error fields violate (§4c); the identity is exact for the simulation of §6 by construction.

A.3 PROPOSITION 2 (CONDITIONAL DEPTH MONOTONICITY)

With the profile $(\bar{\rho}_d)$ held fixed, $D(L') - D(L) = \sum_{d=2L+1}^{2L'}\bar{k}_d\bar{\rho}_d \ge 0$ when every $\bar{\rho}_d \ge 0$, strictly positive if some $\bar{\rho}_d > 0$ in the added range ($\bar{k}_d > 0$ on a connected graph for all d up to the diameter). □

The fixed-profile premise is what the synthetic fields of §6 realize. For a trained model the profile itself depends on depth, $D(L') - D(L) = \sum_{d \le 2L} \bar{k}_d(\bar{\rho}_d^{(L')} - \bar{\rho}_d^{(L)}) + \sum_{d=2L+1}^{2L'} \bar{k}_d \bar{\rho}_d^{(L')}$, and the first sum can be negative: on ogbn-arxiv GCN's $\hat{\rho}_1^{\mathrm{marg}}$ falls from 0.260 at $L{=}1$ to 0.131 at $L{=}4$ (Appendix E). So on real data the proposition is refuted twice over: its strict clause has no positive $\bar{\rho}_d$ beyond $d{=}1$ to act on, and its premise does not hold.

A.4 Proposition 3 (plug-in noise floor)

The shell estimator averages products $e_a e_p$ over the $\approx n_a \bar{k}_d$ (anchor, partner) pairs at distance d. Write it as $\frac{1}{n_a}\sum_a Y_a$ with per-anchor means $Y_a = \frac{1}{k_a}\sum_p e_a e_p$. Expanding, $\mathrm{Var}(\frac{1}{n_a}\sum_a Y_a) = \frac{1}{n_a^2}\big[\sum_a \mathrm{Var}(Y_a) + \sum_{a \ne a'} \mathrm{Cov}(Y_a, Y_{a'})\big]$ and $\mathrm{Var}(Y_a) = \frac{1}{k_a^2}\big[k_a \sigma_d^2 + \sum_{p \ne p'} \mathrm{Cov}(e_a e_p, e_a e_{p'})\big]$. Under the proposition's assumption that products of errors are non-negatively correlated, both covariance sums are ≥ 0, so $\mathrm{Var}(Y_a) \ge \sigma_d^2/k_a$ and, by the harmonic–arithmetic mean inequality over anchors, $\mathrm{Var}(\frac{1}{n_a}\sum_a Y_a) \ge \frac{\sigma_d^2}{n_a \bar{k}_d}$. For binary errors the within-anchor inequality is strict even under independence, since products sharing an anchor share the idempotent factor e_a: $\mathrm{Cov}(e_a e_p, e_a e_{p'}) = p^3 - p^4 > 0$; the bound is therefore conservative, which is the direction that matters. (The implementation averages over pairs rather than anchors; the two weightings coincide when k_a is constant.) The plug-in term is $\bar{k}_d \hat{\rho}_d = \bar{k}_d(\widehat{\mathbb{E}}[e_a e_p | d] - r)/\sigma^2$ for a centring constant r, treated as fixed (the estimated centrings of §4(a)–(b) add sampling variance of their own), so $\mathrm{se}(\bar{k}_d \hat{\rho}_d) \ge \frac{\bar{k}_d}{\sigma^2} \cdot \frac{\sigma_d}{\sqrt{n_a \bar{k}_d}} = \frac{\sigma_d}{\sigma^2}\sqrt{\bar{k}_d / n_a}$. □

On Cora with complete anchoring, against a ground truth of $D = 4.1$ (block decomposition), the textbook partial sums $\widehat{D}_{\le d}$ run $1.57, 1.70, 1.71, 1.09, -1.25, -6.84$ for $d = 1..6$, wrong in sign and off by 10.9 against a noise floor of 0.8 that Prop. 3 shows is a best case, while the corrected version runs $1.62, 3.02, 4.83, 7.15, 6.20, 2.92$, passing through the truth and beyond it.

On small-world graphs $\bar{k}_d$ grows geometrically until it saturates near n, so the bound grows with d while the terms it applies to are exactly the large ones. Numerically on ogbn-arxiv (GCN-$L2$ radius, $n_a{=}141$ test anchors, from `code/plugin_diag.py`): corrected terms $(+0.46, +12.2, -16.8, -9.8)$ against floors $(0.18, 2.37, 6.30, 8.91)$. The floor is 39%, 19%, 37% and 91% of the term it bounds at $d{=}1..4$: at the model's own radius it is within a tenth of the quantity it is supposed to resolve.

