Computation and Performance Studies
Volume 1 · Number 1 · 2026

Choosing a Learning Window by Training Return: Small Evaluation Gains in Released Procgen Runs

Joseph Wright[1] · Blake Stone[2] · Jason Reed[3]

[1] Cyprus · [2] Germany · [3] Canada

Original publication: October 28, 2026 | Article 644

Abstract

Choosing the best-looking training window need not choose the best behavior outside the training levels. We divided each of forty-eight released Procgen training curves into thirty candidate windows and selected using training returns alone. Relative to a fixed terminal candidate, the selected windows gained 0.01235 normalized training units and 0.00651 unrestricted-evaluation units, averaged equally across sixteen games. The evaluation gain's pointwise working interval barely excluded zero, at 0.000067 to 0.012951. Training-based selection therefore showed a small positive association with broader performance in these records, with considerably less evidence for a reliable practical advantage than the training maximum itself suggests.

Retrospective selection analysis using public rolling-return logs. A selected window contains an evolving policy history and is not a saved model checkpoint.

Choosing a Learning Window by Training Return: Small Evaluation Gains in Released Procgen Runs

> Retrospective selection analysis using public rolling-return logs. A selected window contains an evolving policy history and is not a saved model checkpoint.

Abstract

Choosing the best-looking training window need not choose the best behavior outside the training levels. We divided each of forty-eight released Procgen training curves into thirty candidate windows and selected using training returns alone. Relative to a fixed terminal candidate, the selected windows gained 0.01235 normalized training units and 0.00651 unrestricted-evaluation units, averaged equally across sixteen games. The evaluation gain's pointwise working interval barely excluded zero, at 0.000067 to 0.012951. Training-based selection therefore showed a small positive association with broader performance in these records, with considerably less evidence for a reliable practical advantage than the training maximum itself suggests.

The selection decision comes before its interpretation

Suppose a researcher has a long reward trace and must decide which part best represents the learned behavior. Taking the final segment is simple. Taking the segment with the highest training reward appears more attentive to the observed learning dynamics. The latter option also guarantees that the reported training statistic cannot be worse than the terminal candidate when that candidate belongs to the search menu. That guarantee says nothing by itself about behavior under a broader environment distribution.

Our research question asks whether selecting a return window through its training score also improves the corresponding unrestricted-level evaluation score relative to a fixed terminal window. It is deliberately a question about a reporting rule, not an evaluation of an early-stopping algorithm. The complete training histories are available to the selector retrospectively, and the source archive contains return logs rather than recoverable policy checkpoints for every candidate.

The mathematical setting of rewards, policies, and environment interaction comes from Sutton and Barto, 2018. The underlying source learner uses proximal policy optimization, described in Schulman et al., 2017. We do not alter that learner. Instead, we hold its recorded learning trajectory fixed and change the rule used to choose the portion of the trajectory to summarize. This isolates a selection operation on observed records while preserving the limits of retrospective evidence.

There is a natural connection to interactive environments. Meng et al., 2026 concerns interactive game rendering, where an appealing demonstration may be chosen from a much longer interaction history. Our study does not assess that rendering architecture or its visual quality. It addresses an adjacent reporting issue: how choosing a favorable slice by one observable score relates to performance on a different, broader evaluation stream.

Why an in-sample maximum needs a second outcome

The optimizer's curse in Smith and Winkler, 2006 explains why choosing the largest noisy estimate can create an optimistic impression of the selected option. The present setting adds temporal dependence: neighboring windows come from the same evolving policy, and the logged episode buffers can overlap. Both genuine progress and temporary favorable outcomes can influence which training window looks best.

Model-selection concerns in Varma and Simon, 2006 and leakage analysis in Kapoor and Narayanan, 2023 motivate protecting the evaluation stream from the selection rule. We enforce that boundary directly. The selector receives only the thirty training-window means. Corresponding evaluation means are read after the index is fixed. The numerical findings remain exploratory because the analyst designed the secondary analysis after the source logs were released; protecting the selection input is not a substitute for preregistration.

