FORECASTBENCH-SIM: FORECASTING SIMULATED WORLDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Predicting how a complex system will evolve is among the most general tests of reasoning under uncertainty, and among the hardest to benchmark. Real-world forecasting benchmarks often resolve slowly and provide sparse evidence of accuracy in forecasting rare or counterfactual events. We introduce ForecastBench-Sim, a forecasting benchmark that asks models to forecast the outcomes of simulated worlds. We generate questions about rare events, conditional outcomes, and responses to causal interventions. Across 24 models, three simulations, and nine distinct tasks, model scores closely track general measures of model capabilities ($\rho = 0.86$ $[0.69, 0.93]$). ForecastBench-Sim provides tasks whose question sets can be refreshed, varied in difficulty, and immediately scored. We discuss potential mechanisms to prevent saturation and avoid benchmark gaming, focusing on the automatable extensibility of our framework to new unseen questions.

1 INTRODUCTION

Forecasting is an increasingly valuable skill for large language models (LLMs), and recent results indicate that LLM-based forecasters are approaching the best human forecasters (Halawi et al., 2024; Forecasting Research Institute, 2026). However, existing forecasting benchmarks (like ForecastBench; Karger et al., 2025a) rely on observed, real-world forecasting questions, which bring a few notable limitations. First, real-world forecasting questions must resolve, which can take weeks, if not months, forcing evaluators to wait to see evaluation results. Second, real-world forecasting questions are poorly suited to understanding rare events. Rare events are sparse in real-world data by nature, making it difficult to estimate how well models perform on forecasting such questions. However, tail events are highly impactful in real-world forecasting domains: we very often want to know how likely an unlikely event is, and how that likelihood is changing over time. Third, interventional and correlative conditional questions are poorly suited to real-world benchmarks. The real world provides us with only a single timeline. For interventions (i.e., causal effects), the real world cannot reveal both an intervention outcome and its counterfactual. For correlative conditionals, the real world does not always provide suitable repeated observations to estimate the true likelihood of two events co-occurring. Again, both types of conditional questions are highly relevant to real-world forecasting: we frequently want to understand how an intervention will affect a matter of interest, or how the occurrence of one event changes the probability of another.

To address these limitations, we introduce ForecastBench-Sim, a simulated-world forecasting benchmark. ForecastBench-Sim relies on forecasting simulated worlds: it runs a world up to a snapshot point, then describes its state in a world report. The benchmark then poses questions to the model under evaluation about that world's future state. These questions are resolved by playing the world forward from the snapshot (Figure 1). World plays are deterministic given a seed; by playing worlds using many seeds, we can estimate true event probabilities. We can then score a forecast by its excess score (i.e., the loss beyond that of the ground-truth probability itself). Section 2 defines these terms.

Our contributions are as follows.

- A benchmark design (Section 2) under which any simulation can yield forecasting questions that are evaluable against true probability estimates, including rare and conditional forecasting questions.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

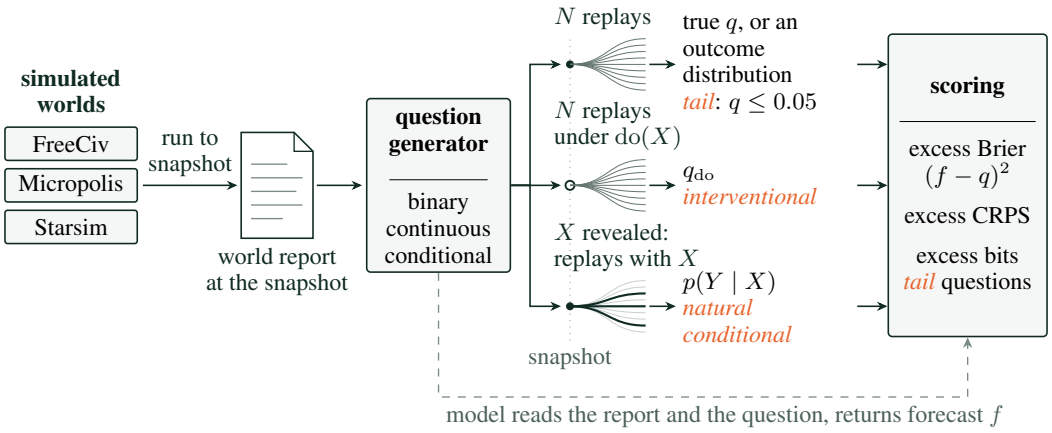


Figure 1: ForecastBench-Sim. A simulated world is run to a snapshot point, producing a world report. The benchmark generates forecasting questions about the world's future state. The benchmark then runs the simulation forward, producing a ground truth for the particular questions. By replaying the same world under new random seeds, we can give each question an estimate of its true ground-truth probability q or outcome distribution. Model forecasts are scored by their excess over the ground truth.

- Results for a shared panel of 24 models on three simulations: Micropolis (Hopkins et al., 2008) for tail questions, FreeCiv (The Freeciv Project, 2026a) for natural conditionals and Starsim (Kerr et al., 2024) for interventional conditionals (Sections 3 to 5).
- A validation against the Epoch Capabilities Index (Epoch AI, 2026) (Section 6). The sign-adjusted Spearman correlation between capability and each of nine scores is positive, and its 95% bootstrap interval over models excludes zero for seven of the nine. A combined score fitted across simulations (Section C.5) correlates with capability at 0.86.
- A planned release of all code required to reproduce our results, or extend our results with new simulations (Section E).

2 Benchmark Design

ForecastBench-Sim uses *simulators* to generate forecasting questions. We use two games and one disease model; the important feature is that the simulators are (a) complex and (b) deterministic given a seed. A *world* is a simulated scenario within a simulation: a city in Micropolis, a game in FreeCiv, or a scenario containing one or two regions in Starsim. A simulator therefore supplies many worlds, and every question is asked about a snapshot of one world. We run the world to a chosen turn, the *snapshot*, and write a *world report*, a text document that gives the state at the snapshot, the history of the main quantities and the events so far. Forecasters (in practice, models) see only the world report and questions. Worlds are then run many times with different seeds, creating a variety of *replays*, from which we can estimate the ground truth probability for any given event (Figure 1).

2.1 Questions and Ground Truth

ForecastBench-Sim allows multiple types of questions. For *binary questions*, the ground-truth probability q is the share of the N replays in which the event occurs. For a *continuous question*, the ground truth is the outcome distribution of the asked quantity over the replays; forecasters answer with per-quantile estimates. A *tail question* (i.e., a question with a rare outcome) is a binary question with $0 < q \leq 0.05$, and a *mid-range question* one with $0.05 < q < 0.95$; both are selected on q before any forecast is made, so a world contributes as many rare events as its replays contain.

ForecastBench-Sim also allows conditional questions. A *natural conditional* reveals one true fact X about the interval after the snapshot and asks the same question Y again. Its ground truth is $p(Y \mid X)$, the share of replays in which Y occurs among those in which X holds. An *interventional conditional* applies an intervention (e.g. a vaccination campaign) to a snapshot and asks about the

world conditional on the intervention being applied. Its ground truth comes from replays of the world with and without the intervention.

2.2 SCORING

We score forecasts against the replay ground truth using proper scoring rules or their quantile approximations, and we report the *excess score*, the expected score of the forecast under the replay distribution minus the expected score of the ground truth itself. For a strictly proper rule on full distributions, excess is nonnegative and zero only at the ground-truth distribution (Gneiting & Raftery, 2007). The finite-quantile approximation instead assesses the elicited quantiles. For a binary question with forecast f, the expected Brier score (Brier, 1950) over the replays is $(f-q)^2+q(1-q)$. The second term is the uncertainty component of Murphy (1973), and the *excess Brier score* $(f-q)^2$ is the binary metric on mid-range questions in every simulation. For a continuous question we approximate the CRPS (Matheson & Winkler, 1976) with pinball losses at the elicited quantiles. The *floor* is the same score evaluated at the replay distribution's own quantiles, and the *excess CRPS* is the expected score over replays minus this floor. Matching the five elicited quantiles need not match the full distribution, and discrete replay distributions can have nonunique quantile minimizers. Because continuous quantities differ in units, each simulation divides excess CRPS by a fixed scale, which gives the *excess nCRPS*: per city and metric in Micropolis and per horizon in Starsim. Squared error does little to distinguish tail forecasts, since for any forecast in $[0, 0.05]$ the excess Brier score on a tail question is at most 0.0025. FreeCiv therefore also scores tail questions in *excess bits*, $\mathrm{KL}(q \,\|\, f)$, the excess of the logarithmic score, which penalizes probability errors logarithmically, with finite endpoint values after clipping. We bound forecasts to $[0.001, 0.999]$ first (Section A.4). Starsim scores its binary questions in excess bits as well. Micropolis scores its tail questions in excess bits, as FreeCiv does.

2.3 SIMULATIONS, MODELS AND PROTOCOL

Table 1 summarizes the three simulations; Sections 3 to 5 give the particulars of each, and Section A the program versions, seeds, run counts and replacements. We ran the same 24 models in every simulation through OpenRouter. We chose them so that each has a score on the Epoch Capabilities Index (ECI), a single capability scale estimated across many public benchmarks by an item-response model (Ho et al., 2025; Epoch AI, 2026). We requested the lowest supported reasoning setting where controllable, or a 1,024-token budget for budget-only models; four models offered no control, and requested settings did not equalize compute (Section A.5).

3 MICROPOLIS: RARE EVENTS IN A CITY SIMULATOR

To evaluate performance on tail events, we use Micropolis as our simulator. Micropolis is the open-source release of the classic 1989 SimCity, the original city-builder game. Its cellular-automaton-based engine simulates the evolution over time and space of crime, pollution, car traffic, population, and other variables. In the original game, a human player takes the role of Mayor and affects city development, but Micropolis can be run in a headless, no-Mayor mode. We use rollouts of these headless simulations, starting from the pre-built cities that ship with the game.

A game mechanic of interest is that of disasters. Some of these occur at fixed background rates, others with rates proportional to city characteristics, and others occur only when a variable crosses some threshold. They provide an interesting setting for studying forecasts of rare events.

Setup. We ran 15 of the game's built-in cities from one seed and paused each at game years 20 and 30. For each of these 30 snapshots, models were asked to make forecasts with horizons of 3, 5, 7, and 10 game years. The world report describes the city state, includes time series of 6 metrics with 6-month resolution and an event log that includes all disasters that happened so far. At every snapshot and horizon we ask 25 binary questions, a total of 3,000 questions per model. The analysis assigns each of these to a set by its realized q, as defined in Section 2.1: 940 tail questions with $0 < q \leq 5\%$, scored in excess bits, and 983 mid-range questions with $5\% < q < 95\%$, scored in excess Brier. The 826 questions with $q = 0$ and the 251 with $q \geq 95\%$ are left out of both sets. The continuous set asks for five percentiles of 5 metrics at each horizon, 600 questions per model. A snapshot's binary questions are split across two prompts of 50 questions, and its continuous questions share

Table 1: The three simulators and their parameters. Parameters necessarily vary somewhat between simulators, as each simulator and question under evaluation (tail events, interventional conditionals, natural conditionals) requires a bespoke setup.

	Micropolis	FreeCiv	Starsim
Substrate	Open-source release of SimCity Classic (Hopkins et al., 2008); no player input; disasters on	Open-source clone of Civilization (The Freeciv Project, 2026a); five civilizations competing with in-game AI on "hard"	Starsim (Kerr et al., 2024); disease simulator
Worlds	15 cities, 2 snapshots each	8 games	4 worlds, 1 snapshot each
Snapshot	game years 20 and 30 (turns 960 and 1,440)	turn 60	day 20
Horizons	3, 5, 7 and 10 game years after the snapshot	turns 90, 120, 150, 180 and 210	days 40 and 60
Replays per snapshot	1,000	1,000	1,000 per arm for continuous rescore; binary retains original matched seeds
Items per model	3,000 binary asked, of which 983 mid-range and 940 tail; 600 continuous	750 mid-range; 300 tail; 300 continuous; 400 natural-conditional cells	8 continuous; 24 interventional; 24 historical binary items in appendix

one prompt of 20 questions. The prompt casts the model as a superforecaster and names the scoring rule (see Section A.1 for details).

3.1 Results

Scores on the three question sets track ECI: ρ is 0.80 for mid-range, 0.75 for tail and 0.82 for continuous (Figure 2). Per-model scores are in Table 14. For the binary questions, pooling all 3,000 (including $q = 0$ and $q \geq 95\%$) gives ECI correlations of 0.80 for excess bits and 0.82 for excess Brier.

In the tail set, every model overestimates the probability of rare events (mean forecasts 0.05–0.30 against a mean q of 0.015) and none beats a constant 0.05 forecast. More capable models overshoot less ($\rho = -0.68$ of mean forecast with ECI). The size of this bias depends on elicitation details (Section D.1).

In the mid-range questions, individual model biases run in both directions and 23 of 24 models beat a constant 0.5 forecast. Mid-range loss follows *discrimination*, the correlation between a model's forecasts and replay truth: across models the two correlate at -0.88, and discrimination rises with ECI ($\rho = 0.77$).

We find continuous loss rises with horizon, from 0.028 excess nCRPS at 3 years to 0.050 at 10 (Table 16). The binary loss does not (Figure 5), which reflects the question mix changing with horizon (Section C.1).

4 FreeCiv: natural conditionals in a strategy game

4.1 Worlds and question sets

To evaluate performance on natural conditional questions, we use FreeCiv as our simulator. FreeCiv is an open-source, turn-based strategy game based on the Civilization series (The Freeciv Project, 2026a), run through the CivRealm environment (Qi et al., 2024). Each world involves five auto-controlled civilizations on a 78 by 52 map. We played eight games to turn 60, paused them, then replayed each game for 1,000 unique "continuations" to turn 210 (Section A.2). The world report, about 4,900 tokens, gives each civilization's state at turn 60 and at every fifth turn before it, the diplomatic state of every pair and a log of the events so far (Section B).

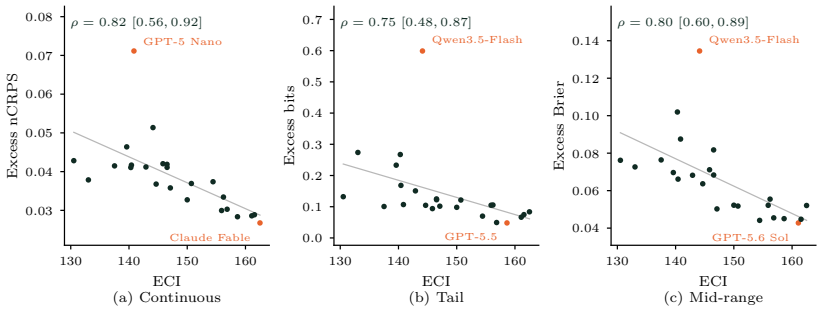


Figure 2: Micropolis forecasting scores against the Epoch Capabilities Index for 24 models; lower is better in every panel. (a) Continuous: excess nCRPS, 600 questions. (b) Tail questions ($0 < q \leq 5\%$): excess bits, 940 questions. (c) Mid-range questions ($5\% < q < 95\%$): excess Brier, 983 questions. $\rho = \rho(\text{ECI}, -\text{loss})$ and its interval are those of Table 2, and orange marks the best and worst model.

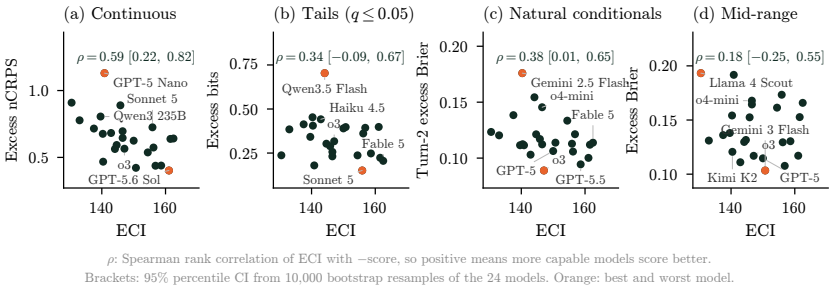


Figure 3: FreeCiv scores against the Epoch Capabilities Index for 24 models; lower is better in every panel. (a) Continuous: excess nCRPS, 300 questions. (b) Tail questions: excess bits, 300. (c) Natural conditionals: turn-2 excess Brier score against $p(Y \mid X)$, 400 cells. (d) Mid-range questions: excess Brier score, 750. ρ is sign-adjusted Spearman with its 95% percentile bootstrap over models (10,000 resamples), the numbers of Table 2; orange marks the best and worst model.

4.2 Results

Tail questions. On the 300 truth-selected tail questions, excess bits average 0.33 and correlate with ECI at $\rho = 0.34$ (Figure 3b). Mean forecasts are 0.13–0.34 against mean replay probability 0.02. All models lose to constant 0.05 on this subset, a reference chosen with knowledge of its probability range. Tail loss ranks are strongly associated with mean forecast level. Joint binary comparisons depend on the question mixture (Section C.3).

Natural conditionals. After the fact is revealed, turn-2 excess Brier averages 0.119 and correlates with ECI at $\rho = 0.38$ (Figure 3c). Mean gain over keeping the turn-1 forecast is near zero (-0.002), but this masks substantial heterogeneity: mechanical same-series questions (C1) lose 0.053 on average, while the other 344 cells gain 0.006; 7 of 24 models have positive overall mean gain. However, models usually update in the right direction—on the 185 cells where the conditional-related shift is above 0.05, the update is directionally correct 86% to 94% of the time. The models' ultimate failure is in sizing the update—update sizes vary widely by question group, and do not point in a specific direction across the board (Figure 10).