B GPBB details and ablations

Algorithm. (i) Partition V into B blocks with capacity-constrained multi-source BFS: seed B nodes uniformly at random, allocated across connected components in proportion to component size; grow all fronts in round-robin, one node per front per pass; cap each block at $\lceil 1.15\, n/B \rceil$ nodes; and assign the nodes left over when every front adjacent to them is full or exhausted, one connected component of G at a time in node-index order, to whichever block is currently smallest, in chunks up to its remaining capacity. The receiving block need not be adjacent, so a block can consist of several pieces (quantified below). (ii) Choose $B = \mathrm{clip}(n/(4|\overline{\mathcal{B}_{2L}}|), 15, 200)$, where $|\overline{\mathcal{B}_{2L}}|$ is the mean $2L$-ball size estimated by BFS from 200 anchors on graphs up to 50,000 nodes and 40 above that, with early exit once the ball exceeds $n/60$, beyond which B would clip to its floor regardless. (iii) For $b = 1..B^*$ replicates: draw B blocks with replacement; let S be the multiset of test nodes they contain; $\hat{a}_b^* = \frac{1}{|S|}\sum_{i \in S} c_i$ (ratio estimator: the denominator varies). (iv) $\widehat{\mathrm{se}} = \sqrt{B/(B-1)} \cdot \mathrm{sd}(\hat{a}^*)$ and the interval is $\hat{a} \pm t_{B-1,1-\alpha/2}\widehat{\mathrm{se}}$. Paired comparisons reuse the same block draws for both systems. $\widehat{D} = \widehat{\mathrm{se}}^2_{\mathrm{raw}}/(\widehat{\sigma}^2/m)$ uses the *unscaled* sd, so that finite-B inflation is not booked as dependence.

Why balance carries weight (`code/gpbb_ablations.py`, 300 replicates of exactly independent errors on each of the 14 topologies). Plain multi-source BFS puts 24% of Cora, 58% of actor and 84% of questions into a single cell; the ratio estimator's denominator then becomes highly variable and covaries with its numerator, and the measured $\widehat{D}$ is inflated by a median of $1.11\times$ and up to $3.40\times$ on data with no dependence at all. The capacity constraint holds the largest block at

7.8% or below on every topology, after which $\widehat{D}$ on independent errors sits at 0.85–0.98: the raw block-bootstrap standard deviation slightly *under*-shoots at $B \approx 15$, which the $\sqrt{B/(B-1)}$ and t_{B-1} factors in the interval more than offset. That net cost is what §7.7 measures on real data.

Block contiguity (`code/partition_diag.py`, which rebuilds every partition the study uses and writes `results/partition_connectivity.json`). A piece of a block that is an entire connected component of G is harmless, since by Proposition 1 it shares no dependence with anything (CiteSeer has 438 components, Cora 78). Counting only pieces that cut a component of G across blocks, the radius-4 partitions used by the audit strand 0% of the nodes on squirrel, squirrel-filtered, tolokers and ogbn-arxiv, under 1.1% on Cora, PubMed, actor and chameleon, 2.5% on CiteSeer, 3.7% on questions, 4.8% on chameleon-filtered, 5.3% on minesweeper, 9.0% on amazon-ratings and 11.2% on roman-empire; where B=15 the stray pieces sit in at most two blocks. The finer partitions of the receptive-field-0 (MLP) models strand more, up to 13% on amazon-ratings. A stray piece is resampled together with a block it does not touch, which is boundary leakage of the usual sign: it can only understate D.

Finite-block resolution. Definition 1 uses normal quantiles and the unscaled bootstrap standard error. The interval GPBB forms uses $\sqrt{B/(B-1)}$ and t_{B-1}, so the gap at which its paired test has 80% power is $\left[(t_{B-1,.975} + t_{B-1,.8})/(z_{.975} + z_{.8})\right]\sqrt{B/(B-1)}$ times the tabulated MRG: 1.113 at B=15, 1.049 at B=32, 1.007 at B=200. Re-scoring the audit against the B=15 threshold leaves the count at 53 of 62.

Block-count sensitivity (`code/gpbb_ablations.py`; Cora topology, synthetic fields, 300 replicates, B^*=2000): coverage / $\widehat{D}$ by target B:

K (D_{true})	B=10	B=20	B=40	B=80
0 (1.00)	.947 / 0.78	.957 / 0.90	.963 / 0.97	.963 / 0.98
1 (1.75)	.937 / 1.25	.950 / 1.39	.900 / 1.36	.920 / 1.42
2 (2.96)	.907 / 1.73	.910 / 1.96	.877 / 1.90	.877 / 1.84
4 (6.43)	.853 / 2.89	.810 / 2.87	.823 / 2.64	.773 / 2.44

Fewer, larger blocks reduce boundary leakage (higher coverage) at the price of a heavier t_{B-1} quantile; the rule of §5 lands at $B \approx$ 15–20 on small graphs and was frozen before the real-data study. All choices were fixed prior to computing any number in §7.

What the correction costs under independence. On the five $D \approx 1$ datasets the paired GPBB interval is 3–15% wider than the i.i.d. one (§7.7), and the excess is the finite-partition factor rather than any measured dependence: $\sqrt{D_\Delta} \cdot \sqrt{B/(B-1)} \cdot t_{B-1}/z$ predicts 1.127 on actor at its B=15 blocks, against 1.123 measured.

Independent verification of extreme D. For ogbn-arxiv, we also computed a design effect that shares no code with GPBB: with a fixed 15-block balanced partition, $D_{\text{anova}} = \frac{B}{B-1}\sum_b w_b^2(\bar{a}_b - \hat{a})^2/(\sigma^2/m)$ with size weights w_b. Agreement with the same-partition bootstrap value is within 10% on every configuration, 7.5% at the median (GCN-$L2$: 71.5 vs. 68.6; MLP: 177 vs. 165), and the same statistic on independent Bernoulli errors at the observed error rate returns 0.97 (calibration control). `code/plugin_diag.py` recomputes the whole table.

