



SEED VARIANCE AND CONFIDENCE INTERVAL RELIABILITY IN PHYSICS-INFORMED NEURAL NETWORK BENCHMARKS

Physics

Olatinwo Adenike Sola	Department of Physics and Engineering Physics, Obafemi Awolowo University, Ile-Ife, Nigeria
Salako Najeem Abiodun*	Department of Physics and Engineering Physics, Obafemi Awolowo University, Ile-Ife, Nigeria *Corresponding Author
Olubudo Paul Ajibola	Department of Physics and Engineering Physics, Obafemi Awolowo University, Ile-Ife, Nigeria

ABSTRACT

Physics-informed neural networks (PINNs) are often benchmarked from a single training run or from a small number of random seeds, although the resulting test error can vary substantially across training runs. We examine the statistical reliability of such finite-run benchmarks empirically. For three representative PDE configurations with distinct spectral and operator characteristics, we train 250 runs for each of two collocation configurations, together with a crossed design separating weight-initialisation and collocation-sampling variability. The resulting 250-run empirical populations are used as finite reference distributions to evaluate seven mean-interval procedures at $n=3$ to 50 seeds, together with a percentile interval for the median. Errors under an unchanged configuration span more than two orders of magnitude. On the mild Poisson problem, the Student-t interval for the mean raw error covers the empirical reference mean only 80.9–86.9 % of the time, whereas the corresponding interval on log-errors shows near-nominal coverage for $n \geq 5$ across the three configurations studied. We further examine benchmark comparison under both independent and matched-seed sampling. Matched-seed comparisons exploit correlations between paired runs and substantially improve the precision of log-scale comparisons; the paired design reduces the observed log-contrast variance by 52–96% across the three configurations, with near-nominal empirical coverage across the tested seed counts. Selective reporting of the best of k seeds can also produce substantial optimism, while pilot-based calculations indicate that 85–136 runs correspond to a 25% upper multiplicative precision target under the observed log-scale variability. These results motivate a reporting protocol based on explicit randomisation, pilot-based run budgets, log-scale inference, common seeds for comparisons, and complete reporting of the resulting runs. All code and data are released.

KEYWORDS

Physics-Informed Neural Networks, Random Seeds, Confidence Intervals, Coverage Probability

INTRODUCTION

Physics-informed neural networks (PINNs) approximate the solution of a partial differential equation (PDE) by a neural network whose weights are trained to minimise the PDE residual and the boundary/initial-condition mismatch at a finite set of collocation points (Lagaris et al., 1998; Raissi et al., 2019; Karniadakis et al., 2021). Since the formulation of (Raissi et al., 2019), the field has produced a large body of methods—loss re-weighting (Wang et al., 2021; Wang et al., 2022), causal training (Wang et al., 2024), curriculum and sequence-to-sequence training (Krishnapriyan et al., 2021), second-order optimisers (Rathore et al., 2024), alternative architectures (Toscano et al., 2025), and software (Lu et al., 2021)—and a growing number of *benchmarks* designed to compare them (Takamoto et al., 2022; Hao et al., 2024; Grossmann et al., 2024).

A PINN benchmark result is, however, not a deterministic property of a method. Training is a non-convex stochastic optimisation whose outcome depends on the random initialisation of the weights and on the random draw of collocation points; different seeds may converge to solutions of very different accuracy. It is well documented in general machine learning that such seed-induced variance can be as large as the improvements reported by new methods (Henderson et al., 2018; Bouthillier et al., 2019; Bouthillier et al., 2021; Picard, 2021; Madhyastha et al., 2019; Summers et al., 2021), and that ignoring it is a driver of irreproducible claims. Scientific machine learning has been shown to be susceptible to the same reporting problems: a systematic review of machine-learning solvers for fluid-related PDEs found that most claims of superiority over numerical methods relied on weak baselines and that negative results were under-reported (McGreivry et al., 2024); comparisons of PINNs with the finite element method have been more sobering still (Grossmann et al., 2024). The specific question of *how finite numbers of repeated PINN trainings affect the reliability of benchmark uncertainty and comparison* has received less systematic attention—although the importance of quantifying training variation for physics-informed losses has recently been demonstrated in a materials-science setting (Lenau et al., 2026). We have encountered the same problem directly in our own work: two recent physics-informed neural network studies from our group each reported results from only three random seeds, with correspondingly wide confidence intervals on the out-of-bandwidth or seed-to-seed error metrics, and one explicitly flagged the practice as a limitation to be revisited with more seeds (Olatinwo et al., 2026; Olatinwo et al., 2026).

In practice, PINN results are frequently reported as a point estimate from one run, or as a mean $\pm$ standard deviation over a handful of seeds. A confidence interval is a repeated-sampling statement about a specified population functional, not a probability statement about the realised interval containing a “typical” error. The classical Student-t interval for a mean is exact under normal sampling and can be inaccurate for strongly skewed errors (Student,

1908; Hall, 1992). PINN errors are positive and, in the configurations studied here, can span several orders of magnitude with occasional large values.

This paper makes the following contributions.

1. We formalise a PINN benchmark score as an estimator of a functional of the *distribution of test errors induced by the declared stochastic training protocol*, and define the reliability of a reported confidence interval as its frequentist coverage (Neyman, 1937) (Materials and Methods).
2. We generate a large empirical training population—250 runs for each of six problem/configuration pairs, plus a 20×20 crossed design separating initialisation from collocation-sampling variability—on three representative PDE configurations with distinct spectral and operator characteristics, all with code and data released (Materials and Methods).
3. Using these finite empirical reference distributions, we measure the coverage of seven confidence-interval procedures for the mean (Student-t, percentile bootstrap and BCa bootstrap (Efron, 1987; DiCiccio et al., 1996), each on the raw and on the logarithmic error scale, plus the normal-theory z-interval on the raw scale) and a percentile interval for the median as a function of the number of seeds n , and show *how and in which direction* they fail (Section 5.4).
4. We quantify the consequences for benchmark practice: the probability of mis-ranking two methods, the false-positive rate of significance tests, and the optimism induced by reporting the best of k seeds (Section 5.5), and we derive the number of seeds needed for a stated precision (Section 5.7).
5. We distil a compact, evidence-based reporting protocol (Section 6.2).

Background and Related Work

Physics-informed Neural Networks and their Optimisation

Given a PDE $\mathcal{N}[u](x) = f(x)$ in Ω with boundary/initial conditions $\mathcal{B}[u] = g$ on $\partial\Omega$, a PINN parametrises u_θ by a neural network and minimises $\mathcal{L}(\theta) = 1/N_r \sum_i |\mathcal{N}[u_\theta](x_i) - f(x_i)|^2 + \lambda \mathcal{L}_{bc}(\theta)$, where derivatives are obtained by automatic differentiation (Baydin et al., 2018) and the x_i are collocation points (Raissi et al., 2019). The soft-constraint loss is notoriously hard to optimise. Wang et al. identified imbalanced gradients between loss terms (Wang et al., 2021) and, via the neural tangent kernel, a discrepancy in the convergence rates of the loss components (Wang et al., 2022). Krishnapriyan et al. showed that PINNs can fail on problems with convection, reaction or diffusion operators even when the architecture is expressive enough, attributing this to an ill-conditioned loss landscape (Krishnapriyan et al., 2021); Rathore et al. connected the ill-conditioning to the differential operator and showed the benefit of combining first- and second-order optimisers (Rathore et al., 2024). Wang et al. assembled several remedies into a single training pipeline (Wang et al., 2023), and Bayesian formulations quantify predictive uncertainty of a trained PINN under noisy data (Yang et al., 2021). These works concern the optimiser, model or predictive uncertainty of a

trained network. The present study instead treats repeated training itself as the sampling process and asks how finite repeated-training samples affect the inference attached to a reported benchmark number.