Continuous questions. On the 300 continuous questions, excess nCRPS averages 0.65 and correlates with ECI at $\rho = 0.59$ (Figure 3a), with the same sign at every horizon. Mean excess more than doubles from turn 90 to turn 210. The elicited 5th-to-95th percentile intervals contain 22%–77% of replay values. This undercoverage can reflect location error as well as insufficient width; it does not by itself identify the cause of the scores.

Mid-range questions. On the 750 mid-range questions, excess Brier averages 0.141 and correlates weakly with ECI ($\rho = 0.18$). Every model has negative mean forecast error on the mid-range set and loses to constant 0.5. Discrimination—the correlation between forecasts and replay truth—rises with ECI at $\rho = 0.72$. The truth-binned plot does not itself measure calibration (Figure 11). On all 1,100 selected binary questions with equal item weights, nine models beat universal 0.5 (Section C.3); the per-band comparisons do not imply uniformly poor mixed-set performance.

5 STARSIM: INTERVENTIONAL CONDITIONALS IN AN EPIDEMIC MODEL

To evaluate performance on interventional conditionals, we use Starsim as our simulator. Starsim is an agent-based disease model (Kerr et al., 2024). In this simulator, the world report specifies disease parameters, then asks models to forecast how a given vaccination program will change future infections (as compared to the counterfactual in which no vaccine is distributed).

5.1 WORLDS AND QUESTION SETS

We use four worlds, each a region of 5,000 agents in Starsim 3.3.4 (Starsim Team (Institute for Disease Modeling), 2026). Worlds differ only in transmission probability, giving $R_0 \approx 1.8$, 2.4, 3.0 and 3.9. For each world, we run 300 seeds and display the single run whose day-20 cumulative infections are closest to the median. The resulting world report gives the disease parameters, cumulative infections at five-day intervals through day 20, and the number infectious on day 20 (Section B.3). (Full parameters are in Section A.3).

We ask how many people will be newly infected between the end of day 20 and the end of day 40 or 60, excluding earlier infections that would dilute the vaccine's effect. Models return the 10th, 25th, 50th, 75th and 90th percentiles. In the interventional prompt, we add one sentence: "On day 21, a share of the population receives a vaccine that reduces infection probability per exposure by 95 percent." We vary the share of the population that receives the vaccine between 25%, 50%, and 90%. In total, these conditions give 24 interventional items. To estimate the true probability, we perform 1,000 replays per item. Full elicitation details and exceptions are in Section A.3; full scoring details are in Section A.3.

5.2 RESULTS

Models with higher ECI scores tend to forecast epidemic outcomes more accurately under vaccination (Figure 4). Across 23 models, the correlation between ECI and negative excess score is $\rho = 0.67$ $[0.34, 0.85]$, where lower excess scores indicate better forecasts and the interval reflects uncertainty from resampling models. At 25%, 50%, and 90% vaccination coverage, the correlations are 0.70, 0.48, and 0.25, respectively, with mean normalized excess scores of 1.08, 0.69, and 0.34. Higher ECI also predicts better forecasts without vaccination: $\rho = 0.75$ $[0.49, 0.88]$ across 24 models. Table 21 reports uncertainty from resampling models separately from uncertainty due to the finite number of simulation replays.

Forecasts under vaccination are less accurate at the longer horizon when scored in people rather than normalized units (Figure 8). Mean excess CRPS rises from 153 people at day 40 to 191 at day 60, with 19 of 23 models receiving worse scores. Among the 17 models shared between the two benchmarks, better performance here is associated with better overall ForecastBench performance ($\rho = 0.45$ $[-0.14, 0.88]$). However, the interval includes zero (Table 21).

6 VALIDATION AGAINST A CAPABILITY INDEX

If our benchmark scores measure skill, we expect them to correlate with broad capability measures such as ECI. To validate this expectation, we examine rank correlations between ECI and our score measures. We find every sign-adjusted correlation is positive, with seven of the nine confidence intervals excluding zero (Table 2). Correlations are stronger for Micropolis (0.75–0.82) and Starsim (0.67–0.75) and weaker for FreeCiv (0.18–0.59). The two confidence intervals that include zero are FreeCiv mid-range (0.18) and tail (0.34). See Section 4.2 for some observations on the source of these weaker FreeCiv numbers. The combined score θ_m, a per-model term fitted to the log losses of

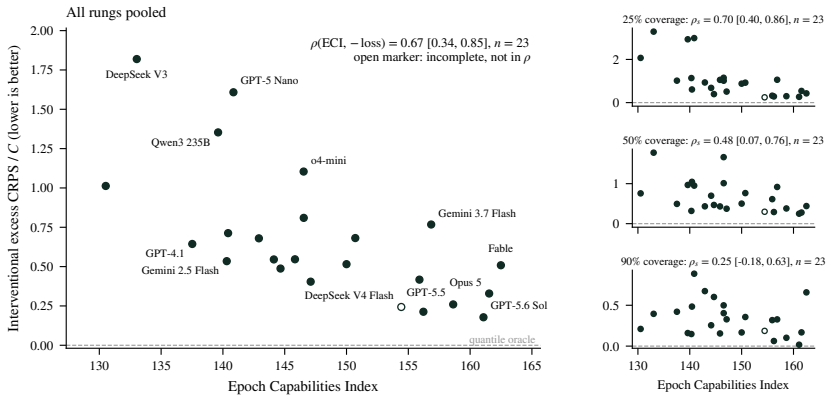


Figure 4: Accuracy of forecasts under vaccination in Starsim versus the Epoch Capabilities Index. Scores are normalized excess CRPS averaged across both forecast horizons; lower is better. The main panel combines all three vaccination coverage levels, while the side panels show each separately. Saved forecasts are rescored using corrected replays from the same full simulation state at day 20. Each panel reports Spearman's $\rho_s = \rho(\text{ECI}, -\text{loss})$ and a 95% interval from 2,000 bootstrap resamples of models. Positive correlations indicate that more capable models make more accurate forecasts. DeepSeek V4 Flash (open marker) answered 21 of 24 questions and is excluded from all reported correlations.

Table 2: Spearman correlation ρ between ECI and each score over 24 models, using $\rho(\text{ECI}, -\text{loss})$ so that a positive value means more capable models score better; 95% CI: percentile bootstrap over models; p: two-sided.

Simulation	Score	n	ρ	95% CI	p
Micropolis	Excess Brier, mid-range	24	0.80	[0.60, 0.89]	<0.001
	Excess bits, tail	24	0.75	[0.48, 0.87]	<0.001
	Excess nCRPS	24	0.82	[0.56, 0.92]	<0.001
Starsim	Excess nCRPS, interventional	23	0.67	[0.34, 0.85]	<0.001
	Excess nCRPS, continuous	24	0.75	[0.49, 0.88]	<0.001
FreeCiv	Excess nCRPS	24	0.59	[0.22, 0.82]	0.002
	Excess bits, tail	24	0.34	[−0.09, 0.67]	0.107
	Excess Brier, natural conditional	24	0.38	[0.01, 0.65]	0.068
	Excess Brier, mid-range	24	0.18	[−0.25, 0.55]	0.412

all nine scores (Section C.5), correlates with ECI at 0.86 [0.69, 0.93]. This is a higher correlation than any in Table 2, which is partly expected: combining the nine scores cancels some of their noise.

The association with ECI is robust to changes in prompt design (Sections D.1 and D.2), forecast horizon (0.71–0.82 for both binary scores at every horizon, computed from Table 15), question selection (Section C.1), and to how the nine scores are combined (Section C.5). Individual model ranks move more under these changes, for instance by up to 13 places across prompt configurations (Section D.2). This is likely in part because separation between model scores is often of the order of their confidence intervals over questions (Table 23), making the ranking unstable. The upshot is that our benchmark separates more capable models from weaker ones, but a single model's rank is uncertain.

In Section C.4 we conduct a similar analysis with ForecastBench overall (Karger et al., 2025a) in place of ECI, for the 17 models that have a ForecastBench score.

7 Related work

Forecasting benchmarks. Benchmarks of LLM forecasting draw their questions from tournaments, markets and the news. ForecastQA and Autocast assembled resolved questions (Jin et al., 2021; Zou et al., 2022), ForecastBench generates new questions over time and scores models beside human forecasters (Karger et al., 2025a), and Prophet Arena scores live market questions (Yang et al., 2026b). Each resolves a question once, against a single realized outcome. Alternatives pose already-resolved questions with evidence frozen before resolution (FutureSearch et al., 2025; Ye et al., 2026), or check consistency without a truth value (Paleka et al., 2025b). Moreover, live date-filtered retrieval can leak post-cutoff information (El Lahib et al., 2026), and the related study did not test frozen snapshots. A 2026 survey (Xu et al., 2026) includes the FreeCiv-only design of Lee et al. (2026), which resolved each question against a single hidden continuation. A relevant human reference class comes from forecasting tournaments (Tetlock, 2005; Mellers et al., 2014), which could not yet assess accuracy on low-probability questions about long-run risks (Karger et al., 2025b). We retain the scoring rules of the forecasting literature but resolve against replays of a privately stored game state. This means we can use fresh question sets with unreleased answers, which reduces the likelihood of contamination.

Scoring rules and rare events. The uncertainty term of the Brier score (Murphy, 1973) depends partly on the probability of the event occurring, and Paleka et al. (2025a) show that averaging raw scores across base rates hides rare-event skill. Lerch et al. (2017) also showed that scoring only the cases in which an extreme event occurred rewards forecasters who overstate its probability, so we avoid this failure mode by selecting tail questions on q before any specific outcome is drawn. Because calibration on overall questions does not translate well to calibration on rare events (Allen et al., 2024; Merkle & Steyvers, 2013), we also score tail questions in excess bits.

Causal, conditional and updating evaluation. Causal benchmarks for LLMs take their truth from a symbolic oracle over a small causal graph (Jin et al., 2023; 2024; Kasetty et al., 2024) or let an agent intervene in a small generative model (Gandhi et al., 2025; Yang et al., 2026a). InterveneBench evaluates causal study designs and intervention reasoning against expert-verified annotations (Shi et al., 2026), without scoring predicted outcome distributions. Tetlock et al. (2017) argued that policy conditionals are most critical to decision makers, but that they cannot be resolved in the real world. IARPA's FOCUS program proposed in 2017 to score counterfactual forecasts by rerunning a simulated world, with forecasters reading reports of AI-played Civilization games (Lehner, 2017; Wiblin & Harris, 2019). An interventional conditional follows the same idea with a proper rule, paired replays of a disease model and a language model as the forecaster. EvolveCast updates LLM forecasts on real news and can compare each update only with those of human forecasters (Yuan et al., 2025). Other work tests whether updates are consistent with Bayes' theorem without an event-level truth (Imran et al., 2025; He et al., 2025), and BayesBench scores beliefs against exact posteriors under a specified likelihood (Samanta et al., 2026). Binary conditional forecasting is established (McCracken et al., 2022); here we evaluate language-model updates after revealed facts against conditional probabilities estimated by replay.

8 Discussion

Across all three simulators, models with higher ECI generally score better on ForecastBench-Sim (Table 2), suggesting that these tasks capture model capability, at least to a degree. However, stronger models do make forecasting errors, and new information does not consistently improve their predictions: in FreeCiv, conditional updating produces inconsistent gains. Compared to existing benchmarks like ForecastBench (Karger et al., 2025a), ForecastBench-Sim allows us to examine these errors against estimated outcome distributions.

However, unlike ForecastBench, ForecastBench-Sim does not have a clear relationship to the real world, raising an important question: to what degree does success on ForecastBench-Sim reflect forecasting ability (versus puzzle-solving skill)? After all, our simulators, however complex, are necessarily simpler than the real world, and the specific simulators we chose are likely in models' corpora. Even if they were not, highly capable models could, in theory, download the source code for these simulators and examine them. One direction for future work, then, is to withhold selected

simulator rules from the models, forcing models to infer how the world works from limited observation. This would better capture one dynamic of real-world forecasting; whether it better predicts real-world performance remains to be seen.

On the topic of simulator inclusion in model corpora, publishing this benchmark creates potential contamination issues for future runs. Models may encounter this paper, or the released code, and use published information about base rates to improve their scores. Simulation gives us some room to respond to this challenge: future evaluators might change the number of players in FreeCiv or the disease parameters in Starsim, which changes the underlying outcome distributions. Our planned ongoing evaluations with ForecastBench-Sim will therefore vary key parameters of the simulated worlds. While this variation does not eliminate the risk of gaming, it will allow us to at minimum test whether models adapt to new conditions, or whether they reproduce patterns from our published work.

9 Conclusion

ForecastBench-Sim uses simulated worlds to evaluate models' forecasting ability against estimated outcome distributions. Across three simulated worlds, more capable models (as measured by ECI) generally score better on our benchmark. It remains an open question whether these skill rankings carry over to real-world forecasting, and the benchmark can be used as a controlled setting to study LLM forecasting alongside real-world evaluation.

AI use statement

We used language-model assistants for drafting and literature retrieval in the paper-writing process, as well as to produce statistical summaries and figures. In the experimental setting, models conducted code generation and simulation and elicitation monitoring. Models also helped construct the Micropolis knowledge test (Section D.3). Models also inspected prior notes about the research setup, audited scoring and provenance to ensure the paper's numbers were attributed correctly, and implemented the retrospective Starsim continuous repair. The authors are responsible for the final analyses, citations and claims.

Reproducibility statement

The code for the benchmark and paper analyses is available in two anonymized repositories, the benchmark repository and the analysis repository. Data files and supporting materials that are not already included in these repositories are listed in Section E. On publication, we will release every question set, along with the world reports, full prompts, raw model responses, and scoring code.

To give a quick overview of reproducibility aids: Section 2.2 defines all the relevant scoring rules used in the paper. Section A describes each simulation, as well as listing the tested models with details on reasoning settings (and cost). Section B contains reproductions of the prompts, question templates in full, and excerpts of the world reports too. Section C reports the scores included in every figure, split by model and horizon. Section E indicates which scripts score the outputs and produce each figure.

References

Sam Allen, Jonathan Koh, Johan Segers, and Johanna Ziegel. Tail calibration of probabilistic forecasts, 2024. URL https://arxiv.org/abs/2407.03167.

Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi: 10.1175/1520-0493(1950)078⟨0001:VOFEIT⟩2.0.CO;2.

Ali El Lahib, Ying-Jieh Xia, Zehan Li, Yuxuan Wang, and Xinyu Pi. Temporal leakage in search-engine date-filtered web retrieval: A retrospective forecasting case study. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2026. URL https://arxiv.org/abs/2602.00758. To appear.

Epoch AI. Epoch Capabilities Index (ECI). https://epoch.ai/eci, 2026. Leaderboard and documentation (https://epoch.ai/data/eci-documentation); public code at https://github.com/epoch-research/eci-public.

Forecasting Research Institute. Ai models have likely reached parity with superforecasters on ForecastBench. https://forecastingresearch.substack.com/p/ai-models-have-likely-reached-parity, July 2026. Forecasting Research Institute Substack, published July 16, 2026; accessed September 2026.

FutureSearch, Jack Wildman, Nikos I. Bosse, Daniel Hnyk, Peter Mühlbacher, Finn Hambly, Jon Evans, Dan Schwarz, and Lawrence Phillips. Bench to the future: A pastcasting benchmark for forecasting agents, 2025. URL https://arxiv.org/abs/2506.21558.

Kanishk Gandhi, Michael Y. Li, Lyle Goodyear, Agam Bhatia, Louise Li, Aditi Bhaskar, Mohammed Zaman, and Noah D. Goodman. BoxingGym: Benchmarking progress in automated experimental design and model discovery, 2025. URL https://arxiv.org/abs/2501.01540.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437.

Danny Halawi, Fred Zhang, Yueh-Han Chen, and Jacob Steinhardt. Approaching human-level forecasting with language models. In *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024. URL https://arxiv.org/abs/2402.18563.

Zhonghao He, Tianyi Qiu, Hirokazu Shirado, and Maarten Sap. Martingale score: An unsupervised metric for Bayesian rationality in LLM reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://arxiv.org/abs/2512.02914.

Anson Ho, Jean-Stanislas Denain, David Atanasov, Samuel Albanie, and Rohin Shah. A rosetta stone for AI benchmarks, 2025. URL https://arxiv.org/abs/2512.00193.

Don Hopkins, Electronic Arts, and Maxis. Micropolis: Open-source city simulation based on SimCity Classic. https://github.com/SimHacker/micropolis, 2008. SimCity Classic source released under GPL-3 by Electronic Arts for the One Laptop Per Child project in January 2008; modern C++/WebAssembly engine at https://github.com/SimHacker/MicropolisCore.

Sohaib Imran, Ihor Kendiukhov, Matthew Broerman, Aditya Thomas, Riccardo Campanella, Rob Lamb, and Peter M. Atkinson. Are LLM belief updates consistent with Bayes' theorem? In *ICML 2025 Workshop on Assessing World Models*, 2025. URL https://arxiv.org/abs/2507.17951.

Woojeong Jin, Rahul Khanna, Suji Kim, Dong-Ho Lee, Fred Morstatter, Aram Galstyan, and Xiang Ren. ForecastQA: A question answering challenge for event forecasting with temporal text data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4636–4650, 2021. doi: 10.18653/v1/2021.acl-long.357. URL https://aclanthology.org/2021.acl-long.357/.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. CLadder: Assessing causal reasoning in language models. In *Advances in Neural Information Processing Systems 36 (NeurIPS)*, 2023. URL https://arxiv.org/abs/2312.04350.

Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona Diab, and Bernhard Schölkopf. Can large language models infer causation from correlation? In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2306.05836.

