

When Is Machine Learning Needed for Quantum Memory Budget Prediction? A Baseline Study with Extreme-Scenario Boundary Exploration

David Vesterlund

Industrial Research at Vesterlund Ventures WestQuant Open Source Project.

***Correspondence author**

David Vesterlund,

Industrial Research at Vesterlund Ventures WestQuant Open Source Project.

ORCID : <https://orcid.org/0009-0000-6455-1141>

Submitted: 8 Sept 2026; Published: 5 Oct 2026; Published: 19 Oct 2026

Citation: Vesterlund, D. (2026). When Is Machine Learning Needed for Quantum Memory Budget Prediction? A Baseline Study with Extreme-Scenario Boundary Exploration. *Int J Math Expl & Comp Edu*.3(3):1-6.

Abstract

Findings: On the original quantum-network dataset, the best nonlinear model, Random Forest, reduces mean absolute error (MAE) by only 4.0% relative to Ridge regression with one-hot-encoded categorical features (2.500 versus 2.605), while Ridge reduces MAE by 49.5% relative to a constant predictor (5.160). Nearly all predictive signal is associated with memory family; every individual numerical feature has $|r| < 0.08$ with the target memory budget mc . In four of five controlled synthetic scenarios—threshold effects, saturation, phase transitions, and family-specific nonlinearity—the best tree ensemble reduces MAE by 31–66% relative to Ridge. In the heterogeneous scenario, Random Forest also reduces the observed under-dimensioning frequency from 55.5% to 35.0%. For a two-way numerical interaction, however, the improvement is only 12.5%. These single-split results support Ridge with one-hot encoding as a necessary baseline, but do not establish a universal model-selection threshold.

Method: The original dataset contains 1,000 seed-averaged entanglement-buffering configurations spanning six memory families, nine topology classes, three noise levels, seven traffic levels, and five network sizes ($N \in \{10, 15, 20, 30, 50\}$), with $mc \in [1, 20]$. We compare a constant predictor, single-feature linear regression, multivariable linear regression, Ridge regression, Random Forest, and XGBoost on one fixed 60/20/20 train/validation/test split. MAE is the primary metric, supplemented by tolerance accuracy and the fraction of predictions below the target. To probe where nonlinear models begin to add value, we construct five synthetic datasets of 1,000 observations each with controlled threshold, interaction, saturation, phase-transition, and family-heterogeneous response functions. Because all reported comparisons use one split and seed, differences of a few percentage points are exploratory.

Keywords: Quantum Networks, Quantum Memories, Memory Allocation, Machine Learning, Regression Baselines, Tabular Data, Under-Provisioning

Introduction

Quantum-network design requires decisions about how many quantum memories to allocate to network nodes and how long entanglement should be buffered [1–5]. The required memory budget depends on memory physics, including retrieval efficiency, readout fidelity, and coherence time; on topology and connectivity; and on workload and control assumptions [6–8]. A natural approach is therefore to train a regressor on simulated configurations and use it to predict the minimum sufficient budget mc for new configurations [3,4,9]. Machine learning (ML) is increasingly used in quantum networking for protocol discovery, routing, repeater control, and surrogate-guided optimization; broader connections between ML and quantum technologies are reviewed in Ref [4,10–13]. Yet comparisons among nonlinear models alone do not answer the operational question: does ML add value beyond a simple, well-specified baseline? Across forecasting, clinical prediction, recommender systems, and information retrieval, apparent

gains have often narrowed when stronger classical baselines and leakage controls were introduced [14–19].

We ask:

1. On the original quantum-network dataset, do tree ensembles materially outperform Ridge regression for predicting mc ?
2. Under which controlled forms of nonlinearity do tree ensembles provide a substantial improvement?
3. Which baselines and risk measures should accompany ML results for quantum-network resource prediction?

Our contribution is a baseline-focused reanalysis. We compare nonlinear models with constant, univariate, and regularized additive baselines; construct five synthetic stress tests with known response structure; and report under-dimensioning frequency alongside MAE. The stress tests are boundary probes, not evidence that the same mechanisms occur in deployed quantum networks.

