Threshold Success and the Meaning of a Normalized Score of One

Wesley Mason (MC), Derek Franklin (MU), Jack Barrett (AD)

Abstract

Reaching a score threshold and improving a mean are different outcomes. In two public BBF result files, equally weighted game-level probabilities of a normalized score of at least one are 35.51 and 12.36 percent. Their difference is 23.15 percentage points, with a game-clustered interval from 13.89 to 32.41. Twenty-two main-file game means cross the threshold, compared with seven comparison-file means. These quantities answer different questions. The analysis develops their implications for reporting a declared success criterion in interactive rendering without interpreting agent returns as visual-quality measurements.

A threshold needs an interpretation and a unit

The event that one run reaches a reference score differs from the event that a game's average score reaches it. A research report can reasonably examine both, provided it does not switch between them while describing success. A run-level event concerns variability across recorded experiments. A threshold applied to a game mean concerns the location of the estimated task distribution. The first can be common even when the mean remains below the reference, or uncommon when a few unusually large outcomes lift the mean.

Make a game raises a related problem for generated interaction (Meng et al., 2026a). A success criterion might concern whether a command produces an intended response, whether a trajectory remains usable, or whether a viewer accepts its appearance. These criteria need different observations and denominators. The present analysis uses an existing agent benchmark to make the distinction concrete. It performs no evaluation of that renderer. The available source description motivates the control question, while the public score archive supplies the actual numeric evidence discussed here.

Two ways of counting threshold attainment

The inputs are the official main RR8 BBF score file and its file labeled without resets. Across them there are 2,912 unique configuration-game-seed records covering 55 games. The run-level indicator is one when the released normalized score is at least one, including equality. No threshold is selected from the observed distribution. The score is used as supplied rather than recomputed from external baseline tables. Consequently, the result remains a statement about this released normalized field and its reference convention.

Giving every run equal weight yields attainment frequencies of 39.53 percent in the main file and 12.29 percent in the comparison file. Giving every game equal influence yields 35.51 percent and 12.36 percent. A balanced linear probability model with game fixed effects estimates their difference as 0.2315. Standard errors clustered by the 55 games give an interval from 0.1389 to 0.3241. Those values describe a probability contrast under a declared task weighting. They are not a percentage increase in rendering quality or a newly identified component effect.

Why a count of successful games is another endpoint

Separately, the analysis calculates the mean normalized score within each configuration and game, then asks whether that mean is at least one. The criterion is met by 22 of the 55 main-file games and seven comparison-file games. This does not mean that every run in those 22 games was successful. Nor does a game mean below one imply that no individual run attained the threshold. The mean and the event probability describe different aspects of the recorded outcome distribution.

This distinction suggests a more informative renderer evaluation than a single passed-or-failed label. A study could report the fraction of histories satisfying a task requirement, the fraction of environments whose estimated success probability exceeds a declared standard, and a continuous measure of the severity of violations. These summaries would be linked through their definitions, not assumed interchangeable. The standard should be established through the intended use or an independent design decision. Selecting it after examining all candidate models would turn a useful criterion into another route for favorable reporting.

A threshold also needs a time horizon. An interaction can satisfy a requirement initially and fail later. Declaring success after the first correct frame asks a different question from retaining correctness over an entire trajectory. Repeated opportunities and early stopping can change apparent event probabilities even when the underlying response process is unchanged. An experimental protocol should therefore preserve the complete histories used for scoring and specify whether a later violation overrides an earlier success. The BBF files analyzed here contain aggregate run scores rather than those rendering trajectories, so they cannot resolve this temporal issue.

Reading two thresholds on the same score scale

A small change in normalization near one can alter the hit rate while leaving the continuous ordering intact. The issue becomes concrete in a hypothetical assessment. A configuration can convert many failures into moderate returns without adding runs above the human reference. Use game-level empirical distribution to connect share at or above one and share below zero to the fraction of games with a material change at either boundary. Display a survival curve at the fixed origin and reference point, then inspect the size of the discontinuity at each tie.

A reader can reconstruct the threshold statistic from an empirical distribution by finding the share of runs to the right of one, including ties. At the origin, ties belong with the nonnegative observations. These two edge conventions are particularly important for games with repeated discrete returns. A follow-up could count exact boundary observations and evaluate movement across a narrow, prespecified threshold neighborhood. A reversal confined to tied or almost tied scores calls for a more qualified claim than a shift extending through the surrounding range.