Ezra Karger, Houtan Bastani, Yueh-Han Chen, Zachary Jacobs, Danny Halawi, Fred Zhang, and Philip E. Tetlock. ForecastBench: A dynamic benchmark of AI forecasting capabilities. In *International Conference on Learning Representations (ICLR)*, 2025a. URL https://arxiv.org/abs/2409.19839.

Ezra Karger, Josh Rosenberg, Zachary Jacobs, Molly Hickman, and Phillip E. Tetlock. Subjective-probability forecasts of existential risk: Initial results from a hybrid persuasion-forecasting tournament. *International Journal of Forecasting*, 41(2):499–516, 2025b. doi: 10.1016/j.ijforecast.2024.11.008.

Tejas Kasetty, Divyat Mahajan, Gintare Karolina Dziugaite, Alexandre Drouin, and Dhanya Sridhar. Evaluating interventional reasoning capabilities of large language models. *arXiv preprint arXiv:2404.05545*, 2024. URL https://arxiv.org/abs/2404.05545.

Cliff Kerr, Robyn Stuart, Romesh Abeysuriya, Paula Sanz-Leon, Jamie Cohen, and Daniel Klein. Starsim: A flexible framework for agent-based modeling of health and disease. Presentation, SciPy 2024 (Python in Science Conference), Tacoma, WA, 2024. URL https://zenodo.org/records/13942359.

Jaeho Lee, Nick Merrill, and Ezra Karger. ForecastBench-Sim: A simulated-world forecasting benchmark, 2026. URL https://arxiv.org/abs/2606.18686. ICML 2026 Workshop on AI Forecasting.

Paul E. Lehner. Forecasting counterfactuals in uncontrolled settings (FOCUS): Proposers' day overview briefing. Intelligence Advanced Research Projects Activity (IARPA), Proposers' Day Briefing, October 19, 2017, October 2017. URL https://www.iarpa.gov/images/PropsersDayPDFs/FOCUS/FOCUS-Proposers-Day-Overview-Briefing_FINAL.pdf. Program overview: https://www.iarpa.gov/research-programs/focus.

Sebastian Lerch, Thordis L. Thorarinsdottir, Francesco Ravazzolo, and Tilmann Gneiting. Forecaster's dilemma: Extreme events and forecast evaluation. *Statistical Science*, 32(1):106–127, 2017. doi: 10.1214/16-STS588. URL https://arxiv.org/abs/1512.09244.

James E. Matheson and Robert L. Winkler. Scoring rules for continuous probability distributions. *Management Science*, 22(10):1087–1096, 1976. doi: 10.1287/mnsc.22.10.1087.

Michael W. McCracken, Joseph T. McGillicuddy, and Michael T. Owyang. Binary conditional forecasts. *Journal of Business & Economic Statistics*, 40(3):1246–1258, 2022. doi: 10.1080/07350015.2021.1920960.

Barbara Mellers, Lyle Ungar, Jonathan Baron, Jaime Ramos, Burcu Gurcay, Katrina Fincher, Sydney E. Scott, Don Moore, Pavel Atanasov, Samuel A. Swift, Terry Murray, Eric Stone, and Philip E. Tetlock. Psychological strategies for winning a geopolitical forecasting tournament. *Psychological Science*, 25(5):1106–1115, 2014. doi: 10.1177/0956797614524255.

Edgar C. Merkle and Mark Steyvers. Choosing a strictly proper scoring rule. *Decision Analysis*, 10 (4):292–304, 2013. doi: 10.1287/deca.2013.0280.

Allan H. Murphy. A new vector partition of the probability score. *Journal of Applied Meteorology*, 12(4):595–600, 1973. doi: 10.1175/1520-0450(1973)012⟨0595:ANVPOT⟩2.0.CO;2.

Daniel Paleka, Shashwat Goel, Jonas Geiping, and Florian Tramèr. Pitfalls in evaluating language model forecasters, 2025a. URL https://arxiv.org/abs/2506.00723.

Daniel Paleka, Abhimanyu Pallavi Sudhir, Alejandro Alvarez, Vineeth Bhat, Adam Shen, Evan Wang, and Florian Tramèr. Consistency checks for language model forecasters. In *International Conference on Learning Representations (ICLR)*, 2025b. URL https://arxiv.org/abs/2412.18544.

Siyuan Qi, Shuo Chen, Yexin Li, Xiangyu Kong, Junqi Wang, Bangcheng Yang, Pring Wong, Yifan Zhong, Xiaoyuan Zhang, Zhaowei Zhang, Nian Liu, Wei Wang, Yaodong Yang, and Song-Chun Zhu. CivRealm: A learning and reasoning odyssey in civilization for decision-making agents. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2401.10568.

Ankur Samanta, Akshayaa Magesh, Tal Lancewicki, Ayush Jain, Youliang Yu, Paul Sajda, Kaveh Hassani, Aditya Modi, Daniel R. Jiang, and Yonathan Efroni. BayesBench: Evaluating LLM belief trajectories under multi-turn evidence accumulation, 2026. URL https://arxiv.org/abs/2606.30850.

Shaojie Shi, Zhengyu Shi, Lingran Zheng, Xinyu Su, Anna Xie, Bohao Lv, Rui Xu, Zijian Chen, Zhichao Chen, Guolei Liu, Naifu Zhang, Mingjian Dong, Zhuo Quan, Bohao Chen, Teqi Hao, Yuan Qi, Yinghui Xu, and Libo Wu. InterveneBench: Benchmarking LLMs for intervention reasoning and causal study design in real social systems. *arXiv preprint arXiv:2603.15542*, 2026. URL https://arxiv.org/abs/2603.15542.

Starsim Team (Institute for Disease Modeling). Starsim: Agent-based disease modeling framework. https://github.com/starsimhub/starsim, 2026. Software, MIT license. Project site https://starsim.org/.

Philip E. Tetlock. *Expert Political Judgment: How Good Is It? How Can We Know?* Princeton University Press, Princeton, NJ, 2005.

Philip E. Tetlock, Barbara A. Mellers, and J. Peter Scoblic. Bringing probability judgments into policy debates via forecasting tournaments. *Science*, 355(6324):481–483, 2017. doi: 10.1126/science.aal3147.

The Freeciv Project. Freeciv: Free and open source empire-building strategy game. https://github.com/freeciv/freeciv, 2026a. GPL-2.0; project homepage https://www.freeciv.org/.

The Freeciv Project. Freeciv-web. https://github.com/freeciv/freeciv-web, 2026b. Browser client/server used by CivRealm.

Robert Wiblin and Keiran Harris. Accurately predicting the future is central to absolutely everything. Phil Tetlock has spent 40 years studying how to do it better. 80,000 Hours Podcast, episode page and transcript, published June 28, 2019, June 2019. URL https://80000hours.org/podcast/episodes/philip-tetlock-forecasting-research/. Accessed September 2026.

Xiaogang Xu, Jiaqi Tang, Jianmin Chen, Yingying Yan, Zhenchao Tang, Xiangxin Zhou, Xiaobin Hu, Wei Wei, Jinfeng Wu, Qifeng Chen, Lu Zhou, Jiafei Wu, Zhe Liu, Jianwei Yin, and Weimin Zheng. LLM-based agents for forecasting and prediction: Methods, training, evaluation, and applications, 2026. URL https://arxiv.org/abs/2608.23058.

Junlin Yang, Dylan Zhang, Xiangchen Song, Qirun Dai, Xiao Liu, Yuen Chen, Aniket Vashishtha, Jing Shi, Chenhao Tan, and Hao Peng. CausaLab: A scalable environment for interactive causal discovery toward AI scientists, 2026a. URL https://arxiv.org/abs/2605.26029.

Qingchuan Yang, Simon Mahns, Sida Li, Anri Gu, Jibang Wu, and Haifeng Xu. LLM-as-a-prophet: Understanding predictive intelligence with Prophet Arena. In *International Conference on Learning Representations (ICLR)*, 2026b. URL https://arxiv.org/abs/2510.17638.

Xiao Ye, Jacob Dineen, Evan Zhu, Shijie Lu, Kevin Song, and Ben Zhou. Hindcast: Replaying prediction markets to evaluate LLM forecasters, 2026. URL https://arxiv.org/abs/2607.14051.

Zhangdie Yuan, Zifeng Ding, and Andreas Vlachos. Do language models update their forecasts with new information? *arXiv preprint arXiv:2509.23936*, 2025. URL https://arxiv.org/abs/2509.23936.

Andy Zou, Tristan Xiao, Ryan Jia, Joe Kwon, Mantas Mazeika, Richard Li, Dawn Song, Jacob Steinhardt, Owain Evans, and Dan Hendrycks. Forecasting future world events with neural networks. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, 2022. URL https://arxiv.org/abs/2206.15474.

A SIMULATIONS AND EVALUATION PROTOCOL

Replay truths condition on the full saved simulator state, while forecasters see a report; zero excess is therefore an oracle reference, not necessarily attainable from the report alone. Section 2 defines ground-truth probability q and the scoring terms; Section B gives the forecasting prompts and templates. Throughout, a horizon T is the turn or day at which a question resolves, and "the window" means the turns after the snapshot up to and including T.

A.1 MICROPOLIS

Program. Micropolis is the open-source release of the SimCity Classic engine (Hopkins et al., 2008). We use MicropolisCore, its C++ and WebAssembly port, through the port's command-line runner. The runner loads a city from a built-in save and advances the engine with no player input and with random disasters enabled. One turn is 16 engine ticks, and 48 turns make one game year. Our fork fixes an out-of-bounds crash in the engine and a reseeding of its random number generator from the clock on every load; with the fixes, the city, the seed and the number of turns determine a run completely. Transport, police and fire funding are fixed at 100%, and the treasury has no effect on the city. We therefore ask no question about money, and the world report withholds the city's funds. The run used our MicropolisCore fork, commit `4fe3a72`.

Cities, snapshots and replays. We use 15 of the 19 populated built-in cities (Table 3). We run each city from seed 42 and pause it at turns 960 and 1,440, game years 20 and 30, which gives 30 snapshots. From each snapshot we run 1,000 replays. The runner takes a byte copy of the engine's memory at the snapshot, restores it for every replay, reseeds the random number generator and runs to the requested horizon. A replay's output depends only on the city, the seed used to reach the snapshot, the snapshot turn, the horizon and the replay seed. The horizons are 144, 240, 336 and 480 turns past the snapshot, or 3, 5, 7 and 10 game years. One replay of 480 turns takes about 110 ms on one core, so 1,000 replays of one snapshot take under two minutes. The world report for a snapshot gives the current state, the reported metrics every 24 turns since the start, the counts of rubble, burning and road tiles, and the disasters and advisories so far (Section B.1 reproduces one).

Binary questions. Section B.1 lists the 25 questions we ask at every snapshot and horizon, as they appear in the prompt. Every question concerns the window from the snapshot turn S, exclusive, to the horizon turn H, inclusive. Event questions count the engine's own messages in the window. One plane crash emits both a plane and a helicopter message, and we count only the first. Population, class, score, pollution and tile counts are read from the engine's per-turn statistics at the end of the named turn, and a yearly checkpoint is a turn that is a multiple of 48. The city classes, in increasing order, are Village, Town, City, Capital, Metropolis and Megalopolis. B8 resolves YES if the largest population at a yearly checkpoint in the window exceeds the largest at any yearly checkpoint up to S. The analysis then assigns the 3,000 question instances per model (25 questions, 30 snapshots, 4 horizons) to sets by the realized q: 940 tail instances with $0 < q \leq 0.05$ and 983 mid-range instances with $0.05 < q < 0.95$ per model. The 826 instances with $q = 0$ and the 251 with $q \geq 0.95$ are in neither set; they enter only the check in Section 3.1 that pools all 3,000 under one scoring rule.

Continuous questions. The continuous set asks for the value at the horizon of population, average traffic, average pollution, average crime and average land value. With five metrics, four horizons and 30 snapshots, that is 600 questions per model. The model returns the 10th, 25th, 50th, 75th and 90th percentiles. The score is the excess nCRPS of Section 2.2: the CRPS of those five percentiles against the replay distribution, less the floor the replays themselves score, divided by a scale computed per city and metric — that metric's mean over the city's turns up to its first snapshot, floored at 10,000 for population, 10 for traffic and 40 for pollution, crime and land value.

Elicitation. A snapshot's binary questions are split across two prompts of 50 numbered questions, ordered by horizon; its continuous questions share one prompt of 20 (5 metrics at 4 horizons). The prompt opens "You are an expert superforecaster, familiar with the work of Tetlock and others", describes the no-mayor setting, and names the scoring rule: the Brier score for the binary set and the CRPS for the continuous set. It ends with a fixed answer block, one line per question. The full prompt details are given in Section B.1. We send each prompt once. Reasoning requests and uncontrolled-model exceptions are listed in Section A.5.

Table 3: The 15 Micropolis cities and their population when loaded. Each city is paused at turns 960 and 1,440 and replayed 1,000 times from each snapshot.

City	Population	City	Population	City	Population
haight	214,260	kamakura	90,380	senri	20,700
joffburg	178,680	yokohama	81,380	wetcity	19,960
kobe	153,120	freds	63,440	southpac	15,240
kowloon	137,640	bruce	14,440	linecity	12,780
kyoto	108,820	happisle	105,420	deadwood	6,580

Scale and selection scope. All Micropolis prompt variants use the same city-and-metric normalization table, computed from history through the main run's first snapshot at year 20, with the stated floors. The scale is not recomputed for each variant's age. Binary prompts name Brier scoring while headline tail evaluation uses excess log loss; both are proper for truthful probabilities, but we have not tested whether models respond the same way when the named rule changes.

Continuous-answer range check. For all Micropolis continuous results, we mark a set of percentiles as unparsed if its median is outside the engine's possible range: 0–255 for average pollution, crime and land value, 0–612 for average traffic, or negative for population. This post-hoc rule removed no answers from the main run; its motivation and effect on the prompt variants are reported in Section D.2.

A.2 FreeCiv

Program. FreeCiv is an open-source, turn-based strategy game (The Freeciv Project, 2026a). We run its server inside the Freeciv-web container (The Freeciv Project, 2026b) through the CivRealm environment (Qi et al., 2024). Each game has five civilizations, created by the server's fill option and played by the game's AI at the "hard" level. The one connected player is switched to AI control at the same level and serves only as the observer through which the environment reads the game; no human or model plays. The map is 78 by 52 tiles. Barbarians run at the server's default settings, under which raids begin around turn 60. The container is `civrealm/freeciv-web:1.4` (January 16, 2024); its server reports version 3.2.91-dev, a Freeciv 3.3 development build, and every game uses the `classic` ruleset, as the headers of the saved games record.

Anchor games and replays. We played eight anchor games, seeds 7001, 7003, 7005, 7008, 7010, 7011, 7014 and 7022, to turn 60 and kept the server's saved game at every turn. From the turn-60 save of each anchor we ran 1,000 replays, seeds 5001 to 6000, to turn 210. The replays ran on a fleet of 27 cloud workers on September 2, 2026, 600 at a time, and finished in about three hours. We read every quantity used for resolution from the per-turn saved games of the replay, not from the client. A cross-check of 6,080,000 (turn, civilization) cells between the serialized series and an independent parse of the saved games found no mismatch. Five replays stopped at a fixed turn with every civilization alive and did so again when rerun. They are program failures, and we replaced each with a fresh seed on the same anchor (7001: seeds 5014 and 5841; 7005: 5439 and 5639; 7022: 5823; substitutes 6001 upward). We also replaced two replays in which the observer civilization was eliminated before turn 210 (7001 seed 5482 at turn 117; 7014 seed 5290 at turn 178), so that every replay reaches turn 210. That replacement is conditioned on an outcome. Replacing one of 1,000 replays changes an unconditional frequency by at most 0.001. Conditional frequencies use smaller reveal-specific denominators: with 100 qualifying replays, one replacement can change the conditional frequency by 0.01. These are bounds, not measured changes; the two originals are retained. A ninth anchor, seed 7016, lost its observer in 3.1% of replays, and we replaced it by seed 7011 before the draw. Replacing an anchor when its observer dies also makes anchor selection depend on survival. Changing one replay outcome can affect eligibility at the $q > 0$ tail boundary; its actual effect on selected items or scores has not been established. The corpus occupies 92 GB.

World report. We generate the report from the turn-60 saved game; it is 17.9 to 20.5 KB, about 4,900 tokens, per anchor. It lists the civilizations and, for each, the score, rank, government, treasury, population, technologies, wonders, cities, territory, units, military, science, trade, food and production at turn 60. It gives the diplomatic state, first-contact turn and embassy status of every

pair, whether each pair's territories touch, and each civilization's government changes. It samples the same quantities every five turns from turn 1. It ends with a log of every city founded, captured or destroyed, every diplomatic change, and every technology and wonder completed through turn 60. It contains no map and no research target. A forecaster therefore sees less than the saved game holds, and the ground truth is conditional on the full saved game.

Question sets. Every question is an instance of a family with parameters: a civilization, a pair, a quantity, a threshold, a count, a wonder, a technology or a government. We render its resolution criteria from the family and parameters, never from the question text. Table 5 lists the 23 binary families with the items each contributes to the mid-range set, the tail set, the mirror set and the natural-conditional cells. Table 6 lists the nine continuous families with their normalization constants, and Table 4 the items per anchor. The draw (version 1.8, September 8, 2026) combines the probability-based eligibility criteria and structural quotas described below. The mid-range set, which the released files call the bank, holds 750 items: 30 from each of five equal-width bands of q on $(0.05, 0.95)$ at each of the five horizons, balanced across families within a band. Its mean q is 0.501. The tail set holds 300 items with $0 < q \leq 0.05$, 60 per horizon; the 10th, 50th and 90th percentiles of q are 0.001, 0.013 and 0.048, and $q = 0$ is excluded by construction. The mirror set holds 50 items with $q \geq 0.95$, 10 per horizon. The continuous set holds 300 items, 60 per horizon: eight count families at about 6 per horizon each, and value-at-T questions on score, population, cities and territory at 3 per metric per horizon. Each continuous item carries its 1,000 replay values, and the model returns the 5th, 25th, 50th, 75th and 95th percentiles. The normalization constant of a family is the median over its items of the 5th-to-95th percentile range of the truth, rounded to one significant figure. The set includes 16 survival questions about elimination, so there is no blanket exclusion of all questions involving elimination. The question generator checks survival in every replay for the listed subject civilizations and designated world-state quantities. Survival and civil-war events are explicit exceptions, and the checks do not cover every world aggregate.