Benchmarks and Reporting Practice in Scientific Machine Learning

PDEBench provides time-dependent PDE tasks with code and data for comparing ML models against numerical solvers and ML baselines (Takamoto et al., 2022). PINNacle collects over twenty PDEs and about ten PINN variants and reports extensive comparisons, highlighting loss re-weighting and domain decomposition as important directions (Hao et al., 2024). Grossmann et al. compared PINNs and the finite element method on Poisson, Allen–Cahn and Schrödinger problems and found that PINNs did not outperform the finite element method in accuracy and solution time in their study (Grossmann et al., 2024). McGreivoy and Hakim documented weak baselines and reporting biases in the ML-for-PDE literature (McGreivoy et al., 2024). Our work is complementary: we take the benchmark protocol itself—repeat, average, attach an interval—as the object of study.

Random Seeds and Statistical Reporting in Machine Learning

Henderson et al. showed that non-determinism and intrinsic variance can make deep-RL results hard to interpret and urged significance metrics (Henderson et al., 2018). Bouthillier et al. modelled the whole benchmarking process, showing that data sampling, parameter initialisation and hyper-parameter choice all contribute markedly to variance (Bouthillier et al., 2021). Picard scanned up to 10^4 seeds on CIFAR-10 and found that although the variance is modest, outlier seeds performing much better or worse than

average are easy to find (Picard, 2021). Colas et al. addressed the number of seeds via power analysis (Colas et al., 2018). Agarwal et al. argued for interval estimates of aggregate performance when only few runs are available (Agarwal et al., 2021), and Madhyastha and Jain, and Summers and Dinneen, documented seed- and nondeterminism-induced instability (Madhyastha et al., 2019; Summers et al., 2021). Demšar recommended non-parametric tests for comparing learners (Demšar, 2006). What remains less explored is how finite numbers of repeated PINN trainings affect the calibration of uncertainty intervals and the reliability of benchmark comparisons. The present work treats repeated PINN training as a finite-sample statistical experiment and evaluates the coverage, ranking reliability and run-budget implications of common reporting procedures.

Confidence Intervals and the Bootstrap

A $(1 - \alpha)$ confidence procedure is defined by its coverage probability under repeated sampling (Neyman, 1937); misreading a single realised interval as a probability statement about the parameter is a well-documented fallacy (Morey et al., 2016; Belia et al., 2005; Cumming et al., 2005). The Student- t interval is exact for normal data (Student, 1908); for skewed data its one-sided coverage error is of order $n^{-1/2}$ (Hall, 1992). The bootstrap (Efron, 1979; Efron et al., 1993) offers interval procedures without a normality assumption; the percentile interval is only first-order accurate and *under-covers badly in small samples*, being outperformed by t -intervals when n is small (Hesterberg, 2015; Carpenter et al., 2000), whereas the bias-corrected and accelerated (BCa) interval is second-order accurate (Efron, 1987; DiCiccio et al., 1996). Which of these behaves best for the finite empirical training-error distributions studied here is an empirical question we answer here.

MATERIALS AND METHODS

Problem Formulation

The Training-induced Error Distribution

Fix a PDE problem, a network architecture, a loss, an optimiser and its hyper-parameters; call this a *configuration* C . Under the declared stochastic training protocol, training is a map from a random element $\omega = (\omega_{\text{init}}, \omega_{\text{col}})$ —the initial weights and the collocation points—to trained weights $\hat{\theta}(\omega)$, and hence to a scalar test error

$$e(\omega) = \frac{\|u_{\hat{\theta}(\omega)} - u^*\|_2}{\|u^*\|_2}$$

the relative L_2 error against the reference solution u^* on a fixed test set. A pseudo-random seed s selects one realisation of the declared stochastic protocol. Independent seed-indexed runs are treated as independent draws

$e_1, \dots, e_n \sim F_C$, where F_C denotes the distribution of test errors induced by that protocol. The scientific quantity of interest is a functional $\vartheta(F_C)$; a benchmark report is an estimate of it together with an interval $[L, U]$ computed from $e_1, \dots, e_n$.

Three functionals are natural: the arithmetic mean $\mu = \mathbb{E}e$, the geometric mean $\text{GM} = \exp(\mathbb{E} \ln e)$, and the median. Because errors are positive and vary over orders of magnitude, we will see that GM and the median describe the “typical” run, while μ is dominated by rare bad runs.

Interval Procedures

Let $\bar{e}$ and s be the sample mean and standard deviation, $y_i = \ln e_i$ with mean $\bar{y}$ and standard deviation s_y , and $t_{n-1} = t_{n-1,0.975}$. We study

$$(z) \bar{e} \pm 1.96 \frac{s}{\sqrt{n}}, \quad (t) \bar{e} \pm t_{n-1} \frac{s}{\sqrt{n}}, \quad (t\text{-log}) \exp(\bar{y} \pm t_{n-1} \frac{s_y}{\sqrt{n}}) \quad (2)$$

the percentile bootstrap interval $[\hat{\vartheta}_{(0.025)}^*, \hat{\vartheta}_{(0.975)}^*]$ built from B resampled estimates, and the BCa interval, which replaces the levels 0.025, 0.975 by

$$\alpha_{1,2} = \Phi\left(z_0 + \frac{z_0 + z_\alpha}{1 - \alpha(z_0 + z_\alpha)}\right), \quad \alpha_{1,2} \in \{0.025, 0.975\} \quad (3)$$

with bias correction $\hat{z}_0 = \Phi^{-1}(\#\{\hat{\vartheta}^* < \hat{\vartheta}\}/B)$ and acceleration $\hat{a} = \sum_i (\bar{\vartheta}_{(i)} - \hat{\vartheta}_{(i)})^3 / [6 \sum_i (\bar{\vartheta}_{(i)} - \hat{\vartheta}_{(i)})^2]^{3/2}$ computed from leave-one-out (jackknife) estimates $\hat{\vartheta}_{(i)}$ (Efron, 1987; DiCiccio et al., 1996). Each of the bootstrap variants is applied to the raw and to the log errors; a log-scale interval is exponentiated and therefore targets the geometric mean. A percentile interval for the median is included for completeness.

Coverage, and Why Skewness Matters

For the finite empirical reference distribution $\hat{F}_{250}$, the reported calibration quantity is $\text{cov}_n(\hat{F}_{250}) = \Pr_{\hat{F}_{250}}[L \leq \vartheta(\hat{F}_{250}) \leq U]$. This is conditional empirical calibration, not population coverage under an unknown infinite-sample distribution. For the Student- t interval on n i.i.d. observations with skewness γ , the Edgeworth expansion of the studentised mean gives (Hall, 1992)

$$P\left(\frac{\sqrt{n}(\bar{e} - \mu)}{s} \leq x\right) = \Phi(x) + \frac{\gamma}{6\sqrt{n}}(2x^2 + 1)\phi(x) + O(n^{-1})$$

Hence the probabilities that the reported interval lies *entirely below* or *entirely above* μ are, to first order, $\Phi(-z) + \delta_n$ and $1 - \Phi(z) - \delta_n$ with $z = 1.96$ and $\delta_n = \gamma(2z^2 + 1)\phi(z)/(6\sqrt{n}) \approx 0.085\gamma/\sqrt{n}$. A right-skewed error distribution ($\gamma > 0$) therefore makes intervals for the mean fall *below* the truth more often than above it: the reported error is too optimistic. The two effects cancel in the total miss probability only to first order, so the two-sided coverage error is $O(n^{-1})$ but with a constant that grows quickly with the skewness. If e is log-normal with $\ln e \sim N(m, \sigma^2)$ then $\gamma = (e^{\sigma^2} + 2)\sqrt{e^{\sigma^2} - 1}$, which exceeds 6 for $\sigma \approx 1$, whereas $\gamma = \ln e$ is exactly normal, the t -interval on y has coverage exactly $1 - \alpha$ for $\text{GM} = e^m$ at every n , and the seed-to-seed variability is summarised by the single number σ . Whether real PINN errors are close enough to log-normal for this to matter is settled empirically below.