Does a better partition rescue the dense-graph regime? (`code/partition_compare.py`; synthetic fields, 600 replicates, against the 3,000 of §6, which is why the coverages here differ from that section's by a few points.) Two explanations compete for GPBB's failure on actor: the capacity-constrained BFS partition is not very good, or the graph admits no good partition at this block count. Leung (2023) makes the second precise: low cluster *conductance*, the ratio of a block's edge boundary to its volume, is what licenses cluster-robust inference. We therefore measured conductance and coverage together for three partitioners: BFS as used throughout, a greedy boundary-refinement pass over it, and normalized spectral clustering (capacity-balanced afterwards, so the comparison is not confounded by balance).

				coverage of 95% intervals			median conductance		
topology	K	B	i.i.d.	BFS	refined	spectral	BFS	refined	spectral
cora	0	137	0.955	0.967	0.963	0.958	0.42	0.37	0.39
cora	2	15	0.762	0.925	0.933	0.828	0.29	0.22	0.80
cora	4	15	0.520	0.825	0.842	0.665	0.29	0.22	0.79
citeseer	0	200	0.960	0.957	0.960	0.950	0.24	0.20	0.68
citeseer	2	15	0.855	0.960	0.962	0.912	0.10	0.07	0.59
citeseer	4	15	0.748	0.935	0.938	0.823	0.10	0.07	0.72
roman-empire	0	200	0.965	0.965	0.962	0.958	0.02	0.02	0.01
roman-empire	2	200	0.938	0.958	0.952	0.948	0.02	0.02	0.01
roman-empire	4	126	0.923	0.940	0.942	0.943	0.01	0.01	0.01
actor	0	200	0.952	0.953	0.955	0.953	0.70	0.69	0.70
actor	2	15	0.755	0.797	0.817	0.823	0.62	0.55	0.73
actor	4	15	0.380	0.503	0.540	0.548	0.62	0.55	0.73

Conductance predicts coverage, monotonically and across an order of magnitude of it. At K=4: roman-empire's chain-like graph partitions at conductance 0.013 and every method is nominal; CiteSeer at 0.10 gives GPBB 0.935 against the i.i.d. interval's 0.748; Cora at 0.29 gives 0.825 against 0.520; and actor at 0.62, where a block loses as many edges as it keeps, gives 0.503 against 0.380, with *no* partitioner repairing it (refinement 0.540, spectral 0.548). The failure is a property of the benchmark, as the cluster-robust theory predicts, not of how we draw the blocks. Two secondary readings. Boundary refinement is a small, uniform gain (coverage $+1$ to $+4$ points) and raises $\widehat{D}$ with it ($2.75 \to 2.97$ on Cora at K=4), so the design effects reported in §7 are conservative in the direction claimed. Spectral clustering is *worse* on the citation graphs (0.665 against 0.825 on Cora at K=4), because balancing its clusters destroys the conductance that motivated them (0.79 against 0.29); contiguous BFS blocks already find the low-conductance structure where it exists. We report the refinement rather than adopt it: the partition rule was frozen before the real-data study (§5), and changing it now would trade a stated protocol for a few points of coverage.

C Simulation details and additional topologies

Error fields: draw $\varepsilon \sim N(0, I_n)$ on V, smooth with the symmetric-normalized adjacency $\hat{A} = \tilde{D}^{-1/2}(A + I)\tilde{D}^{-1/2}$ of the *real* graph, $z = \hat{A}^K \varepsilon$, and standardize each node by the exact ℓ_2 norm of its row of $\hat{A}^K$, so that $\tilde{z}_i \sim N(0, 1)$ at every node; the error is $e_i = \mathbf{1}[\tilde{z}_i > \Phi^{-1}(1 - p)]$ with p=0.2. The marginal error rate is therefore exactly p at every node whatever its degree (this is what makes Corollary 1 exact here), and $\tilde{z}_i$ depends on ε only within the K-ball of i, so pairs at distance $> 2K$ are independent by construction. The baselines in Figure 1: the *split* heuristic draws 10 test sets of m=1000 nodes from the same field and reports mean $\pm\, z_{.975}$·sd of the 10 accuracies (the sd is not divided by $\sqrt{10}$, matching how the field reads such intervals); the *one-hop* correction uses $\widehat{D} = 1 + \bar{k}_1 \hat{\rho}_1$ with a normal interval; *graph-HAC* uses $\widehat{D} = 1 + \sum_{d=1}^{2K} (1 - \frac{d}{2K+1})\, \bar{k}_d \hat{\rho}_d$, a Bartlett kernel in shortest-path distance truncated at the true dependence radius (an oracle bandwidth), $\hat{\rho}_d$ centred on the mean product over all distinct test pairs (§4a), every node as an anchor (a 4,000-anchor subsample where the pair index would exceed 40M), and a normal interval. The paired study draws two fields from latent noises with correlation 0.9 before smoothing and thresholding ($\varepsilon_A, \varepsilon_B = \sqrt{.9}\,\eta + \sqrt{.1}\,\xi_{A,B}$), both at p=0.2 for the type-I-error panel; 1,500 replicates. D_{true} is estimated by Monte Carlo over 3,000 field draws; measured across seeds, the Monte-Carlo standard deviation is about 3% of D at K=2 and 4% at K=4. The block-count table of Appendix B reads its D_{true} from this study rather than re-estimating it, so the two tables carry the same ground truth. Coverage: 3,000 replicates per cell, m=1000 test nodes drawn uniformly per replicate, B^*=2000. The estimand is the superpopulation error rate p, known exactly.

