Journal of Physics Special Topics
An undergraduate physics journal

A2 2 Minecraft Speedrunning: Scientific Data Analysis in Practice

F. Stacey, A. Hawkins, R. McElroy, M. Robertson, S. Dias

Department of Physics and Astronomy, University of Leicester, Leicester, LE1 7RH

October 16, 2026

Abstract

This study analyses MCSR Ranked Minecraft speedrunning data to predict future world-record times. Monthly top 1% and top 0.1% performances were modelled using a decaying exponential, with a modified model for world-record progression. The model predicts a world-record time of 338.6 s after 12 months, approaching a projected long-term limit of 325.9 s. These results suggest that elite performance will continue to improve, but at a decreasing rate.

Introduction

Speedrunning is the practice of completing a video game, or achieving a specific goal within it, as quickly as possible. Minecraft speedrunning [1] provides an interesting example as completion times largely depend on luck due to in game randomness. As player skill improves and new strategies are developed, times fall rapidly [2]. However, as the game matures and strategies become more refined, further improvements become increasingly difficult and luck becomes more significant.

MCSR Ranked is a modified version of Minecraft in which players race on the same filtered seed, standardising much of the game's randomness while retaining some variation [3]. This reduces the effect of luck and allows players to focus more on skill, providing a useful way to study high-level performance.

The main aim of this paper is to analyse MCSR Ranked data to identify trends in elite performance and use these trends to predict the future progression of the speedrun world record.

Data Collection

We obtained data using the MCSR ranked API [4], collecting all recorded runs from the top 50 runners each season since the mod launched in March 2023. We excluded lost games to keep only completed runs and to avoid duplicate matches.

Data Processing

To measure changes at the highest level we grouped the runs by month and calculated the fastest 1% and 0.1% along with the world record at the time. Using percentages allows us to more easily to account for top level improvement over long periods of world record stagnation. The first four months, covering the launch of MCSR Ranked, were excluded as top players were not competing yet. Runs were included once the ranked record matched the main-game record.

Model Fitting

We initially modelled the top 1% performance as decaying exponential:

$$T_1(t) = a + be^{-ct} \tag{1}$$

This represents a rapid initial improvement followed by progressively smaller improvements as players approach a theoretical perfect time. We modelled the top 0.1% as a constant time gap below the top 1%:

$$T_{0.1}(t) = T_1(t) - g_{01} \tag{2}$$

where g_{01} is the mean observed gap between the top 1% and top 0.1% across all months in the dataset.

Modelling the world record with the same exponential form did not describe the observed progression well, particularly in recent data. We therefore modelled it as a shrinking gap below the top 0.1%:

$$WR(t) = T_{0.1}(t) - \left[g_{\min} + \delta e^{-dt}\right]. \tag{3}$$

Combining this with the top 0.1% model gives

$$WR(t) = a - g_{01} - g_{\min} + be^{-ct} - \delta e^{-dt} \tag{4}$$

where $g_{\min}$ is the minimum gap and δe^{-dt} is the gap that decreases over time. We fitted the parameters using SciPy's `curve_fit` [5], which uses the Trust Region Reflective algorithm [6] to iteratively update the parameters from initial guesses. We then extended the fitted curves beyond the observed data to estimate future top 1%, 0.1%, and world record progression.

Results

The analysis shows a clear improvement in Minecraft speedrunning performance over the period studied. Both the top 1% and top 0.1% of runs follow a similar downward trend, with the rate of improvement decreasing over time.

The world record follows a similar trend unlike the percentile trends, however, the world record does not follow the same exponential curve particularly well. The modified model provides a better representation by allowing the gap between the world record and the top 0.1% to decrease over time. As shown in Figure 1, this causes the world record prediction to gradually converge towards the projected elite performance.

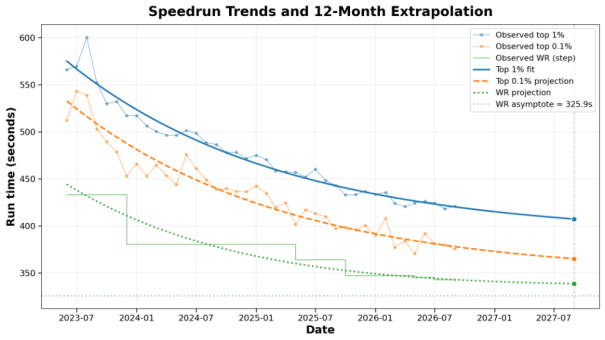


Figure 1: Observed elite Minecraft speedrun performance and fitted model predictions. The curves show the top 1%, top 0.1%, and world record performance over time, with the fitted curves extended beyond the observed data to predict future progression.

The fitted model can therefore be extended beyond the observed data to estimate future world-record progression. After 12 months, the model predicts a world record time of 338.6 s, representing an improvement of approximately 3 s compared with the current world record of 341.5 s. The fitted curve approaches an asymptote of 325.9 s, which represents the projected long-term performance limit of the model. This suggests that while further improvements are expected, the potential for improvement will become increasingly small as performance approaches this value.

Conclusion

This study shows that elite Minecraft speedrun performance has improved over time, with the rate of improvement gradually decreasing. The fitted model successfully captures the overall trend and predicts a world-record time of 338.6 s after 12 months. The model also approaches a projected long-term performance limit of 325.9 s. Overall, the results suggest that future world-record improvements are likely to continue, but at a slower rate as performance approaches this limit.

References

[1] Minecraft, *Speedrunning Minecraft*, `https://www.minecraft.net/en-us/article/speedrunning-minecraft` [Accessed 28 September 2026].

[2] E. Erdil and J. Sevilla, "Power Law Trends in Speedrunning and Machine Learning," arXiv:2304.10004 (2023).

[3] MCSR Ranked, *RNG Standardization*, `https://wiki.mcsrranked.com/gameplay/rng` [Accessed 28 September 2026].

[4] MCSR Ranked, *MCSR Ranked API Documentation*, `https://docs.mcsrranked.com/` [Accessed 28 September 2026].

[5] P. Virtanen *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," Nature Methods **17**, 261–272 (2020).

[6] M. A. Branch, T. F. Coleman, and Y. Li, "A Subspace, Interior, and Conjugate Gradient Method for Large-Scale Bound-Constrained Minimization Problems," SIAM Journal on Scientific Computing **21**, 1–23 (1999).