Comparing Two Configurations

To compare configurations A and B, the target must be specified. With the matched-seed design used here, the primary comparison is the paired log-error difference

$$d_i = \log e_{A,i} - \log e_{B,i}, \quad \exp(\bar{d}) = \frac{\text{GM}_A}{\text{GM}_B}$$

With Welch's test (Welch, 1947) on raw errors the null is equality of means; on log errors it is equality of geometric means (equivalently, the ratio $\text{GM}_A/\text{GM}_B = 1$); the Mann–Whitney test (Mann et al., 1947) tests stochastic dominance. We report (i) paired geometric-mean ranking reliability, (ii) paired log-ratio interval coverage and power, and (iii) an independent-sampling sensitivity analysis. The Mann–Whitney analysis is interpreted through its rank/probability-superiority estimand rather than the geometric-mean estimand.

Seed Budget

For a target upper multiplicative precision factor $1 + \delta$ for the geometric mean (interval $\text{GM} \cdot [(1 + \delta)^{-1}, 1 + \delta]$) with log-scale standard deviation σ , the smallest n satisfying $t_{n-1}\sigma/\sqrt{n} \leq \ln(1 + \delta)$ is approximately

$$n \approx \left(\frac{1.96 \sigma}{\ln(1 + \delta)} \right)^2$$

and, for two-sample detection of a log-ratio Δ with power $1 - \beta$, $n \approx 2(z_{1-\alpha/2} + z_{1-\beta})^2 \sigma^2 / \Delta^2$ per group.

Experimental Design

PDE Problems

We deliberately use small problems whose reference solutions are analytic, so that the error is directly measurable and the planned number of training executions is computationally feasible. The three configurations were chosen to represent distinct spectral and operator characteristics, not a single quantitative difficulty scale.

Poisson-mild (PM). $-u'' = f$ on $(0,1)$, $u(0) = u(1) = 0$, with $u^*(x) = \sin(\pi x) + 1/2 \sin(3\pi x)$.

Poisson-hard (PH). The same equation with $u^*(x) = \sin(2\pi x) + 1/2 \sin(6\pi x)$: higher frequencies, providing a controlled change in spectral content and representation demand.

Convection (CV). $u_t + \beta u_x = 0$, $x \in (0, 2\pi)$, $t \in (0, 1)$, $u(x, 0) = \sin x$, periodic in x , $\beta = 10$, with $u^* = \sin(x - \beta t)$; a canonical failure-prone hyperbolic problem for PINNs (Krishnapriyan et al., 2021).

Network, Loss and Optimiser

All configurations use a fully connected tanh network with three hidden layers of width 32, Glorot-normal initialisation (Glorot et al., 2010), and the mean-squared residual plus boundary/initial loss with unit weight (the Poisson residual is divided by the square of the highest wave number to keep it $O(1)$). Optimisation is full-batch Adam (Kingma et al., 2015) with cosine-decayed learning rate (final value $0.1 \times$ initial) for 6000 iterations (PM, PH; initial rate 10^{-3}) or 3000 iterations (CV; 2×10^{-3}). Configuration A uses $N_r = 128$ (PM, PH) or 400 (CV) random uniform collocation points, drawn once per seed and held fixed; configuration B doubles N_r , as a controlled collocation-budget perturbation. The test error is the relative L_2 norm on 2001 uniform points (PM, PH) or a 64×32 grid (CV). Table 1 summarises the setup. No hyper-parameter was tuned to obtain a particular error distribution.

Table 1. Experimental Setup. All Runs: Tanh MLP 3×32 , Adam, Cosine Decay, Float32. N_r : Collocation Points.

	PM	PH	CV
Iterations	6000	6000	3000
Initial learning rate	10^{-3}	10^{-3}	2×10^{-3}
N_r (config. A / B)	128 / 256	128 / 256	400 / 800
Seeds per configuration	250 (A and B)	250 (A and B)	250 (A and B)
Crossed design (init $\times$ points)	20×20	—	—

Seeds and the Empirical Training Population

Seed $s \in \{0, \dots, 249\}$ determines the initial weights and, through a separate random stream, the collocation points. Configurations A and B use the same initialisation seed, hence the same initial weights, which permits paired comparisons; their collocation point sets are not identical because the collocation budgets differ. Each configuration is trained for all 250 seeds, giving a finite empirical training population $\{e_s\}$. For the crossed design on PM-A, 20 initialisation seeds are crossed with 20 collocation seeds (400 runs) so that the two sources of randomness can be separated by a random-effects analysis of variance on $\log_{10} e$.

Coverage Experiments

We use the empirical distribution $\hat{F}_{250}$ formed by the 250 observed errors as a finite reference distribution, with μ , GM and the median computed from that pool. Coverage estimates are therefore conditional on $\hat{F}_{250}$. For each $n \in \{3, 5, 10, 20, 50\}$ we draw R samples of size n with replacement from the pool-independent draws from F , which avoids the finite-population artefact of sampling without replacement—compute every interval, and record the fraction that contain the target. We use $R = 20,000$ for closed-form intervals and $R = 4000$ with $B = 1999$ resamples for bootstrap intervals; the Monte-Carlo standard error of a coverage estimate is at most 0.4 and 0.8 percentage points respectively, and we report Wilson intervals (Wilson, 1927) in the released

data. The comparison experiments (Section 5.5) use the same machinery with two pools $\{A, B\}$ and $R = 20,000$ pairs of samples.

Software and Reproducibility

All code is written in Python with JAX (Bradbury et al., 2018), NumPy (Harris et al., 2020), SciPy (Virtanen et al., 2020) and Matplotlib (Hunter, 2007). The main batch experiment uses float32 execution and vectorises training over seeds with *vmap*; the diagnostic re-runs use individual training trajectories. The released data record the execution used for each analysis. A seed therefore specifies the stochastic inputs but does not, by itself, guarantee bit-level identity across software, hardware, precision or execution paths. Source code, raw per-seed errors, loss curves and analysis scripts accompany this paper.

RESULTS

The Distribution of PINN Errors

Identical configurations trained from different seeds give errors that differ by orders of magnitude (Figure 1, Table 2). Between the best and the worst of the 250 seeds the error ratio is 293, 461 and 134 for PM, PH and CV, and the ratio of the 95th to the 5th percentile is 26, 66 and 55. The distributions are strongly right-skewed on the raw scale (skewness 4.91, 1.55, 1.33), so the arithmetic mean is $1.5 \times$, $1.7 \times$ and $2.5 \times$ the median: the mean is a statement about the rare bad runs, not about the typical run. On the logarithmic scale the picture is much simpler. For PM the log-errors showed no evidence against normality in the Shapiro-Wilk test (skewness 0.00, Shapiro-Wilk (Shapiro et al., 1965) $p = 0.278$) with a standard deviation of 0.45 decades, i.e. one standard deviation corresponds to a factor 2.8. For PH and CV the Gaussian hypothesis is rejected ($p = 7 \times 10^{-4}$ and $p = 2 \times 10^{-7}$), but the departure is mild in the sense that matters for interval estimation: PH is slightly left-skewed on the log scale (skewness -0.37) and CV is flat-topped (excess kurtosis -1.21), with substantial spread over more than a decade in error (Figure 1). About 20% of PH seeds end with a relative error above 1—worse than predicting zero—so “failure” is an outcome that occurs in one run out of five, and 29% of CV runs are more than three times worse than the median.

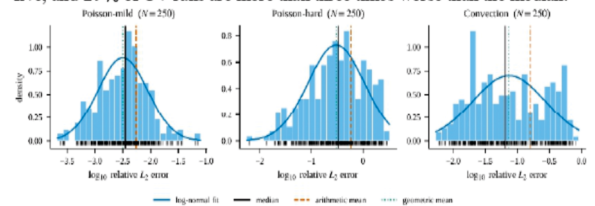


Figure 1. Distribution of the relative L_2 error across training runs (configuration A, 250 seeds per problem) on a $\log_{10}$ axis; ticks at the bottom mark individual seeds. The dashed and dotted vertical lines show that the arithmetic mean is pulled well above the median and geometric mean by rare bad runs.

Table 2. Seed-to-seed distribution of the final relative L_2 error (configuration A, $N = 250$ independent seeds per problem). Skewness γ and Shapiro-Wilk p -values are reported on the raw and $\log_{10}$ scales; ρ_{loss} is the Spearman correlation between final training loss and test error.