Coverage of 95% intervals, i.i.d. bootstrap vs. GPBB, on three further topologies:

topology	K=0	K=1	K=2	K=3	K=4
citeseer (D_{true})	1.03	1.55	2.04	2.68	3.32
i.i.d. / GPBB	.956 / .952	.898 / .940	.832 / .936	.775 / .927	.716 / .920
actor (D_{true})	1.01	1.31	2.72	6.32	14.51
i.i.d. / GPBB	.960 / .957	.905 / .923	.763 / .827	.559 / .651	.387 / .494
roman-empire (D_{true})	0.93	1.07	1.13	1.14	1.21
i.i.d. / GPBB	.958 / .953	.952 / .958	.942 / .953	.935 / .950	.939 / .958

The three regimes bracket practice: citeseer behaves like Cora; roman-empire's sparse, nearly chain-like graph generates almost no dependence even at K=4 (and GPBB stays nominal, paying nothing); actor's dense small-world core is the hard case: at K=4 its $2K$-balls cover most of the graph, blocks cannot contain the dependence, and GPBB fails visibly (0.494). We report this failure because it is the regime bound of the method (§8), and because it is detectable in advance from $\overline{|\mathcal{B}_{2K}|}/n$, which the implementation reports.

D Experimental setup

Datasets. All public, undirected, de-duplicated, self-loop-free; 10 splits per dataset (Planetoid: public split plus 9 resamples of matched sizes; Platonov suite and geom-gcn variants: the authors' 10 splits; ogbn-arxiv: the official temporal split plus 9 random splits of matched sizes; temporal-split results are reported separately in Appendix E, as they give systematically *lower* D).

dataset	n	edges	feats	classes	h_{edge}	m_{test}
cora	2,708	5,278	1,433	7	.81	1,000
citeseer	3,327	4,552	3,703	6	.74	1,000
pubmed	19,717	44,324	500	3	.80	1,000
actor	7,600	26,659	932	5	.22	1,520
chameleon	2,277	31,371	2,325	5	.23	456
squirrel	5,201	198,353	2,089	5	.22	1,041
chameleon-filt.	890	8,854	2,325	5	.24	159–194
squirrel-filt.	2,223	46,998	2,089	5	.21	416–483
roman-empire	22,662	32,927	300	18	.05	5,666
amazon-ratings	24,492	93,050	300	5	.38	6,123
minesweeper	10,000	39,402	7	2	.68	2,500
tolokers	11,758	519,000	10	2	.59	2,940
questions	48,921	153,540	301	2	.84	12,231
ogbn-arxiv	169,343	1,157,799	128	40	.65	48,603

Models and training. The 16 configurations are MLP $L \in \{2, 3\}$; GCN $L \in \{1, 2, 3, 4, 6, 8\}$; SAGE $L \in \{2, 4\}$; GAT $L \in \{2, 4\}$; SGC $K \in \{1, 2, 4, 8\}$. All models implemented in plain PyTorch from their defining equations (no graph-library layers), hidden width 64, dropout 0.5, Adam at learning rate 10^{-2} with weight decay $5 \cdot 10^{-4}$, the same for every dataset, up to 1,000 epochs with early stopping (patience 200) on validation accuracy; batch normalization on every dataset that reaches us through the heterophilous-graphs suite (which includes the geom-gcn datasets and their filtered variants) and on ogbn-arxiv, none on Planetoid, matching the datasets' canonical recipes. GAT uses 4 heads and attends over $\mathcal{N}_i$ rather than $\mathcal{N}_i \cup \{i\}$, since the shared edge list is self-loop-free; this differs from Veličković et al. (2018) and costs it accuracy on ogbn-arxiv (64.9% at L=2, against the 71–72% of the OGB reference GCN and GraphSAGE baselines, Hu et al., 2020). Since badly-fit models inflate D (§8), we re-ran both GAT cells with self-loops to check whether the design effect follows the configuration; it does not (Appendix E). Model selection uses validation accuracy only. Sanity anchor: GCN-L2 reaches 80.5% mean on Cora (public split: 80.9%), consistent with published values (Kipf & Welling, 2017; Shchur et al., 2018). Total compute: one 24 GB consumer GPU for approximately one day; every analysis re-runs from the released correctness caches on CPU.

Analysis protocol. GPBB uses B^*=2000 replicates per interval throughout. The i.i.d. bootstrap it is compared against scales its replicate count to the test-set size ($B^* m \leq 4 \times 10^6$, floor 200), since its target is the binomial standard error and the work is $O(B^* m)$ over thousands of calls; the

count actually used is recorded per configuration in the released result files. At the tightest setting, ogbn-arxiv's B^*=200, the i.i.d. standard error carries 4.8% Monte-Carlo noise, which by Jensen inflates the ratio $\widehat{D}$ by 0.7%; this is the only asymmetry between the two arms. Paired comparisons are capped at 20 (split, seed) replicates per configuration pair, taken from splits 0–1 with all ten seeds, while marginal quantities use all 100 runs per configuration. Correlation profiles follow a fixed schedule on graph size and mean degree: depth 3 with 500 anchors above 10^5 nodes or mean degree 40, depth 4 with 1,500 anchors above 20,000 nodes, depth 5 with 3,000 above 5,000, and depth 6 over all nodes below that, with a further reduction where the shell index would exceed its 40M-pair budget (questions). Five of the fourteen therefore run to depth 4 or 5. On Cora all 2,708 nodes enter the shell index, so every one of the 1,000 test nodes serves as an anchor.