The natural-conditional set holds 400 cells at horizons 120 to 210, 100 per horizon, over 355 distinct questions. A cell pairs a question Y with one fact X about turns 61 to 90; its target $p(Y \mid X)$ is the frequency over the replays in which X held. A cell is admissible when at least 100 replays satisfy X, $0.02 < p(Y \mid X) < 0.98$, at most two facts are paired with one question, and at most eight cells share one event. We drew cells in five structural groups, with 36, 36, 56, 172 and 100 cells: A, irrelevant news; B, related weak news; C1, mechanical same-series; C2, persistence across series; D, inferential. In group A the fact concerns a civilization or pair that the question is not about. In group B the fact is a related but weak event on the state the question asks about: a capture, a government change, a wonder, a cease-fire or a first contact. In group C1 the fact is a change in the same quantity that the question asks about, for instance that a civilization lost three cities when the question asks about its city count at the horizon. In group C2 the fact is the same kind of change on a different series, for instance the same loss of cities when the question compares scores or territory. In group D the fact bears on the outcome only through inference, for instance a war that began when the question asks about war, alliance, civil war or conquest at the horizon. Within each group we drew cells with the strongest kinds of fact first (a loss of three or more cities, a fall of two or more in city count, a war begun), then weaker kinds. The revealed facts are of 14 kinds; the most common are that a civilization lost three cities (191 cells) and that a pair went to war (90 cells). Measured after the draw over all 1,000 replays, 174 cells have an estimated replay shift $|p(Y \mid X) - p(Y)|$ below 0.03, 96 between 0.03 and 0.08, 67 between 0.08 and 0.15, and 63 of at least 0.15. A further 124 turn-1 questions exist only to seed cells and enter no other set. The turn-2 prompt adds one sentence, "One fact about turns 61 through 90 has been revealed to you: …", and never discloses the turn of the event.

Elicitation. We send every prompt through OpenRouter with the report before the questions. On first-party hosts the prompt has two content parts with a cache breakpoint between them; elsewhere it is one string. There is no system message and no persona. The prompt says that answers will be evaluated with a proper scoring rule, without naming the rule. A grouped prompt of the run the paper reports has 8,759 to 17,575 tokens and asks for one line per question in a delimited block, a probability or the five percentiles; a natural-conditional prompt has 7,565 to 8,544 tokens and asks for the probability inside a tag (Section B). Every request uses provider-default sampling and an output limit of 32,768 tokens (16,384 in the run of September 9, with 24,000 for DeepSeek V4 Flash and 6,000 for Kimi K2). We set reasoning effort to the lowest level each model supports, with a 1,024-token budget for four models and no setting for four others (Table 10), and pin every model

to one host (Section A.5). Every stored row records the host, the prompt, completion and reasoning token counts, the cost, the finish reason, the reasoning parameter sent and, where returned, the reasoning text. We read a grouped reply from its last delimited block, matching each line to its question by the stated number, and fall back to numbered lines outside the block or, for percentiles, to five numbers on the line after a restated question. We read a natural-conditional reply from its tag block or, failing that, from a labeled line, which we flag, and ask once more when neither can be read. A missing binary answer scores as a forecast of 0.5; missing continuous and conditional answers are dropped. The no-news control continues the turn-1 exchange of 100 cells, stratified by group, with a sentence that says the game has continued and no new information is available.

The run of September 9, 2026, asked one question per prompt, 2,023 calls per model (1,524 turn-1, 400 turn-2 and 99 no-news), and produced 48,543 of 48,552 expected rows. Twenty-three models finished in 41 minutes; DeepSeek V3 took 85 minutes because its host limited it to about 8 calls per minute. Every reply from twenty-one models could be parsed. Kimi K2 (26 rows) and DeepSeek V4 Flash (27 rows) ran to the output limit with no answer, DeepSeek V3 lacks 9 turn-2 rows whose turn-1 answer was empty, and GPT-5 Nano at minimal effort needed the lenient parse on about 12% of its calls. DeepSeek V4 Flash ignores the effort setting and returns 10,000 to 15,000 reasoning tokens per call. The scored rows cost $393.83 across the 24 models, and a check at medium effort on 7 models and 200 mid-range items a further $34 (Section A.5). The paper uses this run's natural-conditional forecasts for 22 models and compares its other sets with the grouped run in Section D.1.

The run of September 20, 2026, is the one the paper reports. It asked the mid-range set, tail, mirror and continuous questions in groups, with the limits of the Micropolis run: 26 binary prompts of 36 to 50 questions and 18 continuous prompts of 13 to 20, one game per prompt and 44 prompts per model. We sent each prompt once and did not ask again for an answer we could not read. Every model answered at least 98.8 percent of its questions, and 99.9 percent on average. One prompt returned no readable answer at all (o4-mini). The natural-conditional arm kept the protocol of September 9, one question per prompt. We reuse its forecasts for the 22 models whose host did not change. DeepSeek V3 and DeepSeek V4 Flash moved to the hosts that Micropolis used and answered them again. DeepSeek V4 Flash ran at effort low, as in the other two simulations; the setting has no effect on it, and it returned about 13,275 reasoning tokens per grouped call. The grouped calls cost $26.10 and the two natural-conditional reruns $2.93.

Table 4: FreeCiv items per anchor game (draw v1.8). The sets are stratified on q, horizon and family, not on anchor, so anchors contribute unequally. Extra turn-1 questions seed natural-conditional cells and enter no set.

Set	7001	7003	7005	7008	7010	7011	7014	7022	Total
Binary bank	66	116	112	112	99	109	99	37	750
Tail	32	52	38	51	28	36	32	31	300
Mirror	1	8	13	9	2	8	5	4	50
Continuous	34	48	41	37	39	38	37	26	300
Natural-conditional cells	33	68	45	65	63	65	38	23	400
Extra turn-1 questions	9	18	13	23	22	19	8	12	124

Saved-answer coverage. Micropolis drops unparsed forecasts: 71,951 of 72,000 binary and 14,319 of 14,400 continuous answers are scored. Its exported continuous forecast file contains derived scores, not the original five quantiles. FreeCiv retains its distinct policy: one missing mid-range forecast among 18,000 model–question rows is imputed to 0.5; all 7,200 tail and 1,200 mirror rows parsed. Of 7,200 continuous rows, 37 are dropped, and 16 of 9,600 natural-conditional cells lack paired forecasts; all 2,976 extra turn-1 rows parsed. These counts do not account for all failed API attempts. The Micropolis binary missing-answer sensitivity is reported in Section C.1; continuous and FreeCiv missing-not-at-random sensitivities and complete raw-cache provenance remain unresolved.

A.3 STARSIM

Program. Starsim is a framework for models of disease spread among individuals (Kerr et al., 2024; Starsim Team (Institute for Disease Modeling), 2026). Both repositories fix the Starsim version at

Table 5: FreeCiv binary question families (draw v1.8). Each rule paraphrases the generated resolution criteria; the window is turns 61 through T inclusive, and "at T" means as recorded at the end of turn T. A capture is a city changing owner by force, counted per event; a city lost in a civil-war split is not a capture. Great wonders exclude small wonders such as a palace. Twenty-two families appear in the mid-range set; `NW4_survival` appears only in the tail set. Columns give items in the mid-range set (750), the tail set (300), the mirror set (50) and the natural-conditional cells (400). Table 11 gives an example question from each family.

Family	Resolves YES if …	Mid-range	Tail	Mirror	Nat. cond.
`EX_comparative`	civilization A's quantity (score, population, cities, territory, treasury, military units or technologies) at T is strictly greater than B's	53	9	2	82
`EX_government_at`	the civilization's government at T is the named form; anarchy counts as a form	46	22	4	6
`EX_tech_discovered`	the civilization discovers the named technology in the window	50	22	2	1
`NB1_threshold`	the civilization's quantity at T is at least the stated value	48	0	4	2
`NB1_value_threshold`	the civilization's quantity at T is at least the stated value	51	20	3	83
`NB4_drawdown`	the civilization's quantity falls below a stated share of its turn-60 value at the end of any turn in the window	44	20	2	80
`NB6_event`	any city is captured (or at least k captures occur in total) in the window	7	3	3	1
`NEW_civil_war`	the civilization splits in a civil war in the window	17	20	0	19
`NW1_war_at`	the pair is at war at T	49	15	3	25
`NW2_diplo`	the pair is in the named state (peace, alliance, or cease-fire or armistice) at the end of some turn in the window	44	21	1	26
`NW4_survival`	the civilization is eliminated in the window	0	16	0	0
`NW5_wonder`	any civilization completes the named wonder in the window	34	21	3	4
`S3_wars_at_T`	at least k of the ten pairs are at war at T	3	0	1	0
`S4_gov_change_count`	the civilization's government changes at least k times in the window, counting changes into and out of anarchy	32	6	4	3
`S5_civ_wonders`	the civilization completes at least k great wonders in the window	43	21	2	2
`S6_city_founding`	the civilization founds at least k new cities in the window	34	2	2	2
`S7_any_destroyed`	any city is destroyed in the window	2	0	0	0
`W1_wonder_race`	the named civilization completes the named wonder in the window	36	21	2	6
`W2_directed_conquest`	civilization A captures at least one city held by B in the window	49	20	3	22
`W3_capture_k`	the civilization captures at least k cities in the window	38	3	4	2
`W4_lose_k`	the civilization loses at least k cities to capture in the window	35	3	5	20
`W5_tech_lead`	the civilization knows strictly more technologies than each other civilization at T	23	15	0	5
`W6_peace_at`	the pair is at peace at T	12	20	0	9
Total (23 families)		750	300	50	400

Table 6: FreeCiv continuous question families (draw v1.8) with items and the constant that divides the CRPS of each family: the median over the family's items of the 5th-to-95th percentile range of the truth, rounded to one significant figure. `NC1_value_at_T` has one constant per metric.

Family	Resolves to …	Items	Constant
`NC1_value_at_T`	the civilization's score, population, number of cities or territory size at T	60	cities 30; population 70; score 70; territory 300
`P5_techs_at_T`	the number of technologies the civilization knows at T	31	9
`P6_world_techs`	the sum over the five civilizations of technologies known at T	20	20
`P1_civ_conquests`	the number of cities the civilization captures in the window	31	10
`P2_civ_losses`	the number of cities the civilization loses to capture in the window	31	10
`P3_civ_founds`	the number of new cities the civilization founds in the window	31	20
`NC5_world_captures`	the total number of city captures by all players in the window	32	50
`NC14_world_wonders`	the number of great wonders completed by all civilizations in the window	31	4
`S7_world_razings`	the number of cities destroyed in the window	33	20
Total (9 families)		300	

3.3.4. The simulation package (scenarios, report generator and paired runs) is `pandemic_world` in the ForecastBench-Sim repository at commit `65204aa` of August 20, 2026, which corrected the vaccination day (an earlier version started the vaccination on day 0; no reported run used it). The code that built the worlds, elicited the models and scored them is the `results/causal` part of the paper repository; the September 8 run is its commit `4d3a27a`, and the September 16 run adds the per-model reasoning setting and the host of each call to the same code.

World. Table 7 lists the parameters of the run. A region is 5,000 individuals on a random contact network of about six contacts per person per day under a susceptible–infectious–recovered model with a ten-day mean infectious period, no second infection once a person has recovered, and no deaths; 0.5 percent of individuals are infectious on day 0. Four worlds differ only in the per-contact transmission probability. For each we ran 300 seeds and displayed the run whose cumulative count at day 20 is closest to the median (seeds 27, 33, 62 and 47 for $\beta = 0.03$, 0.04, 0.05 and 0.065; 431, 1,129, 2,316 and 3,994 people ever infected at day 20). The original ground-truth construction pooled seeds whose day-20 count lay within 15 percent of the displayed run's: 131, 157, 202 and 290 of the 300, with vaccination runs using the same seeds. This historical construction is retained for binary scores only; continuous scores now use the retrospective fixed-state procedure below. The two-region worlds of the binary sets pair one displayed run with another at a value of β one step apart (Riverton at 0.04 with Southbay at 0.03, 0.05 with 0.04, 0.05 with 0.065, and 0.03 with 0.04); the fourth is the reverse of the first, so two worlds have Riverton ahead and two Southbay.

Items. The continuous template, `new_infections`, asks how many people will be newly infected between the end of day 20 and the end of day t, $t \in \{40, 60\}$, and is answered with the 10th, 25th, 50th, 75th and 90th percentiles; the revised truth is the distribution over 1,000 continuations from the fixed displayed state in the specified arm. The binary template, `cases_comparative`, asks whether Riverton will have more cumulative infections than Southbay at day t, and its ground truth is $q = (n_R n_S)^{-1} \sum_{i,j} \mathbf{1}(R_i > S_j)$ over the full Cartesian product of the matched regional pools; ties are NO. This is not equal-seed pairing. Within a region, control and vaccination reuse seeds and pre-day-21 histories. Across regions, seed IDs also reuse random roots, so the marginal samples are not independently seeded. The unconditional and 90 percent probabilities lie at the endpoints of the empirical distribution. Bootstrap intervals that collapse to a point do not establish certainty about the underlying population. No question about deaths was run: the model has no deaths. The five sets are the 8 unconditional continuous items, the same 8 under a vaccination at 25, 50 and 90 percent coverage (24 items), the 8 unconditional binary items, the same 8 under a vaccination at 90 percent, and the 8 of the *hold* set, in which the vaccinated region is the one ahead and the coverage is the

largest value, in steps of 5 percent, that leaves the ranking the same at both horizons (20, 10, 15 and 20 percent by world; the day-60 ground-truth values are then 0.97, 0.98, 0.01 and 0.03). Table 8 gives the templates and the intervention sentence.

Elicitation and records. The four binary worlds use 20,567, 31,714, 58,580 and 20,567 Cartesian comparisons, respectively, but these are dependent comparisons of reused regional trajectories, not independent joint trials. There are 104, 145, 202 and 104 seed IDs that are eligible in both regions. Restricting the comparison to those pairs changes how the regional outcomes are coupled; it does not repair the original target. Matching uses only day-20 cumulative counts within $\pm 15\%$, ignoring other displayed history and infectious counts. Requiring an exact match to the full report leaves only the displayed run in each cache. The same cached outcomes were used to select ranking flips and hold coverages, further limiting generalization.

Each prompt contains the provenance statement, the report, an intervention sentence where applicable, the question and the answer format, reproduced in full in Section B; one question per prompt, no role, no scoring rule named. Calls go through OpenRouter at default sampling settings, with the output limit at the larger of 16,000 tokens and the model's listed limit (65,536 at most), a 15-minute time limit, and up to four attempts when a response reaches the output limit or contains no answer that can be parsed. Five binary and three continuous repetitions were planned; medians over available saved answers are scored. The September 16 run sets each model's reasoning to its lowest level, with the 1,024-token budget for the four budget-only models (Section A.5); the September 8 run had set no reasoning parameter. The record of every call holds the response, the finish reason, the prompt, completion and reasoning token counts, the cost, the response time, the reasoning setting, and the host that served it. The September 16 records contain 5,045 retained scored responses at a recorded cost of $41, of which the unconditional binary, unconditional continuous and continuous interventional sets are 3,190 calls and $27; DeepSeek V4 Flash retained 111 of 216 planned responses across 49 of 56 items, including 21 of 24 continuous intervention items and 4 of 8 hold items; Kimi K2 retained 182 of 216 across all 56 items. The other 22 models have all planned responses. These are successful retained responses, not a verified count of all API attempts. The scorer treats a set as complete when every item has an answer, even if some items have fewer repetitions than planned.

Protocol details. The unconditional and interventional versions of an item are sent in separate contexts. We do not batch questions: each set has only eight items, and showing both horizons together would reveal that cumulative counts cannot decrease. Eight models returned no reasoning tokens at the lowest setting. DeepSeek V4 Flash returned about 18,000 per recorded call despite that setting; its incomplete continuous intervention and hold sets are excluded from full-set correlations; its complete eight-item binary 90 percent set is included. Kimi K2 returned no answer to many calls on the run day and completed every item on later attempts, some with fewer than three answers.

Scoring and normalization. Binary items are scored in *excess bits*, $\mathrm{KL}(P^* \,\|\, f)$ with f bounded to $[0.001, 0.999]$, the rule that FreeCiv applies to its tail questions (Section 2.2), here on both sides of one half: every P^* in the unconditional and 90 percent sets is exactly 0 or 1, and on the hold set the day-60 ground-truth values are 0.97, 0.98, 0.01 and 0.03, so no binary item falls in the mid-range where the excess Brier score $(f - P^*)^2$ applies. Continuous items are scored by a five-quantile pinball approximation to the CRPS (Matheson & Winkler, 1976), computed as in Section 2.2 and averaged over the 1,000 fixed-state continuation outcomes. The *floor* is the score of the ground-truth distribution's own quantiles at the same five levels, computed the same way; it is the minimum five-quantile score under the empirical replay distribution, and the excess CRPS is the score minus the floor. To put the two horizons on one scale we divide the excess by one constant per horizon, $C_{40} = 200$ and $C_{60} = 300$ people. These historical reference scales were the median 5th-to-95th percentile ranges under the original matched-seed truths, rounded to one significant figure; they are retained unchanged to make the retrospective comparison interpretable, not reestimated from the new distributions. The elicited quantiles stay at the 10th to 90th percentiles, where FreeCiv elicits the 5th to 95th (Section A). We compute the excess per item and then average over the items. Scores assess outcome distributions under the intervention. They combine baseline prediction and intervention response and do not isolate forecasting of the effect itself.