Data and Controlled Scenarios

Original Dataset

The original dataset contains 1,000 seed-averaged instances. Each instance is associated with one of six memory families (AFC, diamond electron, diamond nuclear, neutral atom, superconducting, and trapped ion), one of nine topology classes (binary tree, chain, hierarchical, lattice, random geometric graph, ring, scale-free, star, and Waxman), one of three noise levels, one of seven traffic levels, and $N \in \{10, 15, 20, 30, 50\}$. The memory and topology labels are motivated by established physical platforms and graph families [20–30]. The target m_c is the smallest swept per-node budget that reaches 95% of the best observed request-level availability for that configuration. It ranges from 1 to 20 (mean 12.22, standard deviation 5.89). Family is strongly associated with m_c (Table 1), whereas every individual numerical feature has an absolute Pearson correlation below 0.08. This statement concerns marginal linear association only; it does not exclude nonlinear or conditional dependence.

Synthetic Extreme Scenarios

Each synthetic scenario contains 1,000 independently generated rows, six family labels, nine topology labels, three noise and traffic labels, and $N \in \{4, 5, 6, 7, 8\}$. Numerical inputs are sampled over fixed ranges and topology statistics are assigned by topology class. Gaussian noise is added before rounding and clipping m_c to [1, 20].

Family	n	Mean m_c	SD
AFC	178	16.80	3.32
Diamond electron	154	16.03	2.54
Diamond nuclear	159	7.30	3.80
Neutral atom	168	11.49	3.39
Superconducting	181	16.44	4.45
Trapped ion	160	4.34	2.30

Table 1: Target distribution by memory family in the original dataset

E1: Threshold

A family-shifted step in Π_B , with $m_c \approx 5$ below and $m_c \approx 18$ above the threshold.

E2: Interaction

$m_c = 20(1 - \eta r \alpha / 1.5) + \epsilon$, where η_r is retrieval efficiency and α is algebraic connectivity.

E3: Saturation

$m_c = 20(1 - e^{-\Pi_B/\tau}) + \epsilon$, with family-dependent τ .

E4: Phase Transition

A discontinuous change from approximately 3 to approximately 17 when $\eta r F_{r_0} > 0.5$ and $\Pi_B > 4$, with increased variance near the boundary.

E5: Heterogeneous Nonlinearity

Each family follows a distinct response to Π_B (exponential saturation, sigmoid, power law, rational saturation, sinusoid, or hyperbolic tangent).

These scenarios are intentionally favorable to flexible nonlinear models. Their purpose is to demonstrate model behavior under known structure, not to estimate prevalence or effect sizes in real networks.

Methods

Models and Features

We evaluate:

1. a training-set mean predictor;
2. a univariate ordinary least-squares model;
3. ordinary least squares using all numerical features;
4. Ridge regression with penalty $\alpha = 1$ and one-hot-encoded categorical features;
5. Random Forest with 300 trees and maximum depth 15; and
6. XGBoost with 500 trees, maximum depth 6, learning rate 0.05, subsampling 0.8, and column subsampling 0.8 [31–34].

General background is given in Ref. [35]. The feature set comprises 15 numerical variables and four categorical variables (family, topology, noise, and traffic). All predictions are rounded to integer budgets and clipped to [1, 20]. A permutation generated with seed 42 defines 60% training, 20% validation, and 20% test partitions. The current scripts do not use the validation partition for model selection; thus the effective reported comparison is a fixed train/test evaluation with 600 training and 200 test observations.

Metrics

The primary metric is mean absolute error (MAE) [36]. We also report the fractions within one and two memory units of the target. The empirical under-dimensioning frequency is R_{under} ,

$$R_{\text{under}} = \frac{1}{n} \sum_{i=1}^n \mathbf{1}(\hat{m}_{c,i} < m_{c,i}),$$

and mean under-prediction is computed conditionally on $m_c < m_c$. R_{under} is a descriptive frequency, not a calibrated probability of operational failure: the target itself is simulation-b derived, and no asymmetric resource-cost function is fitted.

We use 10% and 20% as exploratory labels (small, marginal, and substantial), not as inferential cutoffs.

$$\Delta(A; B) = 100 \frac{\text{MAE}(B) - \text{MAE}(A)}{\text{MAE}(B)}.$$

We use 10% and 20% as exploratory labels (small, marginal, and substantial), not as inferential cutoffs.