E FULL PER-CONFIGURATION RESULTS

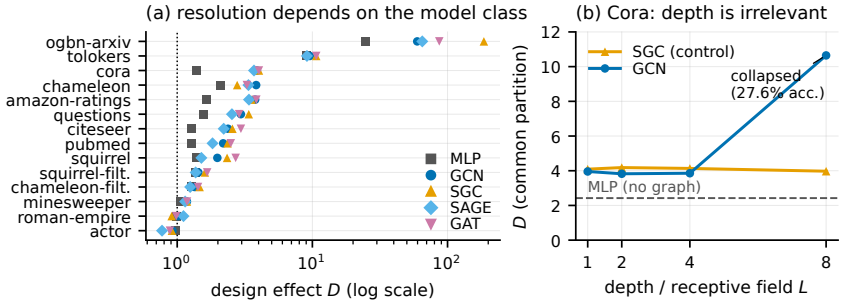

Figure 3: (a) Design effect by dataset and model class (class medians, log scale; radius-matched partitions; the ogbn-arxiv MLP reads 165 on the common partition, §7.3). The MLP column is the graph-free control. (b) The depth prediction fails on real data: SGC widens the receptive field 4× with accuracy within one point and constant D; the only depth effect is the collapsed 8-layer GCN.

ogbn-arxiv (all ten splits, the official temporal one included; $\widehat{D}$ by GPBB, radius-matched partition; D_{com} at the common partition; both plug-in variants shown for completeness). This is also the evidence for §4(b): the textbook $\hat{\rho}_1$ column is negative on fourteen of the sixteen rows and $\widehat{D}_{1\mathrm{hop}}$ below 1 on the same fourteen, against the $\widehat{D}$ column beside it; shell-marginal centring ($\hat{\rho}_1^{\mathrm{marg}}$) is positive on all sixteen and still recovers only 1–7% of $\widehat{D}$.

config	acc	$\widehat{D}$	D_{com}	$\widehat{D}_{1\mathrm{hop}}$	$\widehat{D}_{1\mathrm{hop}}^{\mathrm{marg}}$	n_{eff}	$\hat{\rho}_1$	$\hat{\rho}_1^{\mathrm{marg}}$
MLP $L2$	.4845	24.5	165.3	−0.36	1.77	1,986	−.347	+.186
MLP $L3$	.4759	25.2	172.2	0.21	1.85	1,928	−.191	+.214
GCN $L1$	.5844	149.4	147.7	1.04	2.18	325	+.007	+.260
GCN $L2$	.6933	67.4	68.6	0.95	1.79	721	−.014	+.175
GCN $L3$	.7038	60.7	59.9	0.52	1.63	801	−.103	+.143
GCN $L4$	.7073	58.1	58.2	0.52	1.61	836	−.107	+.131
GCN $L6$	.7055	58.4	58.6	0.69	1.67	832	−.063	+.162
GCN $L8$	.7004	58.7	58.7	0.79	1.71	829	−.040	+.163
SAGE $L2$	.6907	70.3	71.1	0.72	1.81	691	−.058	+.177
SAGE $L4$	.7072	59.7	59.0	0.60	1.67	814	−.089	+.145
GAT $L2$	.6486	89.5	89.1	1.08	2.00	543	+.021	+.234
GAT $L4$	.6343	84.2	83.2	0.89	2.03	577	−.031	+.223
SGC $K1$	.5346	182.4	185.5	0.46	2.06	266	−.144	+.244
SGC $K2$	.5621	173.7	171.8	0.65	2.16	280	−.094	+.279
SGC $K4$	.5332	189.0	191.1	0.77	2.32	257	−.067	+.316
SGC $K8$	.4650	272.9	268.1	0.95	2.67	178	−.027	+.385

Note the MLP rows: the radius-matched partition for a receptive-field-0 model uses many small blocks, which cannot see the region-scale heterogeneity; the common (coarse) partition reveals it (D_{com}=165–172), consistent with the estimator-independent decomposition (D_{anova}=177, §7.3). This is the block-scale dependence of §8: D is defined relative to a dependence range, and for

graph-free models on region-heterogeneous data the honest statement is the coarse-scale one. The same holds, far more mildly, on Cora: the MLP's D reads 1.40 on its own radius-matched partition and 2.42 on the GCN's coarser one, so the model-class gap quoted in §7.1 is $2.8\times$ at the first scale and $1.6\times$ at the second. On Cora the class ordering is the same either way; on ogbn-arxiv it is not: the MLP has the smallest D of any class on its radius-matched partition and the largest on the common one, which is why Table 1 flags that cell and §7.1 does not quote a cross-class ratio for it. The official temporal split gives systematically lower D than the nine random splits (the table below; D_{anova} for GCN-$L2$: 50 vs. 72; MLP: 95 vs. 178) because its test nodes are unevenly spread over blocks; SGC-$K8$ is the one exception (308 vs. 280), being unusually sensitive to *which* region is asked about (its accuracy also differs by 7 points between split types).