A result about the fraction of games with a material change at either boundary needs a denominator that includes all eligible attempts, including ones without a usable score. For the game-level empirical distribution, log interruptions and timeouts separately so readers can see whether a response was observed.

Record attempts in their original order and store state, input and response separately for game-level empirical distribution. When the same design produces several observations of game-level empirical distribution, start with a group-level contrast and inspect the spread among groups. For share below zero, compare conclusions after varying the permitted response window within a declared range.

Use the comparison with share at or above one to identify which parts of the finding survive a reasonable change in the decision threshold. For share at or above one, report whether agreement concerns game progress, visible state or the timing of the response. Collect a separate set of game-level empirical distribution under documented conditions before claiming that the contrast generalizes.

Systems and comparators

The emulator protocol keeps the task definition observable, including how action sets and episode boundaries affect which returns can legitimately be compared. During validation, earlier research investigates data-efficient Atari learning using a learned planning model (Ye et al., 2021). Data-efficiency comparisons benefit from matching environment-interaction budgets and specifying whether any additional planning or pretrained information differs between systems. A relevant contribution defines an emulator-based environment for testing general game agents (Bellemare et al., 2012).

Earlier research uses a transformer world model for sample-efficient reinforcement learning (Micheli et al., 2022). One approach learns action values from Atari screen observations (Mnih et al., 2013). In practical terms, an agent trained from pixels provides a control for task success that can be compared with direct assessments of rendering or transition fidelity. An ablation can hold histories fixed and alter the predictive model, separating changes to rollout quality from changes in task selection.

The distribution that generated an archive constrains follow-up evaluation, particularly where difficult states or rare recovery actions are sparsely represented. A learned latent action and a recorded controller action require separate interpretations, especially when assessing whether two identical commands produce comparable state transitions. A methodological precedent learns interactive environments from video through latent action modeling (Bruce et al., 2024). One study reviews the problem of learning policies from a fixed set of experience (Levine et al., 2020).

A sequential decision formulation keeps reward, action, observation and state distinct, which helps determine where a claimed capability can actually be measured. For implementation, evaluate generated images and agent returns separately, while recording whether competing models received the same interaction histories. Another study defines sequential decision problems in terms of state, action and return (Sutton and Barto, 2018). An existing method uses a diffusion world model to investigate visual detail in Atari (Alonso et al., 2024).

Action conditioning lets an evaluator test whether a command changed the game despite the plausibility of the rendered frames. Tests on unseen scenes and seen control types can separate transfer to new appearance from transfer to new dynamics and commands. In a controlled test, related work demonstrates action-conditioned neural generation in a game engine setting (Valevski et al., 2024). Related work investigates action-conditioned game video and transfer to new scenes (Yu et al., 2025).

A relevant contribution uses imagined trajectories in learned latent dynamics for control (Hafner et al., 2019). One study uses first-frame information to guide spatial and temporal consistency (Ren et al., 2024). First-frame conditioning gives a clear initial anchor; later tests can distinguish ordinary appearance change from drift after a break in visibility. In practical terms, latent rollout and task reward are different outputs, and their errors can be isolated by comparing predictions over a common initial history.

Analysis and interpretation

One study conditions sequence modeling on returns in offline control (Chen et al., 2021). Another study investigates reliable comparisons of reinforcement learning under limited runs (Agarwal et al., 2021). For a prospective comparison, uncertainty intervals and performance profiles help show whether an apparent advantage persists across titles with very different return distributions. Conditioning on a desired return does not establish that the archive contains successful examples of the associated trajectory or command sequence.

The archived rows support comparisons within recorded game and configuration cells, with substantial differences in how many seeds contribute observations to each cell. For a prospective comparison, a longitudinal formulation can separate baseline differences between units from changes within the same unit as the sequence progresses. Related work investigates data-efficient Atari agents and supplies the comparison's released score archive (Schwarzer et al., 2023). One study models repeated longitudinal measurements with random effects (Laird and Ware, 1982).