Results

Original Dataset

Method	MAE	Within 1	Within 2	Under-dim.
Constant mean	5.160	0.120	0.235	58.5%
Univariate (IIB)	5.140	0.120	0.235	58.5%
Linear (numerical)	5.255	0.100	0.220	58.5%
Ridge (all)	2.605	0.305	0.555	49.5%
Random Forest	2.500	0.400	0.525	49.0%
XGBoost	2.510	0.400	0.615	49.0%

Table 2: Fixed-Split test Performance on the Original Dataset (ntest = 200)

Extreme-Scenario Comparison

Scenario	Constant	Ridge	RF	XGB	Best ML vs Ridge
Original	5.160	2.605	2.500	2.510	4.0%
E1 threshold	6.120	3.210	2.210	2.220	31.2%
E2 interaction	3.485	1.435	1.255	1.310	12.5%
E3 saturation	3.260	1.775	0.795	0.770	56.6%
E4 phase transition	4.990	3.720	1.560	1.900	58.1%
E5 heterogeneous	5.020	2.940	0.990	1.040	66.3%

Table 3: MAE by scenario. “Best ML” is the lower MAE of Random Forest and XGBoost

The tree ensembles provide large reductions for E1, E3, E4, and E5. Axis-aligned partitions can represent the E1 and E4 discontinuities directly, while the flexible ensembles capture saturation and family-specific shapes in E3 and E5. E2 produces only a 12.5% gain. One-hot encoding cannot encode the numerical product $\eta\alpha$; rather, over the sampled and partly discrete range of α , additive main effects approximate enough of the bilinear surface that Ridge remains competitive. Explicit interaction terms would be the appropriate linear-family baseline for E2.

Under-Dimensioning Frequency

Scenario	Method	Under-dim.	Mean under-prediction
Original	Constant	58.5%	5.01
Original	Ridge	49.5%	2.85
Original	Random Forest	49.0%	2.86
E5	Constant	62.0%	4.73
E5	Ridge	55.5%	3.25
E5	Random Forest	35.0%	1.51

Table 4: Selected under-dimensioning results

On the original data, Ridge and Random Forest have nearly identical under-dimensioning frequencies. In E5, Random Forest reduces the observed frequency from 55.5% for Ridge to 35.0%, and also reduces the conditional magnitude of under-predictions. The 62.0% constant model value is not the relevant model-to-model baseline. Frequencies near one half are unsurprising for approximately centered predictions trained with symmetric objectives; controlling under-provisioning

Ridge reduces MAE by 49.5% relative to the constant predictor. Random Forest reduces MAE by a further 4.0% relative to Ridge. The nonlinear model therefore adds little on this particular split, while one-hot family information accounts for the dominant improvement. The conclusion is metric-dependent: XGBoost has the best within-two rate even though Random Forest has the lowest MAE. Tree ensembles are generally strong on medium-sized tabular datasets, making the small MAE difference here informative, but not statistically conclusive [37,38].

requires an explicitly asymmetric objective or post-hoc safety margin.

Discussion

When Nonlinear ML adds Little

In the original data, the target is strongly stratified by memory family and only weakly associated, marginally and linearly, with individual numerical variables. An additive model with category-specific intercepts therefore captures much of the available signal. Random Forest lowers MAE by only 0.105 memory units relative to Ridge on the fixed test set. This resembles findings in other applied domains where flexible models did not reliably improve upon strong regression baselines when the effective signal was low-dimensional or close to additive [14–17]. This result does not imply that quantum-network memory budgeting is intrinsically linear. Family labels may proxy fixed bundles of physical parameters, the simulator design may suppress numerical variation, and the random split tests interpolation among observed families rather than extrapolation to a new memory technology.

When Nonlinear ML adds Value

The controlled scenarios identify response structures for which tree ensembles can be useful: discontinuous thresholds, saturating curves, multidimensional transition surfaces, and category-specific nonlinear functions. In these constructed cases, the best ensemble reduces MAE by 31–66% relative to additive Ridge. A pure bilinear interaction produces a smaller gain because an additive approximation performs reasonably over the sampled domain. Before invoking a black-box model, investigators should therefore compare against intermediate baselines such as polynomial or interaction-augmented Ridge and generalized additive models.