Retrospective fixed-state rescore. The continuous results reuse forecasts elicited before this repair. The original target pooled trajectories that approximately matched the displayed cumulative

count, rather than continuing the displayed state. We retain the original runs and retrospectively replace only their continuous truths and empirical quantile floors. This changes the statistical target: changes in excess reflect both changed outcomes and changed floors, not model deterioration on an unchanged task.

For each of four serialized end-of-day-20 anchors, we hold the complete agent, network, disease and pre-drawn recovery state fixed. We draw 1,000 future continuation seed sets per world, each with four arms (0, 25, 50 and 90 percent coverage): 16,000 branches in total. Horizons 40 and 60 share a branch, and arms share future random roots within a seed set. The target is $P(Y \mid S_{20}, a)$ for a fixed full state S_{20} and intervention a, not $P(Y \mid \text{report}, a)$. Report-only conditioning would require a distribution over compatible latent states. On day 21, vaccination is Bernoulli uptake among all 5,000 living agents, including infected and recovered agents, with probability equal to coverage and 0.95 leaky efficacy; it is neither exact-size nor susceptible-only vaccination. New infections subtract the exact displayed cumulative baseline. Checks verify timing, recipients, unchanged pre-snapshot histories, paired arms and $0 \leq Y_{40} \leq Y_{60} \leq 5000 - C_{20}$.

Sample size, continuation seeds and analysis rules were frozen before score inspection. A corrected recipient-checker discrepancy at coverage 0.9 affected no simulator behavior, seeds, retained outcomes or normalization; the stopped attempt and amendment remain in the production record. Binary truths and binary scores were not repaired by this continuous run; binary results are retained only as appendix diagnostics and excluded from headline validation and the nine-cell combined score.

No forecasts were newly elicited or imputed. The numbers of saved calls relative to planned calls are 567/576 (unconditional), 552/576 (25%), 558/576 (50%) and 556/576 (90%). There are respectively 6, 14, 11 and 11 model–item cells below the planned three repetitions, including three wholly missing interventional items for DeepSeek V4 Flash. The median at each elicited quantile uses every saved answer; repeated answer values are retained, and no duplicate repetition keys were found. The panels of models with an answer to every item contain 24 baseline models and 23 interventional models, including models with fewer repetitions. Neither uncertainty calculation below captures this elicitation uncertainty.

For continuous associations, we draw 2,000 paired bootstrap samples of the models while holding the empirical truths fixed. Separately, 1,000 replay bootstrap resamples draw continuation indices independently within each world and use the same sampled indices across arms and horizons, recomputing quantiles, floors, scores and correlations. Combined-score model intervals retain the current paper's 10,000 resamples of fitted model scores without refitting; replay intervals refit the scores while keeping the model panel fixed. These are separate sensitivity intervals, not a joint confidence interval. They do not resample worlds, latent anchor states, elicitation calls or external benchmark measurements. A replay correlation interval that collapses to a point means that the sampled rankings are stable; it does not establish the population association. Original truth estimates are held fixed in before/after comparisons. Table 21 reports both uncertainty sources.

A.4 Elicitation protocol by simulation

Table 9 lists how the three runs differ in role, named scoring rule, questions per prompt, reasoning effort, repeated calls and answer format. The shared structure, with the world report before the questions and an instruction to end with a probability or a set of quantiles, is described in Section 2.3. For the excess-bits score defined in Section 2, we bound forecasts to $[0.001, 0.999]$ before the logarithm, so at $q = 0$ or 1 the score has a floor of $-\log_2(0.999) \approx 0.00144$ bits. Micropolis and FreeCiv use the same positive-tail rule, $0 < q \leq 0.05$, and intend $0.05 < q < 0.95$ for mid-range, but the retained FreeCiv bank includes one $q = 0.95$ boundary item; its original 750-item scores are preserved (Section C.1). They sample the sets differently. FreeCiv draws its mid-range set evenly from five bands of q, while the Micropolis mid-range set keeps every instance that falls in the range, so its mean q, 0.28, is lower.

A.5 Model roster, reasoning settings, hosting and cost

Table 10 gives the 24 models, ECI (Epoch AI, 2026; Ho et al., 2025), available ForecastBench scores (Karger et al., 2025a), requested reasoning settings and mean observed reasoning-token counts in

Table 7: Starsim world parameters of the run. The binary sets use two such regions with transmission probabilities one step apart; the continuous sets use one.

Parameter	Value
Regions	one (continuous sets); Riverton and Southbay, isolated from each other (binary sets)
Individuals per region	5,000
Contact network	random, about 6 contacts per individual per day
Initial share infectious	0.005
Transmission probability per contact	one of $\{0.030, 0.040, 0.050, 0.065\}$ per world ($R_0 \approx 1.8$, 2.4, 3.0, 3.9)
Duration of infection	10 days, mean
Deaths	none
Seeds per world	300; the displayed run is the seed closest to the median day-20 count
Replays per item	continuous: 1,000 per fixed state and arm; binary: regional draws 157×131, 202×157, 202×290, 131×157
Vaccine	given on day 21; reduces a vaccinated person's probability of infection per exposure by 95 percent
Coverage	25, 50 or 90 percent (continuous); 90 percent, or the per-world *hold* value of 20, 10, 15, 20 percent (binary)
Snapshot	day 20
Horizons	day 40 and day 60
Report series	parameters as above; cumulative infections at days 0, 5, 10, 15, 20; number infectious at day 20

Table 8: Starsim question templates and the intervention sentence, as given to the models. Braces mark the fields filled per item.

Template	Question	Resolution
`cases_comparative`	Will Riverton have more cumulative infections than Southbay at day {T}?	Riverton's cumulative infections at day T strictly greater than Southbay's; share of matched seed pairs
`new_infections`	How many people in Riverton will be newly infected between the end of day 20 and the end of day {T}? (That is: cumulative infections at day {T} minus the {c20} already infected by day 20.)	cumulative infections at day T minus the fixed displayed count at day 20; distribution over fixed-state continuations
intervention	{Region} will run a vaccination campaign on day 21 (95% efficacy, {c}% coverage): on day 21, {c}% of {Region}'s population is vaccinated, and vaccination reduces a vaccinated person's probability of infection per exposure by 95%.	appended to the report as "PLANNED INTERVENTION"; the two-region form adds "{Other} has no planned intervention."

Table 9: Elicitation protocol by simulation. Every prompt places the world report before the question; sampling is at provider defaults. Section B gives prompt templates and report excerpts; Section 2.3 summarizes the shared structure.

	Micropolis	FreeCiv	Starsim
Questions per prompt	50 binary, 20 continuous	50 binary, 20 continuous; 1 for natural conditionals	1
Repeated calls per item	1	1	3 continuous, 5 binary planned; median of available responses
Reasoning effort	lowest supported requested where controllable; budget-only exceptions	same nominal requests	same nominal requests
Role assigned to the model	"expert superforecaster, familiar with the work of Tetlock and others"	none	none
Scoring rule named	Brier score (binary); CRPS (continuous)	"a proper scoring rule", not named	none; the opening statement names the task as probabilistic forecasting of simulator output
Continuous answer	p10, p25, p50, p75, p90	p5, p25, p50, p75, p95	p10, p25, p50, p75, p90
Conditional questions	none	second turn of the same conversation after one revealed fact	intervention described in the report: one sentence after the report, in a separate prompt from the unconditional question

Starsim. Seven models lack ForecastBench scores: Fable, Opus 5, GPT-5.6 Sol, GPT-5.6 Luna, Gemini 3.7 Flash, DeepSeek V4 Flash and Qwen3.5 Flash. Exact OpenRouter identifiers, requested-host and per-item billing tables are preserved in the private local supplement described in Section E.

FreeCiv and Micropolis request low effort for 11 models, minimal for five, a 1,024-token budget for four, and no control for four. Starsim used the same lowest-effort rule on September 16. The same requested setting does not imply the same amount of computation: DeepSeek V4 Flash ignores the request, and observed reasoning use varies widely (Table 10). FreeCiv's requested hosts on September 20 follow the committed Micropolis registry; the two DeepSeek conditional sets were rerun after their requested hosts changed. Micropolis's exported provider field comes from the model-ID prefix, not records of the host that actually served each call. Starsim recorded 14 providers for V4 Flash; V3 used DeepInfra (135 responses) and StreamLake (81), Llama DeepInfra (169) and Novita (47), and Fable Anthropic (213) and Azure (3). Routing differs within and across runs; host names do not establish the numerical precision or quantization used.

FreeCiv's scored inputs cost $174.15: $26.10 for 1,056 grouped prompts on September 20 plus $148.05 for conditionals, including $2.93 for the two reruns. Thus $29.03 is the additional September 20 expenditure represented in these records; most conditional calls are reused from September 9. The historical ungrouped run cost $393.83. Grouped-prompt costs are allocated across questions, and the same prompt can contribute to more than one set's prompt count. The conditional arm has 355 distinct turn-1 prompts, 400 reveal cells and 99 no-news prompts. Starsim's $40.99 covers 5,045 successful scored calls, not every failed or unrecorded attempt. Micropolis's recorded total is $32.31 over 2,160 calls, with 60 binary and 30 continuous prompts per model. These costs describe the recorded runs; they do not estimate the cost of refreshing the benchmark for all models.

Exploratory effort check. On 200 FreeCiv mid-range questions, raising seven models to the next reasoning level (medium or a 2,048-token budget) cost $4.83; five improved, with excess-Brier changes from -0.022 to $+0.014$. Every model remained worse than constant 0.5 (excess 0.081), with negative mean forecast bias at both levels (-0.27 to -0.10). Correlations with ECI, signed so that higher values mean better forecasting, were -0.46 and -0.39. Reasoning requests and batch composition both changed, so the comparison does not isolate the causal effect of reasoning effort.

Table 10: The 24 models with their Epoch Capabilities Index (ECI) and ForecastBench overall score (FB; – where none exists); the reasoning setting in FreeCiv and Micropolis, where "budget 1,024" is a 1,024-token reasoning budget; and the mean number of reasoning tokens per call that each model returned in Starsim at the setting of the September 16 run (all five sets, weighted by calls).

Model	ECI	FB	FreeCiv	Micropolis	Starsim tokens
Claude Fable	162.49	–	low	low	424
Claude Opus 5	161.54	–	low	low	342
GPT-5.6 Sol	161.08	–	low	low	1,168
GPT-5.5	158.63	61.1	low	low	1,262
Gemini 3.7 Flash	156.85	–	low	low	651
GPT-5.6 Luna	156.22	–	low	low	1,198
Claude Sonnet 5	155.90	59.9	low	low	435
DeepSeek V4 Flash 0731	154.43	–	low	low	18,044
Gemini 3 Flash Preview	150.71	59.1	minimal	minimal	0
GPT-5	150.00	60.5	minimal	minimal	0
o3	147.10	60.6	low	low	2,112
GPT-5.4 Nano	146.54	55.9	low	low	1,727
o4 Mini	146.53	60.9	low	low	827
GPT-5 Mini	145.83	59.7	minimal	minimal	0
Gemini 3.1 Flash Lite Preview	144.66	54.6	minimal	minimal	0
Qwen3.5-Flash	144.12	–	budget 1,024	budget 1,024	1,003
Claude Haiku 4.5	142.91	59.7	budget 1,024	budget 1,024	684
GPT-5 Nano	140.85	58.0	minimal	minimal	0
Kimi K2	140.42	57.6	none	none	3,263
Gemini 2.5 Flash	140.31	58.2	budget 1,024	budget 1,024	754
Qwen3 235B A22B	139.61	58.6	budget 1,024	budget 1,024	1,020
GPT-4.1	137.52	58.5	none	none	0
DeepSeek V3	133.03	57.6	none	none	0
Llama 4 Scout	130.52	56.6	none	none	0

B PROMPTS, QUESTION TEMPLATES AND WORLD REPORTS

This appendix reproduces the forecasting instructions, question templates and report excerpts for Micropolis and FreeCiv, followed by complete Starsim examples. In both simulations a prompt has the same structure: an opening instruction, the world report, the question or questions, and the answer format. The administered instructions and question templates are retained below; the availability of the source files is described in Section E. We wrapped lines longer than 80 characters at a space and indented the continuation by four spaces, and we truncated the two world reports where stated. Nothing else was changed.[1]

B.1 MICROPOLIS

Our harness combines a preamble, the world report under the heading `## Game report`, the numbered questions under `## Questions`, and an epilogue in which {n} is replaced by the number of questions. The binary and continuous runs each have their own preamble and epilogue, reproduced below. A binary prompt carries 50 questions, and the model answers them in one reply. A continuous prompt carries one snapshot's questions for the five metrics, ordered metric by metric (20 questions per prompt). The report omits the city's funds and no question asks about them. Both prompts open with the same role, "an expert superforecaster, familiar with the work of Tetlock and others", and name the scoring rule.

Binary preamble

```
You are an expert superforecaster, familiar with the work of
Tetlock and others. Your task is to forecast, for each of several yes/no
    questions about a
city in Micropolis, the open-source release of the original SimCity Classic
    simulation engine, the probability that the question resolves YES.

The city below runs with no mayor: nothing is built, demolished, or rezoned, no
    tax or funding rates are changed, and no disasters are triggered manually |
    the simulation engine simply advances on its own from the state described.
    Whether random disasters are enabled is stated in the game report.

Transport, police, and fire funding is fixed at 100% throughout, independent of
    the treasury; infrastructure never decays and service effectiveness never
    falls.

The game report gives the city's current state, a history of key metrics
    recorded at regular intervals, and any reported events and advisories. Form
your forecasts from what the report shows, combined with your knowledge of
the Micropolis engine.

The turns in the questions are counted in the same unit as the game report's,
and 48 turns make one game year. Every question asks only about what happens
strictly after the report's current turn, up to and including the turn it
names: events that already occurred by the current turn | including those
listed in the report | do not count toward any question. City
classifications, in increasing order: Village (<2k population), Town
(2k{10k), City (10k{50k), Capital (50k{100k), Metropolis (100k{500k),
Megalopolis (>500k).

For each question, give a probability that honestly reflects your uncertainty
about whether it resolves YES. Forecasts are scored with the Brier score
against the realized outcome: probabilities that match your true uncertainty
score best, and overconfidence in either direction scores worse.
```

Binary epilogue

[1] One exception: the continuous Micropolis preamble writes its inequalities with the *less than or equal* sign, which the typewriter font lacks, so the four signs in that block are printed as `<=`.

```
You MUST provide a probability for every question UNDER ALL CIRCUMSTANCES.
    Always return a number between 0 and 1.
Avoid responding with exactly 0 or 1, unless you are ABSOLUTELY SURE that the
    event is impossible or certain.

You may analyze the data, but you MUST end your response with your probabilities
    in this exact format:
<<<PROBABILITIES>>>
Q1: 0.65
Q2: 0.03
<<<END>>>

Provide one such line for each of the {n} questions, in order, replacing the
    example values with your probability that the answer is YES, as a decimal
    between 0 and 1.
```

Continuous preamble

```
You are an expert superforecaster, familiar with the work of
Tetlock and others. Your task is to forecast future values of several metrics
    for a
city in Micropolis, the open-source release of the original SimCity Classic
    simulation engine.

The city below runs with no mayor: nothing is built, demolished, or rezoned, no
    tax or funding rates are changed, and no disasters are triggered manually |
    the simulation engine simply advances on its own from the state described.
    Whether random disasters are enabled is stated in the game report.

Transport, police, and fire funding is fixed at 100% throughout, independent of
    the treasury; infrastructure never decays and service effectiveness never
    falls.

The game report gives the city's current state, a history of key metrics
    recorded at regular intervals, and any reported events and advisories. Form
your forecasts from what the report shows, combined with your knowledge of
the Micropolis engine.

For each question, give percentiles that honestly reflect your uncertainty
about that metric's value at that turn. Forecasts are scored with CRPS
against the realized values: intervals that are too narrow and intervals that
are too wide both score worse than intervals that match your true
uncertainty. In every answer line the five values must be non-decreasing:
p10 <= p25 <= p50 <= p75 <= p90, with p10 always the numerically lowest value.
```

Continuous epilogue

```
You MUST provide percentile estimates for every question UNDER ALL
    CIRCUMSTANCES. If for some reason you can't answer, provide reasonable
    mid-range estimates, but always return numeric percentile values.

You may analyze the data, but you MUST end your response with your percentile
    estimates in this exact format:
<<<PERCENTILES>>>
Q1: p10=5, p25=10, p50=15, p75=20, p90=25
Q2: p10=100, p25=200, p50=300, p75=400, p90=500
<<<END>>>

Provide one such line for each of the {n} questions, in order, replacing the
    example values with your actual percentile estimates.
- p10 means you estimate there's a 10% chance the true value is below this
    number
- p25 means you estimate there's a 25% chance the true value is below this
    number
```

```
- p50 (median) means you estimate there's a 50% chance the true value is below
    this number
- p75 means you estimate there's a 75% chance the true value is below this
    number
- p90 means you estimate there's a 90% chance the true value is below this
    number
```

Example world report This binary example uses city `bruce` at turn 960 with disasters enabled. The excerpt shows the current-state fields and the first two history rows; the remaining history and event log are omitted. Events in the full example include three disasters and standing pollution, traffic and treasury advisories. Questions follow the report, numbered 1–50: the 25 templates below repeat for turns 1104 and 1200; the second prompt covers turns 1296 and 1440.

```
## Game report
City: bruce
Disasters: enabled

CURRENT STATE (turn 960)

  Difficulty: Easy
  City class: City
  city score: 682
  population: 23520
  average traffic: 43
  average pollution: 65
  average crime: 79
  average land value: 77
  Tax rate: 0%

  Infrastructure | Roads 140, Rail 84, Police stations 2, Fire stations 1,
    Seaports 0, Airports 0, Hospitals 1, Stadiums 0, Coal plants 1, Nuclear
    plants 1, Powered zones 48, Unpowered zones 9
  Ground survey | rubble tiles 74, tiles on fire 0, road tiles 125

HISTORY (every 24 turns)

Turn,city score,population,average traffic,average pollution,average
    crime,average land value
0,755,14440,2,59,8,71
24,755,14440,2,68,40,101
[remaining history rows and EVENTS omitted]
```

Binary question templates The 25 templates, in the order they appear in the prompt, with {T} the horizon turn. Templates A1 to A16 target mid-range probabilities and B1 to B9 target tail probabilities. The analysis assigns every question to the mid-range or tail set by its ground-truth probability, not by this label.