A relevant contribution combines an automatic curriculum with a reusable Minecraft skill library (Wang et al., 2023). In practical terms, seed choices and implementation details can create measurable variation, making independent repetitions and version records part of the evidence rather than ancillary information. Another study examines how experimental choices affect reproducibility in deep reinforcement learning (Henderson et al., 2017). A curriculum and a reusable skill library can change what a system attempts, so log the attempted tasks alongside completed ones.

A family of endpoints should be declared before evaluation so that reported discoveries reflect every comparison eligible to appear in the result. Mixed-effects models account for repeated observations of the same unit, while the choice of grouping still needs a design rationale. In practical terms, a methodological precedent controls the false discovery rate for a declared family of hypotheses (Benjamini and Hochberg, 1995). An existing method describes estimation and implementation of linear mixed-effects models (Bates et al., 2015).

Repetitions from a common generating condition can have correlated errors, making the number of independent groups more informative than the nominal row count. Because several changes coexist in one system, isolating a component requires an intervention or a comparison with the other choices held fixed. One study combines several extensions of deep Q-learning in a common Atari agent (Hessel et al., 2017). One published study provides guidance on inference under dependence within specified groups (Cameron and Miller, 2015).

A methodological precedent provides a configurable driving simulation environment (Dosovitskiy et al., 2017). A methodological precedent allows covariance estimates to account for heteroskedasticity and serial dependence (Newey and West, 1987). In a controlled test, serial dependence makes neighboring measurements count differently for uncertainty, even if the average contrast itself is calculated using all observed time points. A simulator can vary visibility, traffic and camera placement independently while leaving the underlying events observable for verification against the rendered frames.

Samples and evidence

The task and knowledge inventories help separate performance on well-represented activities from evidence about broader transfer to unusual interactions. In practical terms, another study asks dataset creators to record collection, intended use and limitations (Gebru et al., 2021). Documenting construction and intended use clarifies which exclusions occurred during collection, which arose from a model failure and which were imposed during analysis. One published study organizes open-ended Minecraft tasks and external knowledge resources (Fan et al., 2022).

Measurement and error

Report which raters, measurements and agreement definition enter the coefficient, since different designs give superficially similar statistics different meanings. For implementation, local structural agreement can be inspected alongside task metrics when pixel changes are expected but geometry or regional coherence should remain stable. Another study distinguishes forms of intraclass correlation according to the rating design (Koo and Li, 2016). An existing method compares the local structure of a pair of images (Wang et al., 2004). One published study provides standard environment interfaces for reinforcement learning research (Brockman et al., 2016).

Conclusion

The released configurations differ substantially in run-level threshold attainment, while the count of threshold-crossing game means supplies a separate view. Both can be reported without pretending that they estimate the same quantity. For generated interactive systems, a declared success standard should identify its observation unit, reference point, and time horizon. Its interpretation should survive changes in how many repetitions a game contributes. These requirements make a threshold scientifically useful and prevent a convenient pass rate from standing in for capabilities that were never directly observed.

References

Bellemare, Marc G., et al.. "The Arcade Learning Environment: An Evaluation Platform for General Agents" *arXiv preprint arXiv:1207.4708*, 2012. https://arxiv.org/abs/1207.4708.

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.

Mnih, Volodymyr, et al. "Playing Atari with Deep Reinforcement Learning." *arXiv preprint arXiv:1312.5602*, 2013. https://arxiv.org/abs/1312.5602.

Chen, Lili, et al. "Decision Transformer: Reinforcement Learning via Sequence Modeling." *arXiv preprint arXiv:2106.01345*, 2021. https://arxiv.org/abs/2106.01345.

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

Schwarzer, Max, et al.. "Bigger, Better, Faster: Human-level Atari with human-level efficiency" *arXiv preprint arXiv:2305.19452*, 2023. https://arxiv.org/abs/2305.19452.

Meng, Zijie, et al. "Make a game: A novel paradigm for interactive game rendering." *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2026.

Laird, Nan M. and James H. Ware. “Random-Effects Models for Longitudinal Data.” *Biometrics*, vol. 38, no. 4, 1982, pp. 963. https://doi.org/10.2307/2529876.

Micheli, Vincent, et al.. "Transformers are Sample-Efficient World Models" *arXiv preprint arXiv:2209.00588*, 2022. https://arxiv.org/abs/2209.00588.