The practical proposition is that a training-selected window can have a positive evaluation contrast, but its evaluation advantage need not equal the guaranteed training advantage. This proposition does not assume independent noise across windows or an unchanging policy. It asks whether the available selection signal usefully carries beyond the finite levels on which that signal was accumulated.

Selective classification in Geifman and El-Yaniv, 2017 studies choosing which predictions to accept under a coverage-risk tradeoff. Our rule chooses a learning period rather than abstaining on individual inputs, so it does not inherit that method's risk interpretation. We define a reproducible temporal selector and summarize its observed evaluation contrast across supplied games. The interval is not an uncertainty bound for the largest training score or the unseen value of an optimal policy.

A fixed candidate grid built from public curves

Cobbe et al., 2019 introduced the procedural benchmark underlying the source records. The data used here are the easy-mode fixed-two-hundred-level logs in OpenAI, 2026, acquired from a pinned archived commit. There are sixteen games and three run labels per game. Separate training and evaluation files each contain 3,051 update rows. File hashes and the source tree identify the exact snapshot used by the analysis.

The source evaluation worker samples with an unrestricted level parameter and contributes no gradient weight to PPO. The training worker samples the restricted level set. Level identifiers and individual episode outcomes are not supplied in the acquired CSVs. We therefore cannot verify literal nonoverlap of realized training and evaluation episodes or recompute a standard deviation over independent evaluation episodes.

We omit the first ten update records before creating candidates, consistent with the source plotting concern that early episode buffers may be incompletely populated. The next three thousand rows form thirty contiguous, nonoverlapping windows of one hundred updates. The remaining forty-one rows are unused in this comparison. This creates an equal-width menu with a clearly defined terminal candidate instead of giving the final incomplete segment a different amount of averaging.

The terminal candidate ends at 49,315,840 source training steps. The largest training-window mean chooses an index between one and thirty; an exact tie would choose the earliest index. Every score is normalized using the source easy-game range for its game, without clipping. Because this transformation is increasing and identical within a game, it cannot change which window the training selector chooses.

For the selected index, we retrieve the evaluation mean over exactly the same update rows. We also retrieve the fixed thirtieth window's evaluation mean. This alignment matters: pairing a training peak with an evaluation maximum elsewhere would create a different, evaluation-informed rule. The procedure never searches the evaluation curve to repair an unfavorable selection outcome.

What a selected window actually represents

Each logged return mean is based on an episode buffer with source capacity one hundred. The policy changes as learning proceeds, and the hundred-update average combines these changing policies and partially overlapping episode histories. A selected window is consequently a period in the source learning trace, not one frozen agent's return under a separately administered test. It cannot be deployed simply by reading its midpoint from a CSV.

Frameworks such as Raffin et al., 2021 and Huang et al., 2021 emphasize explicit implementation and evaluation conventions. Their role here is to clarify what would be needed for a genuine checkpoint-selection experiment: saved parameters, a specified inference policy, fresh evaluation episodes, and repeated training runs under an executable protocol. Our archive reanalysis supplies none of those additional measurements and makes no claim to do so.

Classic Atari work in Mnih et al., 2013 and algorithmic improvements in Hessel et al., 2017 concern changes to learners rather than retrospective presentation rules. They therefore cannot validate the selection gain observed here. Towers et al., 2024 similarly provides an environment interface, not an interpretation of which temporal average should stand for a trained system. Separating these roles keeps a small reporting result from being mistaken for algorithmic progress.

Estimation after the selector has been frozen

For each of the forty-eight game-run curves, we compute selected-minus-terminal training return and selected-minus-terminal evaluation return. The three run contrasts are averaged within each game, leaving sixteen equally weighted game rows. We fit one intercept regression for each contrast. Model 1 describes the training gain that follows from the rule; Model 2 describes its corresponding evaluation gain.

Both models use HC3 covariance, following the heteroskedasticity-robust approach associated with White, 1980, and t15 intervals. Their R-squared values are zero because the models estimate means without explanatory covariates. No game fixed effects are required after each row has already been constructed as a within-game paired difference. The intervals remain conditional summaries over the benchmark games and recorded runs, rather than population guarantees for future games.