Problem	Media n	Geo. mean	Mean	Std. dev.	CV	Q_{95}/Q_{05}	γ_{raw}	$\gamma_{\log}$	SW $p_{\text{raw}} / p_{\log}$
Poisson-mild	3.5 $\times 10^{-3}$	3.2 $\times 10^{-3}$	5.4 $\times 10^{-3}$	7.8 $\times 10^{-3}$	1.4 4	26	4.9 1	0.0 0	10^{-25} / 6 0.278
Poisson-hard	3.3 $\times 10^{-1}$	3.0 $\times 10^{-1}$	5.8 $\times 10^{-1}$	6.3 $\times 10^{-1}$	1.0 8	66	1.5 5	0.3 7	10^{-17} / 7×5 10^{-4}
Convection	6.5 $\times 10^{-2}$	7.4 $\times 10^{-2}$	1.6 $\times 10^{-1}$	1.9 $\times 10^{-1}$	1.1 7	55	1.3 3	0.1 0	10^{-18} / 2×9 10^{-7}

Loss curves, and How Well the Loss Ranks Seeds

Training-loss trajectories of different seeds follow visibly different paths (Figure 2, top): runs that end with a low error typically leave the initial plateau earlier. The Spearman correlation between the final training loss and the test error is 0.66 (PM), 0.75 (PH) and 0.99 (CV): the loss is a useful screen for failed runs—almost perfectly so for convection—but on PM a seed with a lower loss is far from guaranteed to have a lower error. The loss can be evaluated without a reference solution, but it does not remove the need to repeat the experiment.

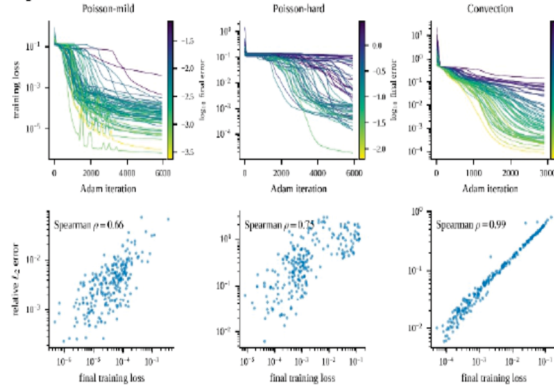


Figure 2. Top: training-loss trajectories of 60 randomly chosen seeds per problem, coloured by their final $\log_{10}$ test error. Bottom: final training loss versus final test error for all 250 seeds, with Spearman rank correlation.

Figure 3 shows what the spread means physically. On PM the best, median and worst seeds differ by a factor of about 300 in error, and only the worst is visibly wrong. On PH the worst seed produces a spurious profile with about $3.5 \times$ the amplitude of the true solution. On CV the median seed shows faint error bands along the characteristics, whereas for the worst seed the error is comparable to the solution amplitude over most of the space–time domain.

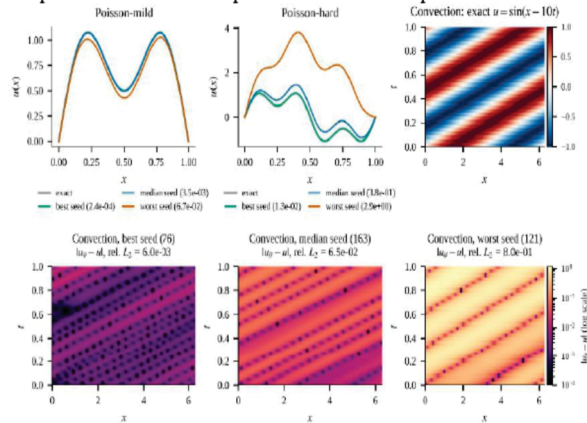


Figure 3. Solutions from the best, median and worst of the 250 seeds (ranked by the error in the pooled batch run). The plotted networks were re-trained individually and the bracketed values are their relative L_2 errors, which can differ from the pooled ones (Section 5.3). Top: Poisson-mild, Poisson-hard, and the exact convection solution. Bottom: pointwise absolute error of the convection solution for the three seeds, on a common logarithmic colour scale (on a linear scale the best and median seeds, whose maximum errors are 0.02 and 0.11, would be indistinguishable from zero next to the worst seed, 1.1).

What “a Seed” Contains

On PM, the crossed design (Table 3) shows that the initialisation seed and the collocation-point seed both matter statistically ($p < 10^{-4}$ for both), but together they explain only $25\% + 7\%$ of the variance of $\log_{10}e$; the remaining 68% is the interaction of the two together with residual chaos of the training trajectory (the design has one run per cell, so these cannot be separated). Practical corollaries: fixing the collocation points and resampling only the initial weights still leaves 93% of the total variance, and fixing the initial weights and resampling only the collocation points leaves 75%. A study that varies only one of the two sources therefore understates the total variance by up to about a quarter, but leaves the other source uncontrolled and undeclared (Bouthillier et al., 2021).

Table 3. Random-effects variance decomposition of $\log_{10}$ error for Poisson-mild, crossing 20 initialisation seeds with 20 collocation-point seeds (400 runs; the interaction is confounded with residual noise).

Source	Variance	Share	p (F-test)
Initialisation seed	0.0415	25.1%	1.3e-19
Collocation-point seed	0.0121	7.3%	1.2e-05
Interaction + residual	0.1117	67.6%	—
Total	0.1653	100%	

A related caveat concerns bit-level reproducibility. Re-training the best, median and worst seeds of each problem *individually*, instead of inside the vectorised batch, reproduced the pooled error to within 4% on PM and below 1% on CV, but by up to a factor of about two (109%) on PH (Figure 3, seeds listed in the released data). The two executions follow different computational paths even though the nominal stochastic inputs are held fixed; the precise source of the discrepancy was not isolated, yet the difference shows that execution details can affect the outcome—in line with reports that nondeterminism alone generates run-to-run variability (Summers et al., 2021). A fixed seed therefore reproduces a benchmark number only when the software, hardware and execution mode are also fixed.

Reliability of Confidence Intervals

Figure 4 and Table 4 give the empirical coverage of nominal 95% intervals as a function of n .

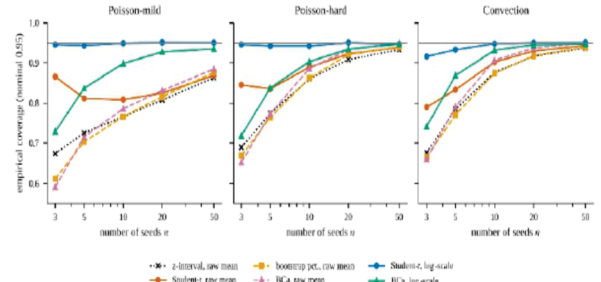


Figure 4. Empirical coverage of nominal 95% confidence intervals versus the number of seeds n . Raw-scale methods target the arithmetic mean; log-scale methods target the geometric mean. Among the intervals shown, only the Student- t interval on log errors is close to nominal for $n \geq 5$.

Table 4. Empirical coverage of nominal 95% intervals as a function of the number of seeds n , shown separately for each problem. Targets: arithmetic mean (raw-scale methods), geometric mean (log-scale methods), median (last row). Monte-Carlo error is at most 0.4 percentage points for the closed-form intervals ($R = 20,000$) and at most 0.8 for bootstrap intervals ($R = 4000$, $B = 1999$). Values more than 3 percentage points below 0.95 are shown in bold.