ogbn-arxiv by split protocol (`code/temporal_split.py`; the official temporal split alone, 10 seeds, against the nine random splits, 90 runs; same partitions and estimators as above; D_{anova} on the common 15-block partition):

	temporal split					random splits		
config	acc	$\widehat{D}$	D_{com}	D_{anova}	n_{eff}	acc	$\widehat{D}$	n_{eff}
MLP $L2$	.4756	19.9	91.8	94.6	2,448	.4855	24.6	1,974
MLP $L3$	.4740	19.9	89.2	88.2	2,449	.4762	25.7	1,893
GCN $L1$	.6023	75.4	76.4	75.8	644	.5824	151.3	321
GCN $L2$	.6890	45.1	45.4	50.4	1,077	.6938	68.1	714
GCN $L3$	.6990	41.9	41.7	44.9	1,159	.7043	61.0	796
GCN $L4$	.7019	41.4	41.3	45.0	1,176	.7079	59.0	824
GCN $L6$	.6986	41.2	42.0	45.8	1,179	.7062	59.6	816
GCN $L8$	.6924	42.8	43.3	45.5	1,136	.7013	59.4	818
SAGE $L2$	.6829	48.2	48.2	52.2	1,008	.6916	71.4	681
SAGE $L4$	.6968	43.4	43.6	46.0	1,121	.7083	60.6	802
GAT $L2$	.6498	60.2	58.3	62.4	807	.6485	91.4	532
GAT $L4$	.6288	57.7	58.1	59.2	842	.6349	85.0	572
SGC $K1$	.5047	129.9	121.4	135.9	374	.5380	186.5	261
SGC $K2$	.5441	82.2	82.9	87.1	592	.5641	178.4	272
SGC $K4$	.4919	112.9	116.3	121.9	430	.5377	191.4	254
SGC $K8$	.4023	283.1	293.0	308.0	172	.4720	271.7	179

On the temporal split every well-fit message-passing configuration ($L \geq 2$; GCN, SAGE, GAT) has $D = 41$–60 against 59–91 on random splits, so $n_{\mathrm{eff}} \approx 800$–$1{,}200$ rather than 530–820; the paired MRG at radius 4, from the ten temporal-split cells of each of the 22 radius-4 pairs, is 2.19 points against 2.64 pooled and 3.08 on random split 1. The two leaderboard gaps in the corpus (0.53 and 0.82 points, both measured on the temporal split) fall below all three. The pooled numbers are what Table 1 and §7.3 report, for consistency with the other thirteen datasets; a reader who wants the leaderboard's protocol should use this table.

GAT with and without self-loops (`code/gatsl_check.py`; `gatsl` attends over $\mathcal{N}_i \cup \{i\}$, everything else identical, same partitions; ogbn-arxiv on a balanced 5×4 sub-grid, Cora on the full 10×10):

		accuracy		$\widehat{D}$		n_{eff}		
dataset	depth	GAT	+self-loops	GAT	+self-loops	GAT	+self-loops	runs
cora	$L2$	.7847	.7970	4.1	4.0	244	249	100
cora	$L4$	.7789	.7893	3.9	3.9	254	255	100
ogbn-arxiv	$L2$	.6486	.6557	89.5	85.7	543	567	20
ogbn-arxiv	$L4$	.6343	.6605	84.2	77.2	577	630	20

Self-loops buy 1.0–1.2 accuracy points on Cora and 0.7–2.6 on ogbn-arxiv, leaving GAT still 3–5 points below GCN there; the design effect moves by -4% at $L{=}2$ and -8% at $L{=}4$, well inside the spread of the message-passing classes. GAT's elevated design effect on ogbn-arxiv is therefore a fact about the benchmark, not an artefact of the self-loop-free neighbourhood.

Cora (all 16 configurations; same conventions):

config	acc	$\widehat{D}$	D_{com}	n_{eff}	config	acc	$\widehat{D}$	D_{com}	n_{eff}
MLP $L2$	.556	1.42	2.50	702	SAGE $L2$	.775	3.59	3.59	279
MLP $L3$	.546	1.37	2.34	730	SAGE $L4$	.791	3.81	3.77	263
GCN $L1$	.747	3.97	3.96	252	GAT $L2$	.785	4.10	4.03	244
GCN $L2$	.805	3.82	3.82	262	GAT $L4$	.779	3.94	3.98	254
GCN $L3$	.804	3.90	3.88	257	SGC $K1$	.739	4.03	4.09	248
GCN $L4$	.791	3.86	3.86	259	SGC $K2$	.792	4.01	4.18	249
GCN $L6$	.765	3.88	3.82	258	SGC $K4$	.804	4.19	4.13	238
GCN $L8$	.276	10.38	10.65	96	SGC $K8$	.802	3.89	3.97	257

Per-configuration tables for the remaining twelve datasets are in the released result files (one JSON per dataset), together with the script that regenerates every table in the paper from them.

F THE PUBLISHED-GAP CORPUS

The corpus holds 86 gaps from published tables spanning the GNN literature (architecture papers, heterophily papers, and OGB leaderboard reports). Rules: (i) a gap is the difference between the table's rank-1 and rank-2 (or rank-5) methods *within one published table*, so training protocol, splits, and tuning are as homogeneous as the original authors made them, and we never compare numbers across papers; (ii) each entry records source table, method names, split protocol, and the gap in points; (iii) datasets enter only if we measure their resolution (14 datasets; actor had no qualifying table). In 1.9% of the 33,600 paired tests the two configurations produce identical correctness vectors on that split and seed, giving a degenerate zero-width interval; these count as non-significant for both methods, which is the conservative reading and leaves the reported shares unchanged. Rule (i) has one exception: from Polynormer's Table 2 we transcribed only the 13 rows it does not share with Platonov et al. (2023)'s Table 4, so its five rank-5 gaps are computed against the rank-5 of that subset and are inflated, and its questions rank-1-vs-2 gap is 0.97 here against 0.06 on the full table. Of these five, the two accuracy gaps and minesweeper's AUC gap fall above their benchmark's MRG; the tolokers and questions AUC gaps fall below, and since the transcription inflates them the un-inflated gaps would fall below as well, so the headline count is unaffected. Median gap: 1.46 points (range 0.01–22.8). For tolokers, minesweeper, and questions the source tables report ROC-AUC (the protocol of Platonov et al. (2023)), so their 24 gaps are scored against the measured AUC MRG of Appendix H rather than against the accuracy-based MRG; 22 of those 24 fall below, as do 53 of the 62 accuracy gaps. The corpus is released with per-gap provenance. It is a convenience sample of the literature, not a systematic survey (coverage per dataset ranges from 2 to 10 gaps), and the pooled 87% should be read next to the per-dataset column of Table 1.