Dependence-aware assessment in Roberts et al., 2017 motivates this reduction to the level at which the practical comparisons differ. Agarwal et al., 2021 further cautions against strong aggregate claims from limited reinforcement-learning runs. We therefore do not use thousands of update records as the regression sample size. Neither HC3 nor a t reference distribution creates independent game sampling where the source collection did not provide it.

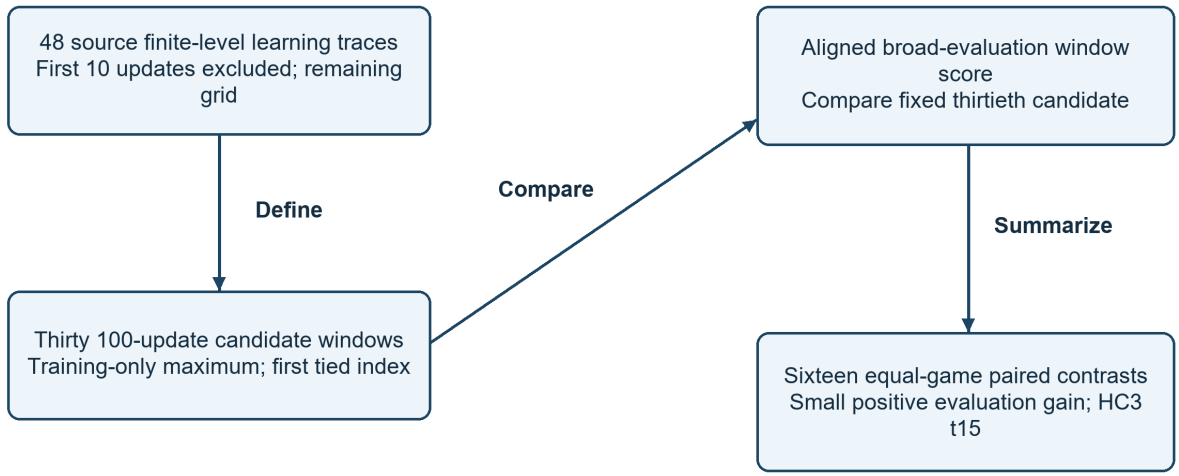


Figure 1. *Thirty candidate windows enter a training-only selection rule; the evaluation curve supplies a separate descriptive outcome at the frozen index. No deployment or causal path is asserted.*

The selected periods scored slightly better beyond training

The average selected training score was 0.675218, compared with 0.662867 for the terminal candidate. Their difference, 0.012352, is nonnegative by construction for every source run. The estimated working interval from 0.004837 to 0.019866 should therefore not be treated as a surprising discovery that maximum selection raises the score being maximized. The empirical information is its scale and distribution across the supplied games.

The corresponding evaluation means were 0.398110 for selected windows and 0.391602 for terminal windows. Their average difference was 0.006509, with a standard error of 0.003022 and a pointwise interval from 0.000067 to 0.012951. This supports a small positive recorded evaluation contrast under the chosen rule, while the lower limit shows that the distinction from zero is weak at conventional precision.

Table 1. *Selected-window minus fixed-terminal differences in normalized return. N equals sixteen game means. Both models are intercept-only OLS with HC3 standard errors and t15 intervals; neither contains controls or fixed effects.*

Model	Term	Estimate (SE)	95% CI	N
M1	Mean outcome/contrast	0.012352 (0.003526)	[0.004837, 0.019866]	16
M1: fit	R-squared = 0.0000	-	-	16
M2	Mean outcome/contrast	0.006509 (0.003022)	[0.000067, 0.012951]	16
M2: fit	R-squared = 0.0000	-	-	16

Model labels: M1 = Selected minus terminal training return; M2 = Selected minus terminal unrestricted evaluation.

Thirty-four of the forty-eight source runs selected a candidate earlier than the thirtieth window. Chosen indices ranged from eight to thirty. After averaging runs, thirteen games had a positive evaluation difference, two had a negative difference, and one had no change. Thus, the aggregate improvement was not created solely by one positive game, but the rule also did not improve evaluation in every environment.