A1 Will at least one earthquake be reported between the current turn and turn {T}?
A2 Will at least one tornado be sighted between the current turn and turn {T}?
A3 Will at least one flood be reported between the current turn and turn {T}?
A4 Will a monster be sighted between the current turn and turn {T}?
A5 Will an airplane crash between the current turn and turn {T}?
A6 Will a shipwreck be reported between the current turn and turn {T}?
A7 Will the city's population at turn {T} be lower than it is at the current turn?
A8 Will the city's population at turn {T} be less than half of its current value?
A9 Will the city's population read zero at any yearly checkpoint between the current turn and turn {T}?
A10 Will the city's classification at turn {T} be lower than it is now (e.g. Metropolis → Capital)?
A11 Will a "Blackouts reported" advisory appear between the current turn and turn {T}?
A12 Will the citywide average pollution level exceed 60 at turn {T}?
A13 Will the map hold at least 50 more rubble tiles at turn {T} than it does now?
A14 Will at least one map tile be actively burning at turn {T}?
A15 Will the map contain fewer road tiles at turn {T} than it does now?
A16 Will the city's evaluation score at turn {T} be higher than it is now?
B1 Will a nuclear meltdown occur between the current turn and turn {T}?
B2 Will two or more earthquakes be reported between the current turn and turn {T}?

B3 Will two or more tornadoes be sighted between the current turn and turn {T}?
B4 Will a "Fire reported!" disaster strike between the current turn and turn {T}?
B5 Will a train crash between the current turn and turn {T}?
B6 Will two or more separate floods be reported between the current turn and turn {T}?
B7 Will the monster be sighted two or more times between the current turn and turn {T}?
B8 Will the city's population reach a new all-time high at any yearly checkpoint between the current turn and turn {T}?
B9 Will the city's classification at turn {T} be higher than it is now?

Continuous question templates Each continuous question reads "What will the *metric* be at turn *T*?" The *metric* is one of population, average traffic, average pollution, average crime and average land value. A sixth template for city funds exists in the code and is not asked, because the funds are censored from the report.

B.2 FreeCiv

Our harness sends one user message and no system message. The user message has two parts: the opening instruction, question background and full world report; then the questions and answer instructions. First-party hosts receive the first part with a cache breakpoint, other hosts receive the same text as one string. In the run the paper reports, a prompt for the mid-range set, tail, mirror and continuous sets carries up to 50 binary or 20 continuous questions of one game, numbered, each with its resolution criteria, and asks for one answer line per question inside a delimited block; the instructions follow the Micropolis prompts above, with the FreeCiv background line and no persona. Natural conditionals retain the one-question prompt from September 9, with the same first part, resolution criteria and an answer in tags. Turn 2 of a cell continues that conversation with three messages: the turn-1 message, the model's own turn-1 reply, and a third message that reveals one fact and asks the question again. The no-news control replaces the third message with a statement that nothing new is known. The wording generated by our harness follows; the item names are those of the released question sets.

Grouped binary prompt, opening of the first part The world report follows the third line.

```
You will be asked several binary forecasting questions about a simulation in
    progress. For each question, your task is to estimate the probability that
    the event described in the question occurs. Please provide your reasoning
    before stating your final probabilities.

Question Background: This is a partial report on a FreeCiv game simulation in
    progress, observed at turn 60. Five AI civilizations are competing.
```

Second part, grouped binary Prompt `seed7008_bin_c1of4`, 43 questions; the first two are shown.

```
Questions (43 in total):

1. Will Peruvian have a higher population than Latvian at turn 90?
   Resolution Criteria: Resolves YES if Peruvian's population as recorded at the
    end of turn 90 is strictly greater than Latvian's. Equal values resolve NO.
    The quantity is the same quantity the world report calls Population.

2. Will Latvian have discovered University by turn 90?
   Resolution Criteria: Resolves YES if Latvian discovers University on any turn
    from 61 through 90 inclusive (an event on turn 90 itself counts). Resolves
    NO if it is not discovered by the end of turn 90.

[questions 3 to 43]

Think step by step about the information provided, reason about uncertainty, and
    then state your final confidence that each question resolves YES. Each
    probability should be a number between 0 and 1.

Your answers will be evaluated with a proper scoring rule, so your best strategy
```

```
    is to report your honest probability estimate for every question. You MUST
    provide a probability for every question.

Your response SHOULD STRICTLY END with your probabilities in this exact format,
    one line per question, in order:
<<<PROBABILITIES>>>
Q1: 0.65
Q2: 0.03
<<<END>>>
Provide one such line for each of the 43 questions, replacing the example values
    with your probability that the question resolves YES.
```

Grouped continuous prompt The first part opens with the paragraph below instead; the background line and the report are the same as in the binary prompt.

```
You will be asked several numeric forecasting questions about a simulation in
    progress. For each question, your task is to estimate the full range of
    plausible values for the quantity described, expressed as percentiles of
    your forecast distribution. Please provide your reasoning before stating
    your final percentiles.
```

Second part, grouped continuous Prompt `seed7003_cont_c1of3`, 16 questions; the first is shown.

```
Questions (16 in total):

1. How many wonders will be completed (across all civilizations) between turn 61
    and turn 90?
   Resolution Criteria: Resolves to the number of great wonders completed by all
    civilizations on any turn from 61 through 90 inclusive (an event on turn 90
    itself counts). Small wonders such as a palace do not count.

[questions 2 to 16]

Think step by step about the information provided, reason about uncertainty, and
    give your final answer to each question as five percentiles of your forecast
    distribution: p5, p25, p50, p75, p95. "p5" means you estimate a 5% chance
    the true value falls below that number; "p50" is your median. Your
    percentiles must be non-decreasing (p5 <= p25 <= p50 <= p75 <= p95). Every
    answer is a whole number.

Your answers will be evaluated with a proper scoring rule, so your best strategy
    is to report your honest estimate of each distribution - percentiles that
    are too narrow and percentiles that are too wide will both cost you. You
    MUST provide percentiles for every question.

Your response SHOULD STRICTLY END with your percentiles in this exact format, one
    line per question, in order:
<<<PERCENTILES>>>
Q1: p5=5, p25=10, p50=15, p75=20, p95=25
Q2: p5=100, p25=200, p50=300, p75=400, p95=500
<<<END>>>
Provide one such line for each of the 16 questions, replacing the example values
    with your actual percentile estimates.
```

Natural-conditional turn-1 prompt, opening of the first part One question per prompt, as in the run of September 9. The world report follows the third line.

```
You will be asked a binary forecasting question about a simulation in progress.
    Your task is to estimate the probability that the event described in the
    question occurs. Please provide your reasoning before stating your final
    probability.

Question Background: This is a partial report on a FreeCiv game simulation in
    progress, observed at turn 60. Five AI civilizations are competing.
```

Second part, natural-conditional turn 1 Item `seed7003:b687`.

```
Question Title: Will Ghana and Sammarinese reach a cease-fire or armistice at
    any point between turns 61 and 120?

Resolution Criteria: Resolves YES if the diplomatic state between Ghana and
    Sammarinese is Cease-fire or Armistice at the end of any turn from 61
    through 120 inclusive. Resolves NO if the state never occurs in that window
    (war, peace and alliance do not count).

Think step by step about the information provided, reason about uncertainty, and
    put your final confidence that the question resolves YES in <probability>
    </probability> tags. The probability should be a number between 0 and 1.

Your answer will be evaluated with a proper scoring rule, so your best strategy
    is to report your honest probability estimate.

Your final answer should be the probability that the event resolves YES and your
    response SHOULD STRICTLY END with <probability> </probability> tags.
```

Turn-2 message of a natural-conditional cell Cell `seed7003:b687|seed7003:r30`. The first sentence is the same in every cell up to the revealed fact; the turn on which the fact occurred is never disclosed.

```
One fact about turns 61 through 90 has been revealed to you: Rhenish had at
    least four more cities at turn 90 than at turn 60.

Given this, please answer the same question again. Reason about what, if
    anything, this changes, and end your response with your updated probability
    in <probability> </probability> tags.
```

No-news control message

```
The game has continued. No new information about turns 61 through 90 is
    available. If you wish, revise your forecast.

Please answer the same question again and end your response with your
    probability in <probability> </probability> tags.
```

Reading the answer For a grouped prompt we read the last delimited block in the reply and match each line to its question by the stated number. When the block is missing we accept numbered lines outside it and, for percentiles, five numbers on one line or on the line after a restated question, and mark the row as leniently parsed; each grouped prompt was sent once, a missing binary answer scores as 0.5, and a missing continuous answer is dropped. For a natural-conditional prompt the answer is the number inside the last `<probability>` tags; when the tags were missing we accepted a trailing "probability … 0.xx" or a bare `<0.xx>` at the end and marked the row as leniently parsed, and when neither could be read we asked once more.

World report The turn-60 report of anchor `seed7001` has 319 lines (19.2 KB). This excerpt retains the header, civilization list, current-state column headings and first civilization row. The omitted sections contain the other current states, diplomacy and borders, government changes, five-turn economic and military series, event definitions and 145 chronological events from turns 2–60.

```
WORLD REPORT TXT v1
Game ID: seed7001
Snapshot turn: 60
Turns analyzed: 1..60
Barbarians: default settings (barbarian and pirate raids begin around turn 60;
    none exist at turn 60)
Map size: 78x52

CIVILIZATIONS
ID | Name
```

```
---+---------------
0  | Brazzaville-Congolese
1  | Santomean
2  | Estonian
3  | Guinean
4  | Sahrawi

CURRENT STATE (TURN 60)
ID | Name                 | Score | Rank | Government | Treasury | Population |
    Techs | Wonders | Cities | Territory | Units | Military | Science | Trade |
    Food  | Shields
---+----------------------+-------+------+------------+----------+------------+
    -------+---------+--------+-----------+-------+----------+---------+-------+
    -------+--------
0  | Brazzaville-Congolese | 38    | 3    | Despotism  | 449      | 21         |
    8     | 0       | 14     | 188       | 8     | 1        | 60      | 21.0  |
    42.0  | 39.0
[remaining civilization rows and report sections omitted]
```

Question families We generated the FreeCiv questions from 23 binary and nine continuous families. Each family is a template with parameters (a civilization, a pair, a threshold, a count or a turn), and its resolution criteria state how we read the quantity from the saved game. The generator is released with the question sets; the tables below give each family's identifier, its number of questions, and one question drawn from it. Table 11 covers the mid-range set and the tail set, Table 12 the continuous set, and Table 13 the kinds of fact revealed in the natural-conditional cells.

Family	Mid-range	Tails	Example question
EX_tech_discovered	50	22	Will Brazzaville-Congolese have discovered Seafaring by turn 90?
NB1_value_threshold	51	20	Will Chinese's territory size be at least 405 at turn 90?
W2_directed_conquest	49	20	Will Guyanese capture at least one city from Libyan by turn 90?
EX_government_at	46	22	Will Abyssinian's government be Anarchy at turn 90?
NW2_diplo	44	21	Will Central American and Spanish be at formal Peace at any point between turns 61 and 90?
NW1_war_at	49	15	Will Spanish and Peruvian be at war at turn 90?
NB4_drawdown	44	20	Will Ghana's number of cities drop below 99% of its turn-60 level at any point up to turn 90?
S5_civ_wonders	43	21	Will Beninese complete at least 1 wonder by turn 90?
EX_comparative	53	9	Will Bissau-Guinean have a higher territory size than Tajik at turn 90?
W1_wonder_race	36	21	Will Greek complete the Great Wall by turn 90?
NW5_wonder	34	21	Will Marco Polo's Embassy be completed by any civilization by turn 90?
NB1_threshold	48	0	Will Cañari's number of technologies be at least 22 at turn 90?
W3_capture_k	38	3	Will American capture at least 2 cities by turn 90?
W5_tech_lead	23	15	Will Crimean Tatar know strictly more technologies than every other civilization at turn 90?
S4_gov_change_count	32	6	Will Guinean's government change at least 1 time by turn 90? (Anarchy counts as a form of government, for this question.)
W4_lose_k	35	3	Will Saka lose at least 5 cities by turn 90?
NEW_civil_war	17	20	Will Rhenish suffer a civil war (a breakaway civilization splitting off) by turn 120?
S6_city_founding	34	2	Will Ghana found at least 8 new cities by turn 90?
W6_peace_at	12	20	Will Santomean and Guinean have a formal peace in effect at turn 90?
NW4_survival	0	16	Will Western Roman be eliminated (cease to exist as a civilization) by turn 210?
NB6_event	7	3	Will at least 10 cities be conquered (in total) by turn 90?
S3_wars_at_T	3	0	Will at least 4 pairs of civilizations be at war at exactly turn 90?
S7_any_destroyed	2	0	Will any city be destroyed by turn 90?
Total	750	300	

Table 11: The 23 binary question families of FreeCiv: the questions each contributes to the 750-question mid-range set and to the 300-question tail set, and one example from each. 22 families appear in the mid-range set and 20 in the tail set.

Family	Questions	Example question
NC1_value_at_T	60	What will Peruvian's number of cities be at turn 90?
S7_world_razings	33	How many cities will be destroyed in total by turn 90?
NC5_world_captures	32	How many cities will be captured (in total, across all civilizations) by turn 90?
NC14_world_wonders	31	How many wonders will be completed (across all civilizations) between turn 61 and turn 90?
P1_civ_conquests	31	How many cities will Ostrogothic capture by turn 90?
P2_civ_losses	31	How many cities will Otomí lose by turn 90?
P3_civ_founds	31	How many new cities will Fijian found by turn 90?
P5_techs_at_T	31	How many technologies will Corsican know at turn 90?
P6_world_techs	20	What will the combined number of technologies known by all five civilizations be at turn 90?
Total	300	

Table 12: The nine continuous question families of FreeCiv. The answer is five percentiles; every quantity is a whole number.

Revealed fact	Cells	Example
civ_lost3	191	American lost at least three cities.
pair_war_began	90	Pelasgian and Abyssinian went to war.
civ_lost	39	Trinidadian and Tobagonian lost a city.
civ_score_leader	13	American had the highest score of any civilization at turn 90.
civ_govchange	12	Greek changed its form of government (not counting a spell of anarchy).
pair_ceasefire	12	Sammarinese and Corsican reached a cease-fire or armistice.
civ_cities_grew4	10	Rhenish had at least four more cities at turn 90 than at turn 60.
civ_cities_fell2	8	Saka had at least two fewer cities at turn 90 than at turn 60.
pair_peace	5	American and Spanish signed a peace treaty.
civ_captured	5	Peruvian captured a city.
civ_wonder	5	Spanish completed a wonder.
civ_techs_K	5	Belgian knew at least 21 technologies at turn 90.
civ_anarchy	4	American fell into anarchy at some point.
pair_contact	1	Estonian and Sahrawi made first contact.
Total	400	

Table 13: The 14 kinds of fact revealed at turn 2 in the 400 natural-conditional cells, with one example of each. The reveal sentence is "One fact about turns 61 through 90 has been revealed to you:" followed by the fact.

B.3 STARSIM

Every Starsim prompt combines the same four parts in one text: a statement of provenance, the world report, the question, and the answer format. The interventional items add one line, `PLANNED INTERVENTION`, between the report and the question; nothing else changes, and the unconditional and interventional versions of an item go out as separate prompts. No role is assigned and no scoring rule is named. The first block is the complete prompt of one continuous interventional item (world w0, day 40, 50 percent coverage); the unconditional item is the same prompt without the intervention line, and the 25 and 90 percent items change only the two coverage figures in that line. The second block is the complete prompt of one binary unconditional item (world w0, day 40); its interventional and hold items insert the two-region form of the intervention line, which ends "Southbay has no planned intervention." The prompts use the symbols $\approx$ and $\geq$, written here as `~=` and `>=`. Lines are wrapped for the page. The institution named in the statement of provenance is redacted for review.