Practical baseline protocol

We recommend a model ladder rather than an ML-versus-ML comparison:

1. constant mean or median;
2. a transparent domain scalar or univariate regression;
3. multivariable linear regression;
4. Ridge with categorical encoding;
5. Ridge with prespecified interactions or a generalized additive model;
6. nonlinear ensembles.

Model choice should use repeated resampling, paired uncertainty intervals, and domain costs. Interpretability is itself relevant when comparable performance can be achieved by a simpler model [39]. The 10% and 20% bands used here are descriptive heuristics and should not replace uncertainty analysis.

Limitations

1. The extreme scenarios are synthetic and deliberately designed to exhibit nonlinear structure. They establish capability, not external validity.
2. All headline results use one split and one seed. A 4.0% gap may be smaller than split-to-split variation; repeated splits, paired tests, and bootstrap intervals are needed [40, 46].
3. The sample size is 1,000. Flexible models may require more data than regression models for stable estimation [41].
4. Only Random Forest and XGBoost represent nonlinear ML. Gaussian processes, neural networks, generalized additive models, and explicit interaction models may yield different comparisons [42,43].
5. The empirical under-dimensioning frequency is not linked to a resource-cost function. Quantile regression or another asymmetric objective would target under-provisioning directly [44].
6. Pearson correlation measures marginal linear association and can miss nonlinear and nonmonotonic dependence [45].
7. The scripts reserve a validation split but do not use it. Moreover, the reported “best single feature” is selected by test MAE, which makes that auxiliary univariate baseline optimistically biased. The central Ridge–ensemble comparison does not use this feature-selection step, but future code should select features and hyperparameters using training/validation data only.
8. Synthetic and original datasets differ in network-size and traffic supports. Cross-scenario MAEs characterize separate generated problems and are not estimates of transfer performance.
9. The target is simulator-derived. Simulator assumptions can change absolute predictions; cross-validation studies show that agreement requires reconciliation of connection and error models [47].

Relationship to prior work

Prior quantum-network studies address resource allocation through simulation, optimization, learned surrogates, and queueing analysis. SeQUeNCe supports network-level simulation and resource-management studies; Prielinger et al. use support-vector and Random Forest surrogates in quantum-network optimization; Davies et al. analyze entanglement buffering with two quantum memories; and Vardoyan et al. study an entanglement distribution switch [3-5,8]. The present study asks a narrower methodological question: whether nonlinear regressors improve on an encoded linear baseline for scalar memory-budget prediction. Cross-simulator work further cautions that matching labels do not guarantee matching semantics: simulators can agree with their respective analytical descriptions while differing because their connection and error models differ [47]. Accordingly, baseline superiority within one generated dataset should not be interpreted as evidence of hardware-level validity. The appropriate claim is conditional: given these data and this split, an encoded additive baseline captures nearly all of the ensemble’s MAE improvement.

Conclusion

On the original dataset, Ridge with one-hot encoding reduces MAE from 5.160 to 2.605, while Random Forest reaches 2.500—a further reduction of only 4.0%. In four controlled nonlinear scenarios, by contrast, the best tree ensemble reduces MAE by 31–66% relative to Ridge. The key methodological lesson is not that ML is unnecessary, but that its incremental value depends on response structure and must be measured against a strong additive baseline. The reported 4.0% difference is not yet a robust estimate because it rests on one split. A submission-ready confirmatory study should repeat the full pipeline over multiple splits, report paired uncertainty, add interaction and generalized-additive baselines, and evaluate an asymmetric loss tied to the cost of under-provisioning.

Reproducibility

Code, generated scenarios, fixed-split outputs, and manuscript artifacts are contained in the project repository at <https://github.com/VesterlundCoder/quantum-network>. The principal scripts are `run_baselines.py`, `generate_extreme_scenarios.py`, and `run_extreme_comparison.py`. Machine-readable results are stored under `code/results/`.

Declaration of Competing Interests

The author declares no competing interests. This research received no external funding. The author has no financial or personal relationships with the organizations or products discussed that could inappropriately influence this work.

AI Use Disclosure

Generative AI tools were used for literature-search assistance, prose editing, reference formatting, and LaTeX preparation. The author defined the research questions and methods, produced the computational results, verified the scientific claims, and accepts responsibility for the manuscript.