The difference between the two aggregate gains was about 0.005843 normalized units. We report that quantity descriptively rather than attaching an uncomputed third interval to it. The training advantage and evaluation advantage are paired and dependent, so comparing their two separate confidence limits would not test their difference. Likewise, counting thirteen favorable game averages is not a test of success probabilities for independent future tasks.

The candidate grid defines a restricted question about presentation. It allows the selector to compare broad periods, which can suppress isolated favorable fluctuations but can also hide a short period of genuinely strong behavior. Increasing the window length would reduce the number of available candidates, while decreasing it would give the maximum more opportunities to capitalize on transient outcomes. Neither change can be assumed to preserve the small evaluation gain. We retain the thirty-window result as the completed analysis rather than choose a new width after observing its favorable direction.

There is also a distinction between choosing when to report and choosing when to stop collecting experience. Our retrospective rule uses a training window that may be followed by many later windows. An online stopping rule cannot know that its current score will remain the largest future score. Converting the retrospective maximum into a stopping policy would introduce a new information constraint and potentially a different distribution of selected times. The current findings therefore support disclosure of a reporting selector, with no claim that the same advantage would survive an implementable stopping decision.

The size and reliability of a favorable result

The evaluation interval barely excluding zero makes presentation especially important. Wasserstein and Lazar, 2016 argues against converting a threshold crossing into a complete scientific conclusion. Benjamini and Hochberg, 1995 addresses multiplicity when many hypotheses are tested; our two pointwise intervals are not a multiplicity-adjusted family or permission to search many undisclosed candidate definitions. The complete chosen grid is retained so that the reader can distinguish this specified calculation from a broad search for favorable summaries.

Henderson et al., 2017 reminds us that implementation and random variation can matter on the same scale as a small reported gain. Kalibera and Jones, 2013 develops rigorous performance-evaluation thinking in a different computational setting, relevant to the need for replication before treating a modest contrast as a durable engineering advantage. We do not translate normalized reward differences into speed, money, or saved training steps.

A robust estimator could downweight extreme game contrasts, as discussed in Hampel, 1974, but that would change the aggregation rule. We keep the equal-game mean and retain all observations. The influence perspective of Cook, 1977 motivates making individual rows available rather than silently suppressing unfavorable or influential games. No game was removed to preserve the small positive lower confidence limit.

Whether the gain is useful also depends on the intended objective. Elkan, 2001 discusses decision costs, and Gneiting and Raftery, 2007 links evaluation criteria to the quantities they are intended to assess. Training-window search has storage, evaluation, and review implications that are not measured here. A positive return contrast alone cannot establish that operating such a selector is worthwhile in a deployment workflow.

Reuse boundaries and a practical reporting rule

The weighting question in Solon et al., 2015 is relevant because game averages can answer different questions from episode-weighted summaries. We deliberately give each supplied game equal influence. The score-scale caution in Stevens, 1946 applies because source normalization does not ensure equal real-world value for equal increments across games. Another stakeholder could adopt a different utility, but it should be declared before interpreting the chosen window as best.

Documentation in Gebru et al., 2021 and reuse principles in Wilkinson et al., 2016 motivate retaining the original files, grid boundaries, selected indices, and both outcomes. Sculley et al., 2015 highlights the engineering risk of allowing evaluation dependencies to become implicit. An apparently simple best-window label depends here on buffering, window width, normalization, and which stream is allowed to select. Those choices should travel with the number.

The records support a cautious reporting rule: if a learning period is selected by training performance, state the selector and show the aligned broader evaluation rather than infer its value from the training maximum. In this instance, that broader value was mildly favorable. The evidence does not justify claiming universal overfitting from selection, a reliable checkpoint-recovery method, or improved interactive rendering. Fresh saved-policy evaluations would be needed to move from a descriptive window result to an operational selection recommendation.