Continuous item, interventional (world w0, day 40, 50 percent coverage).

```
This is a question from ForecastBench-Sim, a forecasting-calibration benchmark
    built by [redacted for anonymous review]. The data below comes from a
    toy SIR simulation (the open-source starsim package): one synthetic
    population of 5,000 agents with fixed, stated parameters, no behavior
    change, no travel, and no interventions unless stated. Your task is
    probabilistic forecasting of simulator output.

PANDEMIC WORLD REPORT - Day 20

One isolated region, Riverton: 5,000 people, no travel in or out.
Disease model: SIR (susceptible -> infectious -> recovered; no deaths).
    Infected people are infectious for 10 days on average, then recover with
    permanent immunity. Each person has about 6 contacts per day. The
    simulation is agent-based and stochastic.
    Transmission probability per contact per day 0.030 (basic reproduction
      number R0 ~= 1.8); 28 people infectious at day 0.

Observed cumulative infections at the end of each day (people ever infected,
    including those infectious at day 0):
    Riverton: d0:32 -> d5:66 -> d10:118 -> d15:225 -> d20:431   (day 20: 327
      currently infectious)

PLANNED INTERVENTION: Riverton will run a vaccination campaign on day 21 (95%
    efficacy, 50% coverage): on day 21, 50% of Riverton's population is
    vaccinated, and vaccination reduces a vaccinated person's probability of
    infection per exposure by 95%.

QUESTION: How many people in Riverton will be newly infected between the end
    of day 20 and the end of day 40? (That is: cumulative infections at day 40
    minus the 431 already infected by day 20.)

The report ends at day 20. The simulation is stochastic, so give a
    distribution: your 10th, 25th, 50th, 75th and 90th percentiles for the
    number of people (integers >= 0). You may reason briefly first. Then end
    your reply with one JSON object, alone on its final line, with exactly
    these keys:
    "p10", "p25", "p50", "p75", "p90"  (non-decreasing numbers of people)
```

Binary item, unconditional (world w0, day 40).

```
This is a question from ForecastBench-Sim, a forecasting-calibration benchmark
    built by [redacted for anonymous review]. The data below comes from a
    toy SIR simulation (the open-source starsim package): two synthetic
    populations of 5,000 agents each, with fixed, stated parameters, no
    behavior change, no travel between regions, and no interventions unless
    stated. Your task is probability calibration on simulator output.

PANDEMIC WORLD REPORT - Day 20

Two isolated regions, Riverton and Southbay: 5,000 people each, no travel
    between them.
Disease model: SIR (susceptible -> infectious -> recovered; no deaths).
    Infected people are infectious for 10 days on average, then recover with
    permanent immunity. Each person has about 6 contacts per day. The
    simulation is agent-based and stochastic.
    Riverton: transmission probability per contact per day 0.040 (basic
      reproduction number R0 ~= 2.4); 23 people infectious at day 0.
    Southbay: transmission probability per contact per day 0.030 (R0 ~= 1.8);
      28 people infectious at day 0.

Observed cumulative infections at the end of each day (people ever infected,
    including those infectious at day 0):
    Riverton: d0:28 -> d5:88 -> d10:227 -> d15:548 -> d20:1129   (day 20: 911
```

```
      currently infectious)
    Southbay: d0:32 -> d5:66 -> d10:118 -> d15:225 -> d20:431   (day 20: 327
      currently infectious)

QUESTION: Will Riverton have more cumulative infections than Southbay at day
    40?

The report ends at day 20. You may reason briefly first. Then end your reply
    with one JSON object, alone on its final line, with exactly this key:
    "p": probability that the answer is YES, a number in [0,1]
```

Two-region intervention line (world w0, 90 percent coverage).

```
PLANNED INTERVENTION: Riverton will run a vaccination campaign on day 21 (95%
    efficacy, 90% coverage): on day 21, 90% of Riverton's population is
    vaccinated, and vaccination reduces a vaccinated person's probability of
    infection per exposure by 95%. Southbay has no planned intervention.
```

C FULL RESULTS

This appendix gives the per-model scores behind the figures of Sections 3 to 5 and the correlations of Section 6. Every table lists models in descending order of the Epoch Capabilities Index (ECI, Section 2.3) and uses the short model names of the figures. Scores follow the definitions of Section 2.2. Excess Brier score, excess bits and excess normalized CRPS (nCRPS) are lower-is-better. Required artifacts are described in Section E.

C.1 MICROPOLIS

Table 14 lists all 24 models of the Micropolis run, pooled over the four horizons. Every model returned a forecast that we could parse on at least 98.3% of its 3,000 binary questions and 92.0% of its 600 continuous questions. Table 15 gives the two binary scores at each horizon, and Table 16 the excess nCRPS by horizon and by metric.

Table 14: Micropolis scores per model, pooled over the four horizons; lower is better. Excess Brier score on mid-range questions ($5\% < q < 95\%$, 983 per model), excess bits on tail questions ($0 < q \leq 5\%$, 940), and excess nCRPS on the 600 continuous questions. Parsed: percentage of questions with a readable forecast.

Model	ECI	Excess Brier	Excess bits	Excess nCRPS	Parsed (%)	
		Mid-range	Tail		Binary	Continuous
Claude Fable	162.5	0.0521	0.0837	0.0268	100.0	100.0
Claude Opus 5	161.5	0.0447	0.0747	0.0289	100.0	100.0
GPT-5.6 Sol	161.1	0.0428	0.0664	0.0286	100.0	100.0
GPT-5.5	158.6	0.0451	0.0481	0.0284	100.0	100.0
Gemini 3.7 Flash	156.8	0.0455	0.0494	0.0303	100.0	100.0
GPT-5.6 Luna	156.2	0.0555	0.1055	0.0334	100.0	100.0
Claude Sonnet 5	155.9	0.0522	0.1048	0.0300	100.0	100.0
DeepSeek V4 Flash 0731	154.4	0.0441	0.0698	0.0374	100.0	100.0
Gemini 3 Flash Preview	150.7	0.0518	0.1214	0.0369	100.0	100.0
GPT-5	150.0	0.0522	0.0980	0.0327	100.0	100.0
o3	147.1	0.0502	0.1017	0.0358	100.0	100.0
GPT-5.4 Nano	146.5	0.0683	0.1224	0.0411	100.0	100.0
o4 Mini	146.5	0.0817	0.1250	0.0419	100.0	100.0
GPT-5 Mini	145.8	0.0711	0.0939	0.0420	100.0	99.8
Gemini 3.1 Flash Lite Preview	144.7	0.0637	0.1045	0.0368	100.0	100.0
Qwen3.5-Flash	144.1	0.1345	0.5989	0.0514	100.0	96.7
Claude Haiku 4.5	142.9	0.0682	0.1508	0.0412	100.0	100.0
GPT-5 Nano	140.8	0.0875	0.1069	0.0711	100.0	92.0
Kimi K2	140.4	0.0661	0.1683	0.0417	98.4	98.3
Gemini 2.5 Flash	140.3	0.1020	0.2673	0.0411	100.0	99.8
Qwen3 235B A22B	139.6	0.0696	0.2331	0.0464	100.0	100.0
GPT-4.1	137.5	0.0764	0.1007	0.0415	100.0	100.0
DeepSeek V3	133.0	0.0727	0.2738	0.0379	100.0	99.8
Llama 4 Scout	130.5	0.0762	0.1321	0.0428	100.0	100.0

Table 15: Micropolis scores by horizon (game years after the snapshot); lower is better. Mid-range questions ($5\% < q < 95\%$) are scored in excess Brier score and tail questions ($0 < q \leq 5\%$) in excess bits. Each cell averages forecast scores for one model at one horizon.

Model	ECI	Mid-range excess Brier, by horizon (years)				Tail excess bits, by horizon (years)			
		3	5	7	10	3	5	7	10
Claude Fable	162.5	0.0583	0.0572	0.0467	0.0499	0.0871	0.0874	0.0743	0.0849
Claude Opus 5	161.5	0.0489	0.0489	0.0404	0.0434	0.0913	0.0890	0.0553	0.0567
GPT-5.6 Sol	161.1	0.0534	0.0480	0.0377	0.0379	0.0708	0.0816	0.0518	0.0578
GPT-5.5	158.6	0.0563	0.0477	0.0421	0.0399	0.0614	0.0608	0.0294	0.0356
Gemini 3.7 Flash	156.8	0.0571	0.0535	0.0399	0.0385	0.0677	0.0522	0.0356	0.0368
GPT-5.6 Luna	156.2	0.0602	0.0561	0.0512	0.0564	0.0965	0.1207	0.0882	0.1182
Claude Sonnet 5	155.9	0.0563	0.0575	0.0487	0.0492	0.1151	0.1172	0.0880	0.0946
DeepSeek V4 Flash 0731	154.4	0.0514	0.0485	0.0395	0.0413	0.0769	0.0717	0.0588	0.0703
Gemini 3 Flash Preview	150.7	0.0616	0.0580	0.0429	0.0501	0.1170	0.1406	0.1130	0.1133
GPT-5	150.0	0.0645	0.0574	0.0471	0.0466	0.1087	0.1130	0.0767	0.0893
o3	147.1	0.0580	0.0504	0.0455	0.0504	0.0941	0.0955	0.1036	0.1173
GPT-5.4 Nano	146.5	0.0614	0.0759	0.0632	0.0709	0.1292	0.1458	0.0902	0.1201
o4 Mini	146.5	0.0996	0.0868	0.0721	0.0774	0.1134	0.1296	0.1035	0.1585
GPT-5 Mini	145.8	0.0863	0.0763	0.0650	0.0650	0.0911	0.0994	0.0770	0.1095
Gemini 3.1 Flash Lite Preview	144.7	0.0773	0.0675	0.0583	0.0586	0.0851	0.1018	0.1087	0.1289
Qwen3.5-Flash	144.1	0.0964	0.1502	0.1240	0.1519	0.5180	0.6602	0.4940	0.7472
Claude Haiku 4.5	142.9	0.0803	0.0694	0.0633	0.0655	0.1378	0.1711	0.1201	0.1771
GPT-5 Nano	140.8	0.0866	0.0963	0.0836	0.0850	0.1639	0.0969	0.0887	0.0636
Kimi K2	140.4	0.0697	0.0741	0.0565	0.0668	0.1577	0.1940	0.1372	0.1848
Gemini 2.5 Flash	140.3	0.0876	0.0983	0.0971	0.1164	0.2743	0.3604	0.1905	0.2288
Qwen3 235B A22B	139.6	0.0690	0.0737	0.0619	0.0739	0.2148	0.2749	0.1994	0.2435
GPT-4.1	137.5	0.0954	0.0794	0.0671	0.0727	0.1133	0.1077	0.0806	0.0977
DeepSeek V3	133.0	0.0621	0.0856	0.0624	0.0776	0.2876	0.3179	0.2213	0.2596
Llama 4 Scout	130.5	0.0833	0.0887	0.0708	0.0682	0.1260	0.1493	0.1099	0.1438

Table 16: Micropolis excess nCRPS on the 600 continuous questions, by horizon and by metric; lower is better. The CRPS of each forecast is scored against the replay distribution and divided by a scale computed per city and metric.

Model	ECI	By horizon (years)				By metric				
		3	5	7	10	Popul.	Traffic	Pollut.	Crime	Land
Claude Fable	162.5	0.019	0.025	0.029	0.034	0.031	0.034	0.021	0.029	0.019
Claude Opus 5	161.5	0.020	0.026	0.031	0.039	0.033	0.032	0.028	0.031	0.022
GPT-5.6 Sol	161.1	0.021	0.027	0.030	0.037	0.031	0.033	0.025	0.032	0.022
GPT-5.5	158.6	0.020	0.026	0.030	0.037	0.030	0.031	0.025	0.035	0.020
Gemini 3.7 Flash	156.8	0.021	0.028	0.032	0.040	0.036	0.035	0.026	0.031	0.023
GPT-5.6 Luna	156.2	0.024	0.031	0.036	0.043	0.031	0.038	0.035	0.039	0.026
Claude Sonnet 5	155.9	0.022	0.028	0.032	0.038	0.036	0.043	0.016	0.034	0.020
DeepSeek V4 Flash 0731	154.4	0.029	0.035	0.039	0.047	0.056	0.040	0.027	0.032	0.032
Gemini 3 Flash Preview	150.7	0.023	0.032	0.041	0.052	0.032	0.031	0.050	0.035	0.036
GPT-5	150.0	0.025	0.031	0.035	0.040	0.035	0.047	0.027	0.031	0.024
o3	147.1	0.026	0.033	0.038	0.047	0.032	0.047	0.038	0.036	0.026
GPT-5.4 Nano	146.5	0.029	0.037	0.044	0.054	0.043	0.058	0.027	0.036	0.041
o4 Mini	146.5	0.034	0.039	0.045	0.050	0.053	0.054	0.033	0.039	0.031
GPT-5 Mini	145.8	0.033	0.039	0.044	0.051	0.040	0.049	0.038	0.041	0.043
Gemini 3.1 Flash Lite Preview	144.7	0.025	0.032	0.039	0.052	0.040	0.039	0.036	0.033	0.036
Qwen3.5-Flash	144.1	0.037	0.047	0.055	0.065	0.049	0.100	0.027	0.041	0.039
Claude Haiku 4.5	142.9	0.031	0.039	0.044	0.051	0.034	0.046	0.030	0.037	0.060
GPT-5 Nano	140.8	0.058	0.068	0.075	0.083	0.148	0.079	0.024	0.056	0.047
Kimi K2	140.4	0.030	0.036	0.043	0.058	0.034	0.041	0.039	0.047	0.048
Gemini 2.5 Flash	140.3	0.030	0.036	0.043	0.055	0.039	0.049	0.036	0.044	0.038
Qwen3 235B A22B	139.6	0.030	0.039	0.051	0.066	0.042	0.052	0.051	0.043	0.044
GPT-4.1	137.5	0.030	0.036	0.044	0.056	0.029	0.036	0.067	0.042	0.033
DeepSeek V3	133.0	0.026	0.033	0.040	0.052	0.036	0.051	0.030	0.041	0.031
Llama 4 Scout	130.5	0.031	0.037	0.045	0.058	0.057	0.059	0.031	0.031	0.037

Figure 5 separates mid-range and tail scores.

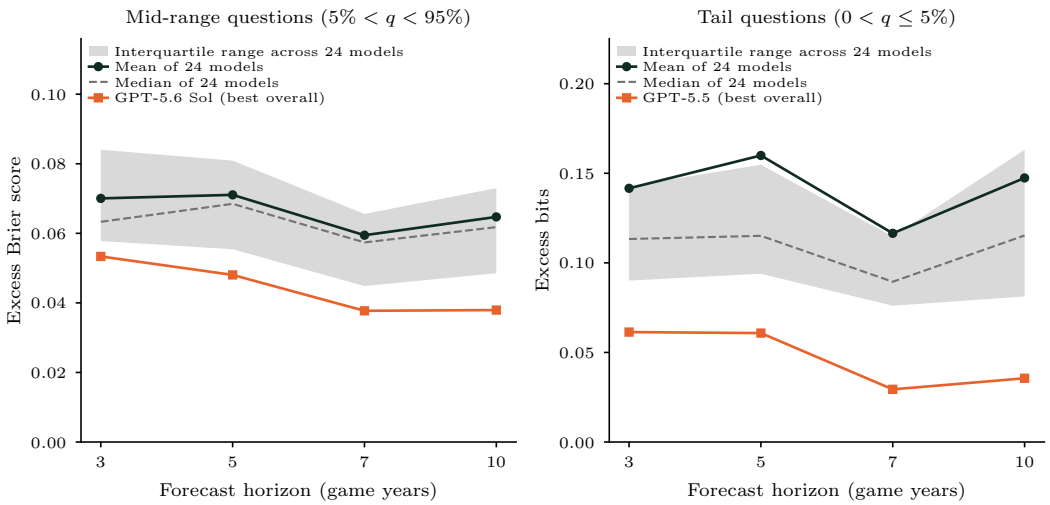


Figure 5: Micropolis forecasting scores by forecast horizon (3, 5, 7 and 10 game years); lower is better in both panels. Left: mid-range questions ($5\% < q < 95\%$), excess Brier score. Right: tail questions ($0 < q \leq 5\%$), excess bits. Solid line: mean over the 24 models; dashed: median; band: interquartile range across models; orange: the best model pooled over horizons (GPT-5.6 Sol for mid-range, GPT-5.5 for the tail).

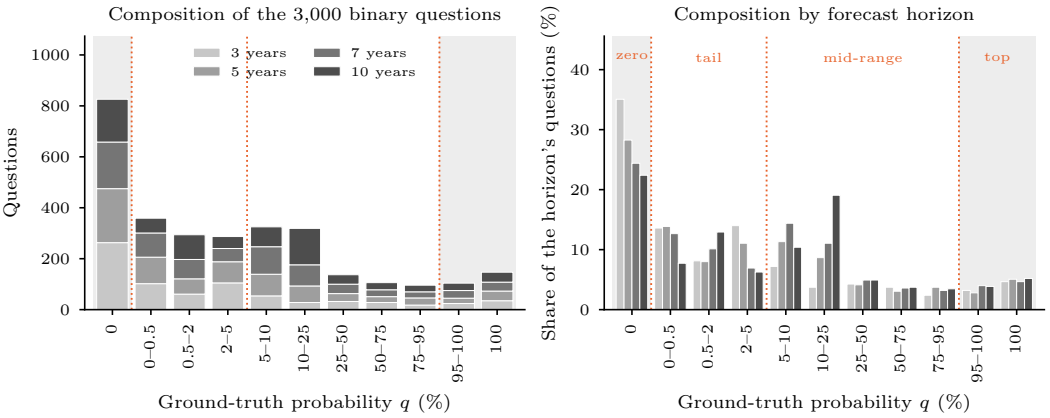


Figure 6: Composition of the Micropolis binary question set by ground-truth probability q, over the 3,000 questions each model is asked. Left: questions per probability band, stacked by forecast horizon. Right: the same counts as each horizon's share of its own questions. The bands contain unequal numbers of questions; a template's intended band does not determine which band its replay probability falls in. Each band includes its upper edge, so that $q = 5\%$ is a tail question, except that the top set starts at $q = 95\%$, and $q = 0$ and $q = 1$ have bars of their own. Dotted lines separate the four sets: zero ($q = 0$, 27.5% of the questions), tail ($0 < q \leq 5\%$, 31.3%), mid-range ($5\% < q < 95\%$, 32.8%) and top ($q \geq 95\%$, 8.4%). The shaded zero and top sets are in neither scored set.