Poisson-mild

Method	$n = 3$	$n = 5$	$n = 10$	$n = 20$	$n = 50$
z-interval, raw mean	0.674	0.725	0.766	0.808	0.863
Student-t, raw mean	0.866	0.812	0.809	0.825	0.869
Bootstrap pct., raw mean	0.611	0.704	0.766	0.816	0.876
BCa, raw mean	0.591	0.716	0.786	0.831	0.885
Student-t, log-scale (geo. mean)	0.946	0.943	0.949	0.951	0.950
Bootstrap pct., log-scale	0.751	0.836	0.901	0.931	0.936
BCa, log-scale	0.729	0.837	0.898	0.927	0.935
Bootstrap pct., median	0.756	0.933	0.936	0.936	0.949

Poisson-hard

Method	$n = 3$	$n = 5$	$n = 10$	$n = 20$	$n = 50$
z-interval, raw mean	0.690	0.774	0.860	0.908	0.932
Student-t, raw mean	0.845	0.836	0.889	0.923	0.938
Bootstrap pct., raw mean	0.669	0.764	0.862	0.921	0.940
BCa, raw mean	0.652	0.773	0.886	0.933	0.950
Student-t, log-scale (geo. mean)	0.946	0.943	0.942	0.950	0.947

Bootstrap pct., log-scale	0.744	0.833	0.897	0.931	0.944
BCa, log-scale	0.719	0.838	0.903	0.934	0.947
Bootstrap pct., median	0.750	0.939	0.935	0.942	0.946

Convection

Method	$n = 3$	$n = 5$	$n = 10$	$n = 20$	$n = 50$
z-interval, raw mean	0.676	0.785	0.876	0.917	0.937
Student-t, raw mean	0.790	0.833	0.902	0.930	0.943
Bootstrap pct., raw mean	0.666	0.771	0.875	0.917	0.939
BCa, raw mean	0.660	0.791	0.907	0.937	0.950
Student-t, log-scale (geo. mean)	0.916	0.933	0.948	0.950	0.951
Bootstrap pct., log-scale	0.756	0.848	0.910	0.932	0.942
BCa, log-scale	0.742	0.869	0.932	0.945	0.947
Bootstrap pct., median	0.761	0.938	0.935	0.944	0.947

Intervals for the mean of the raw errors do not work.

The Student-t interval on the raw errors—what a “mean $\pm$ confidence interval” in a PINN table usually means—has coverage between 80.9% and 86.9% on PM for every n from 3 to 50; on PH and CV the coverage is as low as 83.6% and 79.0% at small n and approaches 95% only at $n = 50$. The z-interval is worse (67.4–86.3% on PM), as are the raw-scale percentile and BCa bootstrap intervals at small n (61.1% and 59.1% at $n = 3$ on PM). Increasing n helps slowly: with the raw skewness of 4.91 on PM the first-order Edgeworth term of Eq. (4) decays only as $n^{-1/2}$.

The failures are one-sided, and in the optimistic direction (Figure 5, Table 5). At $n = 10$ on PM the raw-scale t -interval lies entirely below the true mean error in 19.1% of the experiments and entirely above it in 0.0%, against a nominal 2.5% for each. This is exactly the sign predicted by Eq. (4) for a right-skewed distribution: a few seeds usually miss the rare bad runs that dominate the mean, so the reported mean error is too small. Moreover, 48% of the $n = 5$ intervals on PM have a negative lower bound, i.e. they contain impossible negative errors.

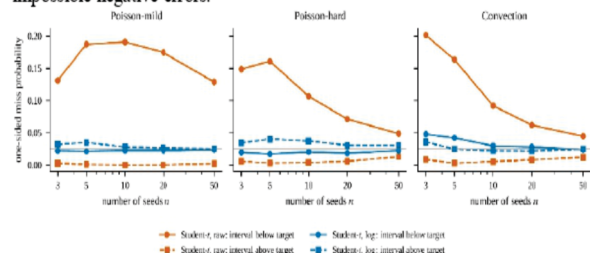


Figure 5. One-sided miss probabilities of the Student-t interval. On the raw scale, intervals fall below the true mean far more often than the nominal 0.025 and almost never above it; on the log scale both tails stay near 0.025.

Table 5. One-sided failure of the Student-t interval on raw errors. “Below”: the reported interval lies entirely under the true mean error (an over-optimistic report); “above”: entirely over it. Nominal value for each is 0.025. The last column is the fraction of intervals whose lower bound is negative (an impossible error).

Problem	n	Below	Above	Total miss	Lower < 0
Poisson-mild	5	0.187	0.001	0.188	0.48
Poisson-mild	10	0.191	0.000	0.191	0.15
Poisson-mild	20	0.175	0.001	0.175	0.02
Poisson-hard	5	0.161	0.003	0.164	0.67
Poisson-hard	10	0.107	0.004	0.111	0.07
Poisson-hard	20	0.071	0.006	0.077	0.00
Convection	5	0.164	0.003	0.167	0.78
Convection	10	0.092	0.006	0.098	0.16

Convection	20	0.062	0.008	0.070	0.00
------------	----	-------	-------	-------	------

Working on the log scale substantially improves interval calibration in the three configurations studied.

The Student-t interval on log-errors, exponentiated to a interval for the geometric mean, has coverage of 94.3–95.1% on PM at all n and 94.2% or higher on PH—consistent with the more regular behaviour of the log-errors in these configurations. On CV, whose log-errors are flat-topped rather than Gaussian, it still reaches 91.6% at $n = 3$ and $\geq 93\%$ from $n = 5$ onwards. The nonparametric log-scale bootstrap intervals are markedly worse at the sample sizes used in practice: percentile and BCa coverage on PM is 83.6% and 83.7% at $n = 5$ and 90.1% and 89.8% at $n = 10$, reaching about 93% only at $n = 50$. This agrees with the general finding that the bootstrap percentile interval is less accurate than the t -interval in small samples (Hesterberg, 2015), and shows that the second-order accuracy of BCa (Efron, 1987; DiCiccio et al., 1996) is asymptotic and does not help at $n \leq 10$. The bootstrap percentile interval for the median is reliable (≈ 93 –95%) from $n = 5$ but has only about 75% coverage at $n = 3$ (75.6% on PM). This is not a bootstrap artefact but arithmetic: for $n = 3$ the percentile interval collapses to the sample range, which contains the population median with probability $1 - 2 \cdot 2^{-3} = 0.75$.

Comparing Methods

Configuration B doubles the collocation budget of A. The pooled geometric-mean error ratio GM_A/GM_B is $1.23 \times$ on PM, $1.41 \times$ on PH, and $1.06 \times$ on CV. These ratios quantify the change induced by the collocation-budget perturbation under the declared training protocol; they are not intended as a general method comparison. Figure 6 and Table 6 show what n seeds per configuration allow one to infer.

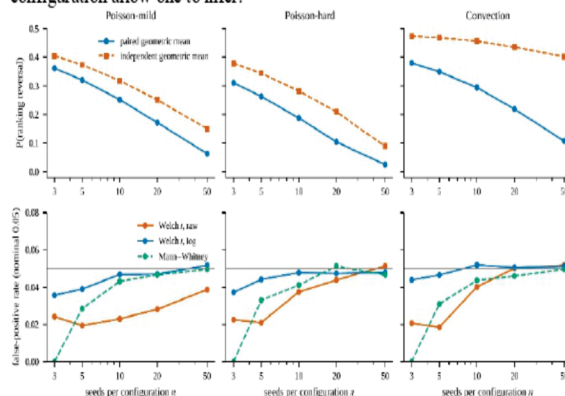


Figure 6. Comparing configurations A and B with n seeds each. Top: probability that the reported geometric-mean winner is the population loser; the paired curve uses the common initialisation seeds and the independent curve is a sensitivity analysis. Bottom: false-positive rate of three tests when two independent replicates of the same configuration are compared.

Table 6. Comparing configuration A (128/128/400 collocation points) with B (twice as many) using n seeds per configuration, under the matched-seed (paired) and independent-sampling designs. GM_A/GM_B is the true pooled geometric-mean effect (> 1 : B is better). Misrank is the probability that the observed ordering contradicts the pooled ordering; paired power is the probability of a significant paired log-t test at $\alpha = 0.05$; paired CI coverage is the empirical coverage of the paired log-ratio interval for $\log(GM_A/GM_B)$; $P(A > B)$ is the Mann-Whitney probability-superiority estimate.