G REPRODUCIBILITY FLOOR: GPU NONDETERMINISM

Thirty configuration cells were independently trained twice at identical configuration, split, and seed, differing only in nondeterministic GPU reduction order. The two runs of the same cell differ by 0.129 points accuracy at the median (max 0.689), while 5.48% of per-node *correctness* values flip (max 7.71%). The caches hold correctness, not predicted labels, so this is a lower bound on the prediction-flip rate: two runs can predict different wrong classes and agree on correctness. These figures are measured by `code/merge_runs.py` as it reconciles a recomputed cell against the cached one, and are reported from that merge; the released caches hold one correctness vector per cell, so this is the one measurement in the paper that a reader cannot recompute without retraining. Two readings: (i) run-level nondeterminism is ≈3.6% of a typical MRG (0.129 of 3.57 points) and cannot explain the effects in this paper; (ii) the aggregate is stable while the prediction set beneath it is not, which is the same pattern this paper measures at the scale of a whole benchmark.

H RESOLUTION ON A RANK METRIC: THE DESIGN EFFECT OF ROC-AUC

(`code/auc_analysis.py`.) Corollary 1 is an identity about a mean of node-level indicators, which is why an accuracy-scale MRG cannot score a table reported on ROC-AUC. The block bootstrap has no such restriction: a replicate is a multiset of test nodes, so any statistic can be recomputed

on it, and $D(T) = \mathrm{Var}(T)/\mathrm{Var}_{i.i.d.}(T)$ with the denominator from the i.i.d. bootstrap needs nothing more. A resample is an integer weight per node, so the AUC is computed with weights rather than by materialising the multiset; the score vector is fixed across replicates, so its sort and tie groups are computed once and a replicate costs one matrix–vector product against the per-block group weights. The implementation is checked against `sklearn` at unit weights and against the per-node path at random integer weights.

This needed the three ROC-AUC benchmarks retrained with their predicted scores cached, which correctness vectors do not contain: an independent re-run of the same grid at 10 splits $\times$ 3 seeds, written to its own cache so that the accuracy results elsewhere keep the exact vectors they were computed from. Paired quantities are taken at dependence radius 4, as in §7.4. The released score caches hold dense ranks rather than probabilities, since the AUC depends on the scores only through their order; the conversion is verified run by run to leave every cached AUC unchanged.

		D(AUC) by model class						MRG (AUC points)	
dataset	m	MLP	GCN	SGC	SAGE	GAT	n_{eff}	i.i.d.	measured
minesweeper	2,500	0.98	0.94	1.03	1.11	0.78	2,596	4.26	4.29
tolokers	2,940	2.9	5.3	5.7	7.0	12.8	512	3.59	4.66
questions	12,231	1.11	1.85	1.08	1.12	2.32	6,683	5.35	5.91

Two readings. First, the paper's central claim survives the change of metric: resolution is indexed by the model class on a rank statistic too, and on tolokers by a factor of 4.4 (2.9 for the graph-free MLP against 12.8 for GAT). Second, the three benchmarks separate. Tolokers carries a real AUC design effect ($D_\Delta = 2.0$), minesweeper essentially none (0.92), and questions a little (1.24); yet all three have AUC-scale MRGs of 4.3–5.9 points, because a rank statistic on a class-imbalanced test set is simply a coarser instrument than an accuracy. Scoring these tables on the accuracy scale would therefore have understated the resolution they need: minesweeper's accuracy MRG is 0.50 points and its AUC MRG 4.3. Of the 24 harvested AUC gaps, 22 fall below the measured resolution; the two that clear it are minesweeper's 4.71 and 6.57.

I NETWORK CROSS-VALIDATION

(`code/ncv.py`.) Following Neville et al. (2012): partition V into k=10 disjoint contiguous subnetworks, using the same capacity-constrained BFS the block bootstrap uses so that both methods see the same notion of a region, and for fold i test on subnetwork i while training and validating on samples drawn from the other nine at the canonical train/val sizes; compare two systems by a paired t-test over the ten fold accuracies. The null is the one of §6.1: the same architecture at two of twelve seeds, i.e. 66 comparisons per (dataset, configuration). The i.i.d. and GPBB columns are the same null on the canonical split, from the cached runs.