References

- Wehner, S., Elkouss, D., & Hanson, R. (2018). Quantum internet: A vision for the road ahead. *Science*, 362, eaam9288. <https://doi.org/10.1126/science.aam9288>
- Azuma, K., et al. (2023). Quantum repeaters: From quantum networks to the quantum internet. *Reviews of Modern Physics*, 95, 045006. <https://doi.org/10.1103/RevModPhys.95.045006>
- Wu, X., et al. (2021). SeQUeNCe: A customizable discrete-event simulator of quantum networks. *Quantum Science and Technology*, 6, 045027. <https://doi.org/10.1088/2058-9565/ac22f6>
- Prielinger, L., Iñesta, A. G., & Vardoyan, G. (2025). Surrogate-guided optimization in quantum networks. *npj Quantum Information*, 11, Article 89. <https://doi.org/10.1038/s41534-025-01048-3>
- Davies, B., Iñesta, A. G., & Wehner, S. (2024). Entanglement buffering with two quantum memories. *Quantum*, 8, 1458. <https://doi.org/10.22331/q-2024-09-03-1458>
- Lvovsky, A. I., Sanders, B. C., & Tittel, W. (2009). Optical quantum memory. *Nature Photonics*, 3, 706–714. <https://doi.org/10.1038/nphoton.2009.231>
- Heshami, K., et al. (2016). Quantum memories: Emerging applications and recent advances. *Journal of Modern Optics*, 63, 2005–2028. <https://doi.org/10.1080/09500340.2016.1148212>
- Vardoyan, G., Guha, S., Nain, P., & Towsley, D. (2021). On the stochastic analysis of a quantum entanglement distribution switch. *IEEE Transactions on Quantum Engineering*, 2, 1–16. <https://doi.org/10.1109/TQE.2021.3058058>
- Coopmans, T., et al. (2021). NetSquid, a NETWORK simulator for QUantum Information using Discrete events. *Communications Physics*, 4, Article 164. <https://doi.org/10.1038/s42005-021-00647-8>
- Wallnöfer, J., Melnikov, A. A., Dür, W., & Briegel, H. J. (2020). Machine learning for long-distance quantum communication. *PRX Quantum*, 1, 010301. <https://doi.org/10.1103/PRXQuantum.1.010301>
- Le, L., & Nguyen, T. N. (2022). DQRA: Deep quantum routing agent for entanglement routing in quantum networks. *IEEE Transactions on Quantum Engineering*, 3, 1–12. <https://doi.org/10.1109/TQE.2022.3148667>
- Haldar, S., Barge, P. J., Khatri, S., & Lee, H. (2024). Fast and reliable entanglement distribution with quantum repeaters: Principles for improving protocols using reinforcement learning. *Physical Review Applied*, 21, 024041. <https://doi.org/10.1103/PhysRevApplied.21.024041>
- Krenn, M., Landgraf, J., Foesel, T., & Marquardt, F. (2023). Artificial intelligence and machine learning for quantum technologies. *Physical Review A*, 107, 010101. <https://doi.org/10.1103/PhysRevA.107.010101>
- Hand, D. J. (2006). Classifier technology and the illusion of progress. *Statistical Science*, 21, 1–14. <https://doi.org/10.1214/0883423060000000060>
- Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2018). Statistical and machine learning forecasting methods: Concerns and ways forward. *PLOS ONE*, 13, e0194889. <https://doi.org/10.1371/journal.pone.0194889>
- Christodoulou, E., et al. (2019). A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models. *Journal of Clinical Epidemiology*, 110, 12–22. <https://doi.org/10.1016/j.jclinepi.2019.02.004>
- Ferrari Dacrema, M., Cremonesi, P., & Jannach, D. (2019). Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems* (pp. 101–109). <https://doi.org/10.1145/3298689.3347058>
- Lin, J. (2019). The neural hype and comparisons against weak baselines. *ACM SIGIR Forum*, 52, 40–51. <https://doi.org/10.1145/3308774.3308781>
- Kapoor, S., & Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, 4, 100804. <https://doi.org/10.1016/j.patter.2023.100804>
- Afzelius, M., Simon, C., de Riedmatten, H., & Gisin, N. (2009). Multimode quantum memory based on atomic frequency combs. *Physical Review A*, 79, 052329. <https://doi.org/10.1103/PhysRevA.79.052329>
- Pompili, M., et al. (2021). Realization of a multinode quantum network of remote solid-state qubits. *Science*, 372, 259–264. <https://doi.org/10.1126/science.abg1919>
- Bradley, C. E., et al. (2019). A ten-qubit solid-state spin register with quantum memory up to one minute. *Physical Review X*, 9, 031045. <https://doi.org/10.1103/PhysRevX.9.031045>
- Ritter, S., et al. (2012). An elementary quantum network of single atoms in optical cavities. *Nature*, 484, 195–200. <https://doi.org/10.1038/nature11023>
- van Leent, T., et al. (2022). Entangling single atoms over 33 km telecom fibre. *Nature*, 607, 69–73. <https://doi.org/10.1038/s41586-022-04764-4>
- Magnard, P., et al. (2020). Microwave quantum link between superconducting circuits housed in spatially separated cryogenic systems. *Physical Review Letters*, 125, 260502. <https://doi.org/10.1103/PhysRevLett.125.260502>
- Krut'yanskiy, V., et al. (2023). Entanglement of trapped-ion qubits separated by 230 meters. *Physical Review Letters*, 130, 050803. <https://doi.org/10.1103/PhysRevLett.130.050803>
- Penrose, M. (2003). *Random geometric graphs*. Oxford University Press. <https://doi.org/10.1093/acprof:oso/9780198506263.001.0001>
- Barabási, A.-L., & Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286, 509–512. <https://doi.org/10.1126/science.286.5439.509>
- Waxman, B. M. (1988). Routing of multipoint connections. *IEEE Journal on Selected Areas in Communications*, 6, 1617–1622. <https://doi.org/10.1109/49.12889>
- Fiedler, M. (1973). Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23, 298–305. <https://doi.org/10.21136/CMJ.1973.101168>
- Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12, 55–67. <https://doi.org/10.1080/00401706.1970.10488634>