Problem (true effect)	n	Paired misrank	Paired power	Paired CI coverage	Independent misrank	$P(A > B)$
Poisson-mild (1.23 $\times$)	3	0.361	0.036	0.955	0.405	0.564
Poisson-mild (1.23 $\times$)	5	0.320	0.050	0.951	0.373	0.564
Poisson-mild	10	0.252	0.077	0.951	0.318	0.564

(1.23 ×)						
Poisson-mild (1.23 ×)	20	0.172	0.138	0.947	0.251	0.564
Poisson-mild (1.23 ×)	50	0.064	0.312	0.951	0.150	0.564
Poisson-hard (1.41 ×)	3	0.310	0.030	0.960	0.379	0.573
Poisson-hard (1.41 ×)	5	0.263	0.049	0.954	0.344	0.573
Poisson-hard (1.41 ×)	10	0.188	0.105	0.953	0.282	0.573
Poisson-hard (1.41 ×)	20	0.104	0.212	0.950	0.209	0.573
Poisson-hard (1.41 ×)	50	0.026	0.495	0.948	0.090	0.573
Convection (1.06 ×)	3	0.380	0.018	0.969	0.473	0.514
Convection (1.06 ×)	5	0.349	0.028	0.967	0.467	0.514
Convection (1.06 ×)	10	0.294	0.057	0.958	0.456	0.514
Convection (1.06 ×)	20	0.219	0.106	0.954	0.435	0.514
Convection (1.06 ×)	50	0.107	0.219	0.952	0.402	0.514

Matched-seed rankings remain unstable at small n .

Using the common initialisation seeds, the probability that the paired geometric-mean ordering is reversed is about 32.0%, 26.3% and 34.9% on PM, PH and CV at $n = 5$, falling to 25.2%, 18.8% and 29.4% at $n = 10$. Independent-sampling results are retained as a sensitivity analysis; they give different rates because they discard the pairing present in the actual experiment.

Paired inference improves precision, but moderate effects can remain hard to detect. At $n = 50$, the paired log- t power is about 31.2% on PM, 49.5% on PH and 21.9% on CV for the finite empirical reference distributions. These are empirical-reference power estimates, not universal power guarantees. The corresponding independent log-Welch powers are lower in these data (about 17.8%, 26.2% and 4.5%). Under the null calibration using independent draws from the same configuration, the false-positive rates of the tested procedures are generally near their nominal levels at moderate n . The Mann-Whitney analysis addresses probability superiority rather than the geometric-mean estimand; for the three pools, $P(A > B)$ is approximately 0.564, 0.573 and 0.514 for PM, PH and CV, respectively.

Paired log-ratio intervals are well calibrated in these configurations. For $d_i = \log e_{A,i} - \log e_{B,i}$, the interval $\bar{d} \pm t_{n-1,0.975} s_d / \sqrt{n}$ targets $\log(GM_A / GM_B)$. Its empirical coverage at $n = 10$ is 95.1%, 95.3% and 95.8% on PM, PH and CV, respectively, and remains close to 95% across the tested seed counts. The independent Welch interval for the log-ratio is retained as a secondary sensitivity analysis.

The matched-seed design is informative. Seed s gives configurations A and B the same initial weights; the log-errors are correlated across seeds ($\rho = 0.52$ on PM, 0.54 on PH, and 0.96 on CV). The paired analysis therefore uses variation common to both configurations rather than treating the two observed samples as unrelated. The largest correlation occurs for CV, where the matched-seed design provides the strongest variance reduction. Common random numbers provide a direct variance-reduction mechanism for this type of PINN comparison. The effect can be expressed directly in computational terms. Relative to treating the two configurations as independent samples, the paired log-contrast variance is reduced by 52.1%, 54.3% and 96.2% on PM,

PH and CV, corresponding to effective sample multipliers of 2.1 ×, 2.2 × and 26.0 ×. Using a normal-theory calculation at two-sided $\alpha = 0.05$ and 80% power for the observed log-geometric-mean contrasts, this corresponds to about 175 matched pairs on PM, 103 on PH and 269 on CV, compared with 365 independent runs per configuration, 225 and 6998, respectively. These are post-hoc planning values conditional on the observed effects and variances; they are not universal seed requirements.

Table 7. Practical value of matched-seed (common-initialisation) sampling for the observed A/B contrasts. ρ is the correlation between paired log-errors; variance reduction and the effective sample multiplier quantify the variance saved by pairing; paired n and independent n /group are the sample sizes required for 80% power at two-sided $\alpha = 0.05$ under a normal-theory calculation for the observed effect; total-run ratio is the ratio of independent to paired total training runs.

Problem	ρ	Variance reduction	Effective multiplier	Paired n	Independent n /group	Total-run ratio
Poisson-mild	0.523	52.1%	2.1 ×	175	365	2.1 ×
Poisson-hard	0.544	54.3%	2.2 ×	103	225	2.2 ×
Convection	0.962	96.2%	26.0 ×	269	6998	26.0 ×

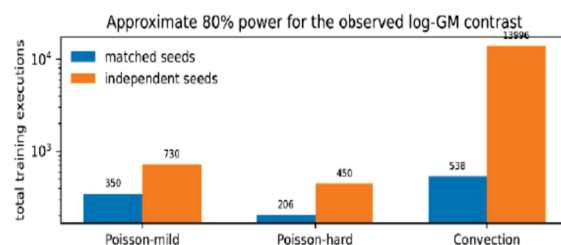


Figure 7. Computational implication of matched seeds for the observed A/B contrasts. Bars show the total number of training executions (both configurations combined) from a normal-theory 80% power calculation for the observed log-geometric-mean effect. The paired design uses common initialisation seeds; the independent design treats the two configurations as unrelated samples. The calculation is conditional on the observed effects and variances.

Selective Seed Reporting

Reporting the best of k seeds for one's own method is the seed-level analogue of a weak baseline (McGreivoy et al., 2024). Figure 8 and Table 8 quantify it.

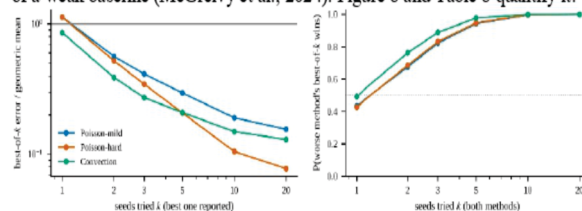


Figure 8. Effect of reporting the best of k seeds. Left: best-of- k error relative to the geometric mean of the same configuration. Right: probability that the worse configuration's best-of- k error beats the better configuration's mean over k seeds.

Table 8. Selective seed reporting. Optimism = median over draws of (best of k seeds)/(geometric mean of the worse configuration); Win = probability that the worse configuration's best-of- k error beats the better configuration's arithmetic mean over k seeds.

Problem	Optimism ($k = 3$)	Win ($k = 3$)	Optimism ($k = 5$)	Win ($k = 5$)	Optimism ($k = 10$)	Win ($k = 10$)
Poisson-mild	0.41	0.82	0.30	0.94	0.19	1.00
Poisson-hard	0.35	0.83	0.21	0.95	0.11	1.00
Convection	0.27	0.89	0.21	0.98	0.15	1.00

The best of $k = 5$ seeds has an error of only $0.30 \times (0.21 \times, 0.21 \times)$ the geometric mean of the same configuration on PM (PH, CV), so a claimed accuracy can be better than the typical accuracy by a factor of three to five. Consequently the worse of the two configurations reports a lower error than the mean of the better one in 94%, 95% and 98% of experiments at $k = 5$ and in essentially all experiments at $k = 10$; with a single seed each the worse method wins in 43–49% of experiments. The asymmetry (best of k against mean of k) is precisely what selective reporting exploits, and it is amplified because the arithmetic mean of a right-skewed sample is itself inflated by outliers.

How Many Seeds are Needed?