		NCV		i.i.d. bootstrap		GPBB	
dataset	config	rej.	se (pp)	rej.	se (pp)	rej.	se (pp)
cora	GCN $L2$	0.000	1.04	0.184	0.69	0.113	0.81
cora	MLP $L2$	0.091	1.08	0.276	1.10	0.258	1.15
citeseer	GCN $L2$	0.061	1.02	0.138	0.87	0.111	0.95
citeseer	MLP $L2$	0.045	1.34	0.158	1.25	0.147	1.29
pubmed	GCN $L2$	0.030	0.40	0.116	0.60	0.087	0.67
pubmed	MLP $L2$	0.000	0.82	0.107	0.87	0.100	0.89
actor	GCN $L2$	0.000	0.57	0.193	1.18	0.187	1.15
actor	MLP $L2$	0.015	0.46	0.040	1.03	0.040	1.02
roman-empire	GCN $L2$	0.015	0.23	0.129	0.38	0.138	0.38
roman-empire	MLP $L2$	0.227	0.10	0.118	0.19	0.113	0.19
amazon-ratings	GCN $L2$	0.091	0.63	0.149	0.50	0.058	0.55
amazon-ratings	MLP $L2$	0.000	0.27	0.082	0.38	0.067	0.40
pooled		0.048		0.141		0.118	

NCV holds at the nominal rate where the within-graph intervals do not, and its standard errors are *smaller* (median 0.60 against 0.85 points). The reason is what it estimates: each fold retrains, so the

quantity tested is an average over ten independently trained pairs, and the training noise the within-graph intervals cannot see is averaged down by $\sqrt{10}$ rather than ignored. The trade is therefore ten retrainings per fold against a calibrated method-level test. One cell is out of line, roman-empire's MLP at 0.227, where NCV's fold standard error is 0.10 points and the architecture is stable enough that real fold-to-fold differences clear it; the procedure is not uniformly conservative.

J VARIANCE COMPONENTS: WHAT ONE TRAINING RUN CANNOT SEE

(`code/variance_components.py`; all 14 datasets, architecture pairs at radius 4, two splits.) MRG here is recomputed over the 44 pairs of those two splits, so it differs by up to 20% from Table 1's figure, which pools all runs. *within* is the paired GPBB standard error of an accuracy difference on a fixed pair of trained models; *train* is the standard deviation of that difference over the ten training seeds at a fixed split; MRG and $\mathrm{MRG}_{\mathrm{method}}$ are Definition 1 applied to the first and to $\sqrt{\text{within}^2 + \text{train}^2}$. The last two columns give the share of same-configuration seed pairs each interval calls significant. Those calls are correct about the two trained models and wrong about the architecture.

	s.d. of the difference			minimum resolvable gap			seed pair called sig.	
dataset	within	train	train share	MRG	$\mathrm{MRG}_{\mathrm{method}}$	ratio	i.i.d.	GPBB
ogbn-arxiv	0.893	0.176	3%	2.50	2.78	1.11	0.49	0.29
tolokers	0.317	0.221	25%	0.89	1.22	1.38	0.10	0.05
cora	1.369	1.268	43%	3.84	5.39	1.41	0.25	0.18
chameleon	3.108	1.831	25%	8.71	10.18	1.17	0.13	0.08
amazon-ratings	0.907	0.423	18%	2.54	2.88	1.13	0.11	0.07
questions	0.047	0.024	27%	0.13	0.18	1.34	0.17	0.06
citeseer	1.831	1.298	43%	5.13	5.92	1.16	0.23	0.17
pubmed	1.573	0.887	21%	4.41	5.28	1.20	0.16	0.08
squirrel	2.158	1.275	29%	6.05	7.11	1.18	0.09	0.06
squirrel-filt.	2.167	1.846	42%	6.07	7.81	1.29	0.07	0.06
chameleon-filt.	3.658	2.614	34%	10.25	12.68	1.24	0.07	0.06
minesweeper	0.200	0.242	38%	0.56	0.99	1.77	0.19	0.20
roman-empire	0.694	0.382	25%	1.94	2.25	1.16	0.22	0.22
actor	1.312	1.066	40%	3.68	4.82	1.31	0.09	0.09

The training term is never negligible and never dominant: 3% of the paired variance on ogbn-arxiv, where the within-graph term is large, but 38–43% on Cora, CiteSeer, minesweeper and squirrel-filtered, where it is small. The two components are thus complementary rather than competing, which is why the datasets on which §7.7 finds GPBB nearly free are also the datasets on which the seed term matters most. Note minesweeper, where $\mathrm{MRG}_{\mathrm{method}}$ is $1.77\times$ the within-graph figure: its errors are almost independent (D=1.18) and its test set is large, so training randomness is essentially all of the uncertainty.

K ESTIMANDS: WHAT ONE GRAPH CAN AND CANNOT IDENTIFY

Under the finite-population estimand a_F, the without-replacement draw of T from the $n-n_{\mathrm{train}}$ non-training nodes contributes a finite-population correction $(1 - m/(n - n_{\mathrm{train}}))$ that *shrinks* variance, partially offsetting D (Wang et al., 2011); both our i.i.d. and GPBB intervals omit it, which is conservative for a_F claims. Under the superpopulation estimand a_S, the one that general claims implicitly make, there is no offset, and in addition a graph-level variance component (the training set and graph are one draw) is invisible to any within-graph method, including ours: from a single graph it is unidentifiable. Our reported D therefore bounds from below the variance inflation relevant to a_S; the split-to-split variability that Shchur et al. (2018) documented is a partial, within-graph glimpse of the same gap. Conformal transductive guarantees (Zargarbashi et al., 2023; Huang et al., 2023) concern coverage of prediction sets under exchangeability and are compatible with everything here: exchangeability licenses those guarantees while leaving the variance of the *mean* inflated by exactly the covariance term this paper measures.

L ASSETS AND LICENSES

Planetoid datasets (Yang et al., 2016), geom-gcn splits (Pei et al., 2020), the heterophilous suite (Platonov et al., 2023) (all public research datasets), and ogbn-arxiv (Hu et al., 2020) (ODC-BY) are used as distributed by their authors. Our code, correctness caches, corpus, and analysis scripts are released under MIT license in the supplementary material.