32. Breiman, L. (2001). Random forests. *Machine Learning*, 45, 5–32. <https://doi.org/10.1023/A:1010933404324>
33. Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29, 1189–1232. <https://doi.org/10.1214/aos/1013203451>
34. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). <https://doi.org/10.1145/2939672.2939785>
35. Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer. <https://doi.org/10.1007/978-0-387-84858-7>
36. Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error over the root mean square error in assessing average model performance. *Climate Research*, 30, 79–82. <https://doi.org/10.3354/cr030079>
37. Grinsztajn, L., Oyallon, E., & Varoquaux, G. (2022). Why do tree-based models still outperform deep learning on typical tabular data? In *Advances in Neural Information Processing Systems*, 35.
38. Shwartz-Ziv, R., & Armon, A. (2022). Tabular data: Deep learning is not all you need. *Information Fusion*, 81, 84–90. <https://doi.org/10.1016/j.inffus.2021.11.011>
39. Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1, 206–215. <https://doi.org/10.1038/s42256-019-0048-x>
40. Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap*. Chapman & Hall. <https://doi.org/10.1201/9780429246593>
41. van der Ploeg, T., Austin, P. C., & Steyerberg, E. W. (2014). Modern modelling techniques are data hungry. *BMC Medical Research Methodology*, 14, Article 137. <https://doi.org/10.1186/1471-2288-14-137>
42. Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press.
43. Hastie, T. J., & Tibshirani, R. J. (1990). *Generalized additive models*. Chapman & Hall. <https://doi.org/10.1201/9780203753781>
44. Koenker, R., & Bassett, G. (1978). Regression quantiles. *Econometrica*, 46, 33–50. <https://doi.org/10.2307/1913643>
45. Reshef, D. N., et al. (2011). Detecting novel associations in large data sets. *Science*, 334, 1518–1524. <https://doi.org/10.1126/science.1205438>
46. Bouthillier, X., et al. (2021). Accounting for variance in machine learning benchmarks. *Proceedings of Machine Learning Systems*, 3.
47. Chung, J., et al. (2025). Cross-validating quantum network simulators. *arXiv*. <https://doi.org/10.48550/arXiv.2504.01290>

Copyright: ©2026. David Vesterlund. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.