References

1. Kapoor, Sayash; Narayanan, Arvind. (2023). Leakage and the reproducibility crisis in machine-learning-based science. Patterns. https://doi.org/10.1016/j.patter.2023.100804.
2. Wasserstein, Ronald L.; Lazar, Nicole A.. (2016). The ASA Statement on p -Values: Context, Process, and Purpose. The American Statistician. https://doi.org/10.1080/00031305.2016.1154108.
3. Henderson, Peter; Islam, Riashat; Bachman, Philip; Pineau, Joelle; Precup, Doina; Meger, David. (2017). Deep Reinforcement Learning that Matters. arXiv:1709.06560. https://arxiv.org/abs/1709.06560.
4. Benjamini, Yoav and Yosef Hochberg. “Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing.” *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 57, no. 1, 1995, pp. 289-300. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x.
5. White, Halbert. “A Heteroskedasticity-Consistent Covariance Matrix Estimator and a Direct Test for Heteroskedasticity.” *Econometrica*, vol. 48, no. 4, 1980, pp. 817-838. https://doi.org/10.2307/1912934.
6. Smith, James E.; Winkler, Robert L.. (2006). The Optimizer’s Curse: Skepticism and Postdecision Surprise in Decision Analysis. Management Science. https://doi.org/10.1287/mnsc.1050.0451.
7. Roberts, David R., et al. “Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure.” *Ecography*, vol. 40, no. 8, 2017, pp. 913-929. https://doi.org/10.1111/ecog.02881.
8. Sculley, D., et al. "Hidden Technical Debt in Machine Learning Systems." *Advances in Neural Information Processing Systems*, vol. 28, 2015. https://papers.nips.cc/paper/2015/hash/86df7dcfd896fcaf2674f757a2463eba-Abstract.html
9. Schulman, John; Wolski, Filip; Dhariwal, Prafulla; Radford, Alec; Klimov, Oleg. (2017). Proximal Policy Optimization Algorithms. arXiv:1707.06347. https://arxiv.org/abs/1707.06347.
10. Kalibera, Tomas; Jones, Richard. (2013). Rigorous benchmarking in reasonable time. Proceedings of the 2013 international symposium on memory management. https://doi.org/10.1145/2464157.2464160.
11. Geifman, Yonatan, and Ran El-Yaniv. “Selective Classification for Deep Neural Networks.” *Advances in Neural Information Processing Systems*, vol. 30, 2017. https://arxiv.org/abs/1705.08500.
12. Towers, Mark; Kwiatkowski, Ariel; Terry, Jordan; Balis, John U.; De Cola, Gianluca; Deleu, Tristan; Goulão, Manuel; Kallinteris, Andreas; Krimmel, Markus; KG, Arjun; Perez-Vicente, Rodrigo; Pierré, Andrea; Schulhoff, Sander; Tai, Jun Jet; Tan, Hannah; Younis, Omar G.. (2024). Gymnasium: A Standard Interface for Reinforcement Learning Environments. arXiv:2407.17032. https://arxiv.org/abs/2407.17032.
13. OpenAI. (2026). train-procgen: released easy-mode training and evaluation logs, commit1a2ae2194a61f76a733a39339530401c024c3ad8 [Archived repository; snapshot acquired8November2026]. https://github.com/openai/train-procgen/tree/1a2ae2194a61f76a733a39339530401c024c3ad8.

14. Wilkinson, Mark D.; Dumontier, Michel; Aalbersberg, IJsbrand Jan; Appleton, Gabrielle; Axton, Myles; Baak, Arie; Blomberg, Niklas; Boiten, Jan-Willem; da Silva Santos, Luiz Bonino; Bourne, Philip E.; Bouwman, Jildau; Brookes, Anthony J.; Clark, Tim; Crosas, Mercè; Dillo, Ingrid; Dumon, Olivier; Edmunds, Scott; Evelo, Chris T.; Finkers, Richard; Gonzalez-Beltran, Alejandra; Gray, Alasdair J.G.; Groth, Paul; Goble, Carole; Grethe, Jeffrey S.; Heringa, Jaap; 't Hoen, Peter A.C; Hooft, Rob; Kuhn, Tobias; Kok, Ruben; Kok, Joost; Lusher, Scott J.; Martone, Maryann E.; Mons, Albert; Packer, Abel L.; Persson, Bengt; Rocca-Serra, Philippe; Roos, Marco; van Schaik, Rene; Sansone, Susanna-Assunta; Schultes, Erik; Sengstag, Thierry; Slater, Ted; Strawn, George; Swertz, Morris A.; Thompson, Mark; van der Lei, Johan; van Mulligen, Erik; Velterop, Jan; Waagmeester, Andra; Wittenburg, Peter; Wolstencroft, Katherine; Zhao, Jun; Mons, Barend. (2016). The FAIR Guiding Principles for scientific data management and stewardship. Scientific Data. https://doi.org/10.1038/sdata.2016.18.