Henderson, Peter, et al. "Deep Reinforcement Learning that Matters." *arXiv preprint arXiv:1709.06560*, 2017. https://arxiv.org/abs/1709.06560.

Bruce, Jake, et al.. "Genie: Generative Interactive Environments" *arXiv preprint arXiv:2402.15391*, 2024. https://arxiv.org/abs/2402.15391.

Levine, Sergey, Aviral Kumar, George Tucker, and Justin Fu. “Offline Reinforcement Learning: Tutorial, Review, and Perspectives on Open Problems.” *arXiv*, 2020, arXiv:2005.01643. https://arxiv.org/abs/2005.01643.

Alonso, Eloi, et al.. "Diffusion for World Modeling: Visual Details Matter in Atari" *arXiv preprint arXiv:2405.12399*, 2024. https://arxiv.org/abs/2405.12399.

Wang, Guanzhi, et al. "Voyager: An Open-Ended Embodied Agent with Large Language Models." *arXiv preprint arXiv:2305.16291*, 2023. https://arxiv.org/abs/2305.16291.

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

Valevski, Dani, et al.. "Diffusion Models Are Real-Time Game Engines" *arXiv preprint arXiv:2408.14837*, 2024. https://arxiv.org/abs/2408.14837.

Yu, Jiwen, et al. "GameFactory: Creating New Games with Generative Interactive Videos." *arXiv preprint arXiv:2501.08325*, 2025. https://arxiv.org/abs/2501.08325.

Ren, Weiming, et al. "ConsistI2V: Enhancing Visual Consistency for Image-to-Video Generation." *arXiv preprint arXiv:2402.04324*, 2024. https://arxiv.org/abs/2402.04324.

Wang, Zhou, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. "Image Quality Assessment: From Error Visibility to Structural Similarity." *IEEE Transactions on Image Processing*, vol. 13, no. 4, 2004, pp. 600-612. https://doi.org/10.1109/TIP.2003.819861.

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.

Bates, Douglas, Martin Mächler, Ben Bolker, and Steve Walker. “Fitting Linear Mixed-Effects Models Using lme4.” *Journal of Statistical Software*, vol. 67, no. 1, 2015. https://doi.org/10.18637/jss.v067.i01.

Hafner, Danijar, et al.. "Dream to Control: Learning Behaviors by Latent Imagination" *arXiv preprint arXiv:1912.01603*, 2019. https://arxiv.org/abs/1912.01603.

Hessel, Matteo, et al. "Rainbow: Combining Improvements in Deep Reinforcement Learning." *arXiv preprint arXiv:1710.02298*, 2017. https://arxiv.org/abs/1710.02298.

Brockman, Greg, et al.. "OpenAI Gym" *arXiv preprint arXiv:1606.01540*, 2016. https://arxiv.org/abs/1606.01540.

Cameron, A. Colin, and Douglas L. Miller. “A Practitioner’s Guide to Cluster-Robust Inference.” *Journal of Human Resources*, vol. 50, no. 2, 2015, pp. 317-372. https://doi.org/10.3368/jhr.50.2.317.

Ye, Weirui, et al. "Mastering Atari Games with Limited Data." *arXiv preprint arXiv:2111.00210*, 2021. https://arxiv.org/abs/2111.00210.

Newey, Whitney K., and Kenneth D. West. “A Simple, Positive Semi-Definite, Heteroskedasticity and Autocorrelation Consistent Covariance Matrix.” *Econometrica*, vol. 55, no. 3, 1987, pp. 703-708. https://doi.org/10.2307/1913610.

Fan, Linxi, et al. "MineDojo: Building Open-Ended Embodied Agents with Internet-Scale Knowledge." *arXiv preprint arXiv:2206.08853*, 2022. https://arxiv.org/abs/2206.08853.

Koo, Terry K., and Mae Y. Li. "A Guideline of Selecting and Reporting Intraclass Correlation Coefficients for Reliability Research." *Journal of Chiropractic Medicine*, vol. 15, no. 2, 2016, pp. 155-163. https://doi.org/10.1016/j.jcm.2016.02.012.

Dosovitskiy, Alexey, et al. "CARLA: An Open Urban Driving Simulator." *arXiv preprint arXiv:1711.03938*, 2017. https://arxiv.org/abs/1711.03938.