Equation (6) with the log-scale standard deviations of Table 2 gives the seed budget of Figure 9. For the observed log-scale variability, an upper multiplicative factor of 1.50 corresponds to about 28, 40 and 43 seeds on PM, PH and CV; an upper multiplicative factor of 1.25 corresponds to 85, 126 and 136; $\pm 10\%$ requires 453, 676 and 732. Conversely, with $n = 5$ seeds the 95% interval for the geometric mean is a multiplicative factor of $3.6 \times$ – $5.1 \times$ wide on either side, and with $n = 10$ still about $2.1 \times$ or more. These counts are pilot-based planning estimates conditional on the observed log-scale variability, not universal seed requirements. For the variability observed here, $n = 3$ – 10 provides only coarse information about the scale of the error and limited resolution for moderate differences between configurations.

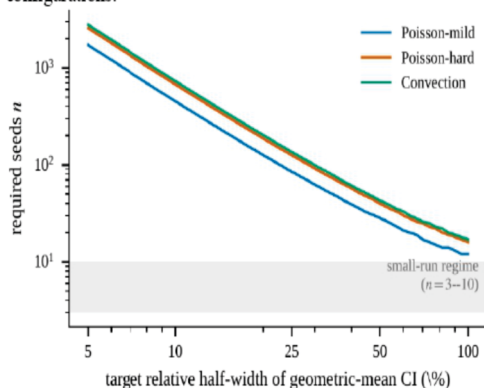


Figure 9. Seeds required so that the 95% t -interval on the log scale has a given relative half-width, from Eq. (6) with the estimated log-scale standard deviations. The shaded band marks the small-run regime $n = 3$ – 10 for visual comparison.

DISCUSSION

What the Results Imply

Three findings stand out. First, on the three configurations studied, the raw test errors are broad and strongly right-skewed, while the log-errors are more regular but not identical in shape across problems. Second, the usual raw-scale Student- t interval for the arithmetic mean under-covers at small and moderate seed counts in these empirical reference distributions, with misses concentrated on the optimistic side; the log-scale Student- t interval is much better calibrated for the geometric mean in the configurations studied. Third, matched-seed comparisons make better use of the experimental design than independent-sample comparisons: the paired log-ratio interval is close to nominal in empirical coverage and the paired test has higher power in these data. Selective best-of- k reporting can still produce large optimism. These findings are calibration results for the declared training protocol, not universal properties of all PINNs.

A Reporting Protocol

Algorithm 1 condenses the evidence into a protocol that costs no more than the extra seeds themselves.

Algorithm 1. Seed-robust Reporting of a PINN Benchmark Result

Require: configuration, pilot size, target relative precision

1. Declare every random source (initial weights, collocation points, data noise), the library versions and the hardware; fix the seed list in advance.
2. Train n_0 seeds; set $y_i = \ln e_i$ and $\hat{\sigma} = \text{sd}(y)$.

3. $n \leftarrow \max\{n_0, \lceil (1.96 \hat{\sigma} / \ln(1 + \delta))^2 \rceil\}$; train the remaining seeds; never discard or replace a run.
4. Report all n errors (strip plot), the median and inter-quartile range, the failure rate, and the geometric mean $e^{\bar{y}}$ with the interval $\exp(\bar{y} \pm t_{n-1} s_y / \sqrt{n})$.
5. If comparing C with C' : use the same seeds for both (common random numbers); with $d_i = \ln e_i - \ln e'_i$ report the ratio $\exp(\bar{d})$ with $\exp(\bar{d} \pm t_{n-1} s_d / \sqrt{n})$ and a paired t -test on d .

Concretely, on the evidence of Secs. Section 5.4–Section 5.7: (i) compute statistics on the log scale and report the geometric mean and median rather than the arithmetic mean; if the arithmetic mean is required, say explicitly that no interval with nominal coverage is available at small n ; (ii) use the Student- t interval on log-errors, which showed near-nominal empirical coverage for the geometric mean across the three configurations studied, and avoid the raw-scale t and z intervals and, for $n \lesssim 20$, percentile and BCa bootstrap intervals; (iii) size the study with Eq. (6) from a pilot rather than by convention, and report the achieved half-width- $n = 3$ – 10 suffices only for order-of-magnitude statements; (applying Eq. (6) to our own three-seed studies (Olatinwo et al., 2026; Olatinwo et al., 2026) indicates that far more than three seeds would be needed for a tight interval, consistent with the wider seed counts those studies flagged as future work); (iv) for comparisons, share initialisation seeds, work on log-errors and report the ratio with its paired interval rather than only a p -value; (v) report the fraction of failed runs (for example $e > 1$ or $e > 3 \times \text{median}$), since a mean over a bimodal outcome conceals it; (vi) never select seeds after seeing results—by test error, or by training loss, which correlates with error but imperfectly (Figure 2); (vii) for many problems, aggregate on the log scale and bootstrap jointly over problems and seeds (Agarwal et al., 2021), and use rank-based tests across problems (Demšar, 2006). Algorithm 1's two-stage rule for n is a heuristic—the pilot estimate $\hat{\sigma}$ is itself uncertain—but the paired log-ratio interval was also evaluated for empirical coverage and was near nominal across the tested seed counts.

Limitations

The study uses three small one- and two-dimensional problems with analytic solutions, one architecture family (3×32 tanh MLPs), and Adam without a second-stage L-BFGS refinement, which is often used in practice and has been argued to change the optimisation landscape faced by PINNs (Rathore et al., 2024); a different optimiser could change both the scale and the shape of the training-induced error distribution. The coverage study is conditional on the finite empirical reference pool of 250 errors, so rare catastrophic events beyond that pool are not represented. The results therefore do not estimate population coverage under an unknown heavier-tailed distribution. Only relative L_2 error is analysed. Hyper-parameter variance (Bouthillier et al., 2021) and data-noise variance are not included, and Configuration B differs from A only in collocation budget. All experiments ran on one CPU platform in float32, and the bit-level reproducibility check (Section 5.3) suggests that results may not transfer exactly across execution modes. Finally, the logarithmic-scale t -interval showed adequate coverage empirically on these configurations and is exact under log-normal sampling, but this does not establish its adequacy for problems with strongly non-normal or multimodal log-errors.

CONCLUSIONS

Treating repeated PINN training as a finite-sample statistical experiment shows that benchmark uncertainty can be poorly calibrated when only a few runs are used. In the three configurations studied, raw-scale intervals for the arithmetic mean under-cover at small and moderate seed counts and tend to miss on the optimistic side, whereas the Student- t interval on log-errors gives near-nominal empirical coverage for the geometric mean. Matched-seed comparisons provide a useful variance-reduction mechanism, and the paired log-ratio interval is near nominal in the tested empirical reference distributions. Selective best-of- k reporting can substantially bias a reported result, while pilot-based calculations provide a way to choose a run budget for a stated multiplicative precision. The resulting protocol is to declare the stochastic sources, use pilot-based budgets, share seeds for comparisons, analyse positive errors on the log scale where appropriate, and report all runs without selective removal. These conclusions are conditional on the three

PDE configurations, architecture, optimiser and execution protocol studied here.

Acknowledgements

The authors acknowledge the Department of Physics and Engineering Physics, Obafemi Awolowo University, Ile-Ife, for providing the academic and computational environment in which this work was carried out. All model training was performed on locally available computing resources (CPU-only), without external funding.

Statement of Competing Interests

The authors declare that no competing interests exist.

List of Abbreviations

PINN, physics-informed neural network; PDE, partial differential equation; CI, confidence interval; BCa, bias-corrected and accelerated (bootstrap interval); SW, Shapiro–Wilk (normality test); GM, geometric mean; PM, Poisson-mild problem; PH, Poisson-hard problem; CV, Convection problem (Materials and Methods) or coefficient of variation (Table 2 only); MLP, multilayer perceptron; FPR, false-positive rate; i.i.d., independent and identically distributed.

Data and Code Availability

All code (PINN training in JAX, statistical analysis, figure and table generation), the raw per-seed errors and loss curves for all 1,909 training executions (6 × 250 main runs, 400 crossed-design runs, and nine single-seed diagnostic re-runs used for Figure 3), and the LaTeX source accompany this paper. Every number in the text is generated by script from these data.