15. Varma, Sudhir and Richard Simon. "Bias in error estimation when using cross-validation for model selection." *BMC Bioinformatics*, vol. 7, no. 1, 2006. https://doi.org/10.1186/1471-2105-7-91.

16. Meng, Zijie; Che, Jinming; Wei, Bingcai; Cao, Xixin. (2026). Make a game: A novel paradigm for interactive game rendering. ICASSP 2026, 1026-1030. https://doi.org/10.1109/ICASSP55912.2026.11461927.

17. Cobbe, Karl, et al. "Leveraging Procedural Generation to Benchmark Reinforcement Learning" *arXiv preprint arXiv:1912.01588*, 2019. https://arxiv.org/abs/1912.01588.

18. Gneiting, Tilmann; Raftery, Adrian E. (2007). Strictly Proper Scoring Rules, Prediction, and Estimation. Journal of the American Statistical Association. https://doi.org/10.1198/016214506000001437.

19. Elkan, Charles. "The Foundations of Cost-Sensitive Learning." *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, 2001, pp. 973-978. https://cseweb.ucsd.edu/~elkan/rescale.pdf.

20. Agarwal, Rishabh, et al. "Deep Reinforcement Learning at the Edge of the Statistical Precipice" *arXiv preprint arXiv:2108.13264*, 2021. https://arxiv.org/abs/2108.13264.

21. Sutton, Richard S., and Andrew G. Barto. *Reinforcement Learning: An Introduction*. 2nd ed., MIT Press, 2018. http://incompleteideas.net/book/the-book-2nd.html.

22. Hessel, Matteo; Modayil, Joseph; van Hasselt, Hado; Schaul, Tom; Ostrovski, Georg; Dabney, Will; Horgan, Dan; Piot, Bilal; Azar, Mohammad; Silver, David. (2017). Rainbow: Combining Improvements in Deep Reinforcement Learning. arXiv:1710.02298. https://arxiv.org/abs/1710.02298.

23. Cook, R. Dennis. (1977). Detection of Influential Observation in Linear Regression. Technometrics. https://doi.org/10.1080/00401706.1977.10489493.

24. Solon, Gary; Haider, Steven J.; Wooldridge, Jeffrey M.. (2015). What Are We Weighting For?. Journal of Human Resources. https://doi.org/10.3368/jhr.50.2.301.

25. Mnih, Volodymyr; Kavukcuoglu, Koray; Silver, David; Graves, Alex; Antonoglou, Ioannis; Wierstra, Daan; Riedmiller, Martin. (2013). Playing Atari with Deep Reinforcement Learning. arXiv:1312.5602. https://arxiv.org/abs/1312.5602.

26. Stevens, S. S.. (1946). On the Theory of Scales of Measurement. Science. https://doi.org/10.1126/science.103.2684.677.

27. Hampel, Frank R.. (1974). The Influence Curve and its Role in Robust Estimation. Journal of the American Statistical Association. https://doi.org/10.1080/01621459.1974.10482962.

28. Raffin, Antonin; Hill, Ashley; Gleave, Adam; Kanervisto, Anssi; Ernestus, Maximilian; Dormann, Noah. (2021). Stable-Baselines3: Reliable Reinforcement Learning Implementations. Journal of Machine Learning Research 22(268), 1-8. https://jmlr.org/papers/v22/20-1364.html.

29. Huang, Shengyi, et al. "CleanRL: High-quality Single-file Implementations of Deep Reinforcement Learning Algorithms." *arXiv preprint arXiv:2111.08819*, 2021. https://arxiv.org/abs/2111.08819.

30. Gebru, Timnit, et al. "Datasheets for datasets" *Communications of the ACM*, 2021. https://doi.org/10.1145/3458723.