REFERENCES

- Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A., & Bellemare, M. G. (2021). Deep reinforcement learning at the edge of the statistical precipice. *Advances in Neural Information Processing Systems*, 34.
- Baydin, A. G., Pearlmutter, B. A., Radul, A. A., & Siskind, J. M. (2018). Automatic differentiation in machine learning: A survey. *Journal of Machine Learning Research*, 18(153), 1–43.
- Belia, S., Fidler, F., Williams, J., & Cumming, G. (2005). Researchers misunderstand confidence intervals and standard error bars. *Psychological Methods*, 10(4), 389–396.
- Bouthillier, X., Laurent, C., & Vincent, P. (2019). Unreproducible research is reproducible. *Proceedings of the 36th International Conference on Machine Learning*, 97, 725–734.
- Bouthillier, X., Delaunay, P., Bronzi, M., Trofimov, A., Nichyporuk, B., Szeto, J., Sepahvand, N. M., Raff, E., Madan, K., Voleti, V., Kahou, S. E., Michalski, V., Arbel, T., Pal, C., Varoquaux, G., & Vincent, P. (2021). Accounting for variance in machine learning benchmarks. *Proceedings of Machine Learning Systems*, 3, 747–769.
- Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., & Zhang, Q. (2018). JAX: Composable transformations of Python+NumPy programs. GitHub. Retrieved from <https://github.com/google/jax>.
- Carpenter, J., & Bithehl, J. (2000). Bootstrap confidence intervals: When, which, what? A practical guide for medical statisticians. *Statistics in Medicine*, 19(9), 1141–1164.
- Colas, C., Sigaud, O., & Oudeyer, P. (2018). How many random seeds? Statistical power analysis in deep reinforcement learning experiments. *arXiv preprint, arXiv:1806.08295*.
- Cumming, G., & Finch, S. (2005). Inference by eye: Confidence intervals and how to read pictures of data. *American Psychologist*, 60(2), 170–180.
- Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7, 1–30.
- DiCiccio, T. J., & Efron, B. (1996). Bootstrap confidence intervals. *Statistical Science*, 11(3), 189–228.
- Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7(1), 1–26.
- Efron, B. (1987). Better bootstrap confidence intervals. *Journal of the American Statistical Association*, 82(397), 171–185.
- Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap*. Chapman & Hall.
- Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics*, 9, 249–256.
- Grossmann, T. G., Komorowska, U. J., Latz, J., & Schönlieb, C. (2024). Can physics-informed neural networks beat the finite element method?. *IMA Journal of Applied Mathematics*, 89(1), 143–174.
- Hall, P. (1992). *The bootstrap and Edgeworth expansion*. Springer.
- Hao, Z., Yao, J., Su, C., Su, H., Wang, Z., Lu, F., Xia, Z., Zhang, Y., Liu, S., Lu, L., & Zhu, J. (2024). PINNacle: A comprehensive benchmark of physics-informed neural networks for solving PDEs. *Advances in Neural Information Processing Systems*, 37.
- Harris, C. R., Millman, K. J., van der Walt, S. J., et al. (2020). Array programming with NumPy. *Nature*, 585(7825), 357–362.
- Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., & Meger, D. (2018). Deep reinforcement learning that matters. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32, 3207–3214.
- Hesterberg, T. C. (2015). What teachers should know about the bootstrap: Resampling in the undergraduate statistics curriculum. *The American Statistician*, 69(4), 371–386.
- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3), 90–95.
- Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., & Yang, L. (2021). Physics-informed machine learning. *Nature Reviews Physics*, 3(6), 422–440.
- Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. *International Conference on Learning Representations*.
- Krishnapriyan, A. S., Gholami, A., Zhe, S., Kirby, R. M., & Mahoney, M. W. (2021). Characterizing possible failure modes in physics-informed neural networks. *Advances in Neural Information Processing Systems*, 34, 26548–26560.
- Lagaris, I. E., Likas, A., & Fotiadis, D. I. (1998). Artificial neural networks for solving ordinary and partial differential equations. *IEEE Transactions on Neural Networks*, 9(5), 987–1000.
- Lenau, A., Dimiduk, D., & Niezgoda, S. R. (2026). Training variation of physically-informed deep learning models. *Machine Learning: Science and Technology*, 7(3), 035005.
- Lu, L., Meng, X., Mao, Z., & Karniadakis, G. E. (2021). DeepXDE: A deep learning library for solving differential equations. *SIAM Review*, 63(1), 208–228.
- Madhyastha, P., & Jain, R. (2019). On model stability as a function of random seed. *Proceedings of the 23rd Conference on Computational Natural Language Learning*, 929–939.
- Mann, H. B., & Whitney, D. R. (1947). On a test of whether one of two random variables is stochastically larger than the other. *The Annals of Mathematical Statistics*, 18(1), 50–60.
- McGreiv, N., & Hakim, A. (2024). Weak baselines and reporting biases lead to overoptimism in machine learning for fluid-related partial differential equations. *Nature Machine Intelligence*, 6(10), 1256–1269.
- Morey, R. D., Hoekstra, R., Rouder, J. N., Lee, M. D., & Wagenmakers, E. (2016). The fallacy of placing confidence in confidence intervals. *Psychonomic Bulletin & Review*, 23(1), 103–123.
- Neyman, J. (1937). Outline of a theory of statistical estimation based on the classical theory of probability. *Philosophical Transactions of the Royal Society of London. Series A*, 236(767), 333–380.
- Olatinwo, A. S., Salako, N. A., & Olubudo, P. A. (2026). Physics-informed neural networks for open quantum systems: Modelling, noise-aware simulation and residual-based quantum control. *International Journal of Innovative Science and Research Technology*, 11(9), 314–324.
- Olatinwo, A. S., & Salako, N. A. (2026). PINN-KK: A physics-informed neural network for causality-compliant spectroscopic phase retrieval in low-loss regimes. *International Journal of Science and Research*, 15(8), 1305–1314.
- Picard, D. (2021). *torch.manual_seed(3407) is all you need: On the influence of random seeds in deep learning architectures for computer vision*. *arXiv preprint, arXiv:2109.08203*.
- Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378, 686–707.
- Rathore, P., Lei, W., Frangella, Z., Lu, L., & Udell, M. (2024). Challenges in training PINNs: A loss landscape perspective. *Proceedings of the 41st International Conference on Machine Learning*, 235, 42159–42191.
- Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, 52(3–4), 591–611.
- Student (1908). The probable error of a mean. *Biometrika*, 6(1), 1–25.
- Summers, C., & Dinneen, M. J. (2021). Nondeterminism and instability in neural network optimization. *Proceedings of the 38th International Conference on Machine Learning*, 139, 9913–9922.
- Takamoto, M., Praditia, T., Leiteritz, R., MacKinlay, D., Alesiani, F., Pflüger, D., & Niepert, M. (2022). PDEBench: An extensive benchmark for scientific machine learning. *Advances in Neural Information Processing Systems*, 35.
- Tosciano, J. D., Oommen, V., Varghese, A. J., Zou, Z., Daryakenari, N. A., Wu, C., & Karniadakis, G. E. (2025). From PINNs to PIKANs: Recent advances in physics-informed machine learning. *Machine Learning: Science and Engineering*, 1(1), 1–43.
- Virtanen, P., Gommers, R., Oliphant, T. E., et al. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, 17(3), 261–272.
- Wang, S., Teng, Y., & Perdikaris, P. (2021). Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5), A3055–A3081.
- Wang, S., Yu, X., & Perdikaris, P. (2022). When and why PINNs fail to train: A neural tangent kernel perspective. *Journal of Computational Physics*, 449, 110768.
- Wang, S., Sankaran, S., Wang, H., & Perdikaris, P. (2023). An expert's guide to training physics-informed neural networks. *arXiv preprint, arXiv:2308.08468*.
- Wang, S., Sankaran, S., & Perdikaris, P. (2024). Respecting causality for training physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 421, 116813.
- Welch, B. L. (1947). The generalization of “Student's” problem when several different population variances are involved. *Biometrika*, 34(1–2), 28–35.
- Wilson, E. B. (1927). Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158), 209–212.
- Yang, L., Meng, X., & Karniadakis, G. E. (2021). B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data. *Journal of Computational Physics*, 425, 109913.