Figure 6 shows the realized base-rate composition. Most mid-range questions lie below $q = 25\%$, rather than near one half.

As horizons lengthen, the same templates can move between probability bands. Horizon comparisons therefore change question composition and position within horizon-ordered prompts (Section C.1).

Selection and uncertainty checks. The current Micropolis split has 940 positive-tail, 983 mid-range, 826 zero and 251 top instances. Selection and scoring use the same empirical replay probabilities. Replay estimation error near a cutoff can change which questions enter a set, and excluding zero estimates selects positive estimates of very rare events. Zero successes in 1,000 independent Bernoulli replays would imply a one-sided 95% upper bound of 0.003, not a known probability of zero; this is neither a joint bound over items nor a verification of replay independence. Varying the positive-tail cutoff through 1, 2.5, 4, 5 and 6 percent gives ECI correlations 0.708, 0.729, 0.757, 0.750 and 0.763. These checks vary the composition of the question set; they do not correct for selection bias or finite-replay error.

The existing whole-city bootstrap intervals are $[0.74,\ 0.84]$ (mid-range), $[0.66,\ 0.81]$ (tail), and $[0.66,\ 0.85]$ (continuous). Each resampled city contributes both snapshots and all its questions together. They measure variation from reweighting the 15 selected cities while holding forecasts and empirical truths fixed; they do not combine uncertainty from models and replays.

For the 21 missing mid-range and 10 missing tail answer entries, a sensitivity check that assigns 0.5 to missing answers leaves ECI correlations unchanged. The reported Micropolis scores still exclude unparsed forecasts. Continuous excess is unbounded, and the exported scores lack the original quantiles, so we cannot establish a finite worst-case bound or rescore the forecasts against new replay samples from these files alone. Audit code, exact inputs and sensitivities are retained separately in the local supporting records.

C.2 STARSIM

Historical binary diagnostic, excluded from headline scores. The report-visible rule "the region ahead at day 20 remains ahead" solves all eight unconditional binary questions and attains the clipped minimum, $-\log_2(0.999) = 0.0014434$ bits per item. Reversing that rule solves all eight 90 percent items. Ignoring intervention gets the correct side on all eight hold items, although their non-endpoint probabilities still require calibration. These are findings about the selected cached sets, not guarantees on new worlds. The unconditional binary ECI association ($\rho = 0.64$ in skill orientation) describes this ranking diagnostic; it is excluded from headline validation and the combined score because of what the task measures, independently of that correlation.

Historical hold set. The unchanged, appendix-only binary `hold` set has eight items, four per horizon, with five responses planned per item. The vaccinated region is the one ahead at day 20; coverage is the largest 5-percent step preserving the ranking at both horizons (20, 10, 15 and 20 percent). Historical probabilities are fractions over all combinations of regional draws, rather than matched pairs across regions; day-60 probabilities are 0.97, 0.98, 0.01 and 0.03. Neither these truths nor the other binary sets were repaired by the continuous rescore. The historical pooled skill-oriented correlation $\rho(\mathrm{ECI}, -\mathrm{loss})$ is 0.56 ($n = 23$, $p = 0.005$), with 0.64/0.41 by horizon; this is not a headline validation result. The set records 905 calls costing $6.48. DeepSeek covers only three day-40 and one day-60 items at repetition 0, so its mean over answered items (0.0171203 bits) is omitted from tables comparing the full set; it ranks first on those same four items, including a repetition-0-only comparison. Kimi covers all eight items with 21 of 40 planned responses. The original answers and historical details are preserved.

Coverage, unconditional and horizon results. The revised skill-oriented interventional correlations with ECI at 25, 50 and 90 percent coverage are $\rho = 0.70$ $[0.40, 0.86]$, 0.48 $[0.07, 0.76]$ and 0.25 $[-0.18, 0.63]$, using the model-panel intervals of Section A.3. In people, mean excess rises from 153 at day 40 to 191 at day 60, pooled over coverage levels, and 19 of 23 models do worse at day 60. At 50 percent coverage it rises from 142 to 199, with 17 of 23 worse (Figure 8). Unconditional continuous excess falls from 272 to 245 people, with 9 of 24 worse; normalized model means range from 0.227 to 4.363, and the skill-oriented ECI correlation is 0.75 $[0.49, 0.88]$. Original binary scores are unchanged: 14 of 24 models are within 0.05 bit of their original truth and the largest excess is 0.34 bit. These binary targets were not repaired by the continuous production run.

Tables 17 to 19 report pooled scores and horizon breakdowns. Calls and costs count each retained response once (Section A.3). Partial continuous scores carry a dagger; partial hold means are omitted from full-set comparisons (Section C.2). Mean interventional excess rises in people but falls after horizon normalization.

Table 17: Starsim scores per model, pooled over day 40 and day 60; lower is better in every score column. Binary bits: excess bits $\mathrm{KL}(P^* \,\|\, f)$ on the unconditional comparison of the two regions' cases (lower is better). Cont. excess: excess CRPS on unconditional new infections, divided by the horizon's constant (0 attains the floor at the scored quantiles). Interventional excess CRPS: the same for new infections under a vaccination of 25, 50 or 90 percent of Riverton, and pooled. Binary c90 bits: excess bits on the binary interventional set at 90 percent coverage. Hold bits: excess bits on the hold set (Section C.2). Calls and cost (OpenRouter USD) cover all five sets. †: DeepSeek V4 Flash completed 21 of 24 continuous intervention items; its hold score is omitted because it covers only 4 of 8 items (3/4 at day 40, 1/4 at day 60). The September 16 forecasts, rescored for continuous outcomes only; binary truths are unchanged.

Model	ECI	Binary	Cont.	Interventional excess CRPS, by coverage			Binary	Hold	Calls	Cost
		bits	excess	25%	50%	90%	pooled	c90 bits	bits		($)
Fable	162.5	0.018	0.274	0.428	0.440	0.658	0.509	0.103	0.097	216	8.15
Opus 5	161.5	0.024	0.255	0.541	0.278	0.168	0.329	0.125	0.047	216	3.01
GPT-5.6 Sol	161.1	0.001	0.308	0.267	0.250	0.017	0.178	0.004	0.038	216	2.88
GPT-5.5	158.6	0.003	0.393	0.297	0.380	0.102	0.260	0.052	0.159	216	9.49
Gemini 3.7 Flash	156.8	0.005	0.424	1.058	0.918	0.327	0.768	0.069	0.018	216	0.81
GPT-5.6 Luna	156.2	0.016	0.496	0.285	0.291	0.063	0.213	0.060	0.471	216	0.35
Sonnet 5	155.9	0.055	0.502	0.321	0.613	0.318	0.417	1.628	0.161	216	1.64
DeepSeek V4 Flash†	154.4	0.001	0.239	0.246	0.298	0.186	0.243	0.004	–	111	0.45
Gemini 3 Flash	150.7	0.013	0.577	0.924	0.764	0.356	0.681	0.133	0.046	216	0.45
GPT-5	150.0	0.051	0.814	0.878	0.501	0.167	0.516	1.619	0.086	216	0.81
o3	147.1	0.115	0.314	0.511	0.374	0.328	0.404	0.373	0.703	216	4.16
GPT-5.4 Nano	146.5	0.042	0.227	1.014	1.012	0.403	0.810	0.144	0.219	216	0.53
o4-mini	146.5	0.016	0.435	1.150	1.663	0.499	1.104	0.052	1.077	216	1.07
GPT-5 Mini	145.8	0.050	1.498	1.054	0.431	0.155	0.547	0.716	0.244	216	0.24
Gemini 3.1 Flash Lite	144.7	0.027	0.424	0.394	0.467	0.601	0.488	0.114	0.376	216	0.20
Qwen3.5 Flash	144.1	0.059	1.340	0.683	0.699	0.255	0.546	0.186	0.916	216	0.26
Haiku 4.5	142.9	0.092	2.123	0.933	0.433	0.673	0.680	1.931	0.281	216	1.35
GPT-5 Nano	140.8	0.272	3.836	2.987	0.951	0.886	1.608	1.001	0.497	216	0.04
Kimi K2	140.4	0.122	0.793	0.609	1.046	0.484	0.713	0.668	0.282	182	1.67
Gemini 2.5 Flash	140.3	0.013	0.540	1.137	0.319	0.148	0.535	0.054	0.991	216	1.56
Qwen3 235B	139.6	0.130	2.901	2.932	0.967	0.160	1.353	1.486	0.636	216	0.49
GPT-4.1	137.5	0.018	0.652	1.017	0.494	0.421	0.644	1.530	0.036	216	1.20
DeepSeek V3	133.0	0.157	2.390	3.287	1.775	0.395	1.819	0.449	0.833	216	0.13
Llama 4 Scout	130.5	0.342	4.363	2.070	0.757	0.210	1.013	1.404	0.513	216	0.06

Table 18: Starsim scores per model at day 40 and day 60; lower is better. Columns as in Table 17; Interv. excess pools the three coverage levels. The hold set is defined in Section C.2.

Model	ECI	Binary bits		Cont. excess		Interv. excess		Binary c90 bits		Hold bits	
		d40	d60	d40	d60	d40	d60	d40	d60	d40	d60
Fable	162.5	0.018	0.018	0.304	0.244	0.448	0.569	0.125	0.082	0.058	0.136
Opus 5	161.5	0.026	0.022	0.284	0.226	0.263	0.395	0.144	0.105	0.054	0.040
GPT-5.6 Sol	161.1	0.001	0.001	0.398	0.217	0.209	0.146	0.006	0.001	0.021	0.056
GPT-5.5	158.6	0.002	0.005	0.542	0.244	0.243	0.277	0.095	0.010	0.077	0.241
Gemini 3.7 Flash	156.8	0.001	0.008	0.531	0.318	0.921	0.615	0.089	0.048	0.020	0.016
GPT-5.6 Luna	156.2	0.018	0.014	0.354	0.639	0.139	0.287	0.095	0.026	0.158	0.784
Sonnet 5	155.9	0.048	0.063	0.700	0.304	0.443	0.391	1.416	1.840	0.260	0.062
DeepSeek V4 Flash†	154.4	0.001	0.001	0.268	0.210	0.187	0.317	0.005	0.003	–	–
Gemini 3 Flash	150.7	0.011	0.014	0.906	0.248	0.970	0.393	0.152	0.114	0.088	0.003
GPT-5	150.0	0.052	0.051	1.228	0.400	0.559	0.472	2.127	1.110	0.103	0.069
o3	147.1	0.105	0.125	0.296	0.331	0.323	0.485	0.423	0.322	0.358	1.047
GPT-5.4 Nano	146.5	0.065	0.018	0.286	0.168	0.678	0.941	0.192	0.095	0.318	0.120
o4-mini	146.5	0.018	0.014	0.501	0.369	0.998	1.210	0.048	0.056	1.348	0.806
GPT-5 Mini	145.8	0.048	0.051	1.471	1.525	0.485	0.608	0.696	0.736	0.479	0.009
Gemini 3.1 Flash Lite	144.7	0.033	0.022	0.559	0.288	0.553	0.422	0.154	0.074	0.084	0.668
Qwen3.5 Flash	144.1	0.052	0.067	2.340	0.340	0.606	0.485	0.238	0.135	0.986	0.846
Haiku 4.5	142.9	0.098	0.086	3.050	1.196	0.942	0.418	1.996	1.867	0.349	0.213
GPT-5 Nano	140.8	0.257	0.288	3.684	3.987	1.918	1.298	1.152	0.851	0.525	0.470
Kimi K2	140.4	0.096	0.148	1.019	0.567	0.781	0.644	0.837	0.499	0.408	0.155
Gemini 2.5 Flash	140.3	0.013	0.013	0.580	0.501	0.612	0.457	0.063	0.044	0.911	1.072
Qwen3 235B	139.6	0.145	0.114	3.594	2.208	1.756	0.951	1.486	1.486	0.699	0.574
GPT-4.1	137.5	0.018	0.018	0.908	0.397	0.782	0.505	2.300	0.761	0.042	0.030
DeepSeek V3	133.0	0.074	0.241	3.192	1.588	1.904	1.734	0.673	0.224	0.856	0.809
Llama 4 Scout	130.5	0.370	0.313	5.634	3.092	1.092	0.933	1.397	1.411	0.416	0.610

Table 19: Starsim interventional excess CRPS per model by coverage level and horizon, divided by the horizon's constant; lower is better. Each cell contains four items with three repetitions planned per item. We score the median of the available saved answers and report the mean score over items. †: DeepSeek V4 Flash completed 3 of 4 items in three cells: day 60 at 25 percent coverage, day 40 at 50 percent and day 60 at 90 percent.

Model	ECI	25% coverage		50% coverage		90% coverage	
		d40	d60	d40	d60	d40	d60
Fable	162.5	0.351	0.505	0.268	0.612	0.725	0.591
Opus 5	161.5	0.473	0.609	0.207	0.349	0.110	0.227
GPT-5.6 Sol	161.1	0.325	0.209	0.282	0.217	0.021	0.013
GPT-5.5	158.6	0.256	0.338	0.380	0.380	0.092	0.112
Gemini 3.7 Flash	156.8	1.154	0.963	1.214	0.621	0.394	0.260
GPT-5.6 Luna	156.2	0.223	0.348	0.151	0.431	0.042	0.083
Sonnet 5	155.9	0.346	0.295	0.648	0.578	0.336	0.301
DeepSeek V4 Flash†	154.4	0.141	0.385	0.292	0.302	0.128	0.264
Gemini 3 Flash	150.7	1.246	0.601	1.334	0.194	0.331	0.382
GPT-5	150.0	0.885	0.871	0.672	0.331	0.120	0.214
o3	147.1	0.232	0.790	0.237	0.511	0.501	0.155
GPT-5.4 Nano	146.5	0.917	1.111	0.678	1.345	0.439	0.368
o4-mini	146.5	1.228	1.073	1.201	2.125	0.565	0.432
GPT-5 Mini	145.8	0.780	1.327	0.482	0.379	0.193	0.116
Gemini 3.1 Flash Lite	144.7	0.537	0.251	0.461	0.474	0.660	0.543
Qwen3.5 Flash	144.1	0.877	0.488	0.793	0.605	0.148	0.363
Haiku 4.5	142.9	1.433	0.433	0.348	0.519	1.044	0.302
GPT-5 Nano	140.8	3.449	2.526	1.152	0.750	1.153	0.619
Kimi K2	140.4	0.583	0.635	1.111	0.980	0.649	0.318
Gemini 2.5 Flash	140.3	1.114	1.161	0.528	0.109	0.195	0.101
Qwen3 235B	139.6	4.016	1.849	1.033	0.902	0.218	0.102
GPT-4.1	137.5	1.669	0.365	0.292	0.697	0.386	0.455
DeepSeek V3	133.0	3.134	3.440	2.155	1.395	0.422	0.368
Llama 4 Scout	130.5	2.254	1.887	0.754	0.761	0.268	0.152

Additional figures. Figure 7 shows the two unconditional Starsim questions against ECI and Figure 8 interventional excess at day 40 and day 60 by coverage; Section 5 quotes their numbers.

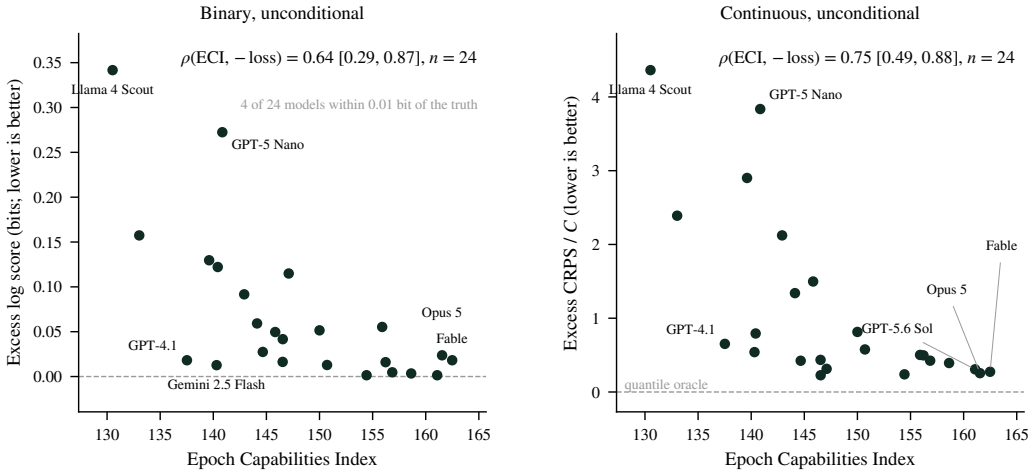


Figure 7: Unconditional Starsim questions against the Epoch Capabilities Index, $n = 24$, pooled over horizons; lower is better in both panels. Left: binary comparison of the two regions' cumulative cases, excess bits (lower is better); 14 of 24 models are within 0.05 bit of the empirical truth; the visible ranking rule attains the clipped minimum. Right: continuous new-infections question, excess CRPS divided by the horizon's constant (0 attains the floor at the scored quantiles, dashed line); four models are above 2. The continuous panel is retrospectively rescored; the binary panel is unrepaired. Spearman $\rho(\mathrm{ECI}, -\mathrm{loss})$ has model-panel intervals (2,000 draws for continuous; the original 10,000 for binary); a positive ρ means that the more capable models have the lower excess.

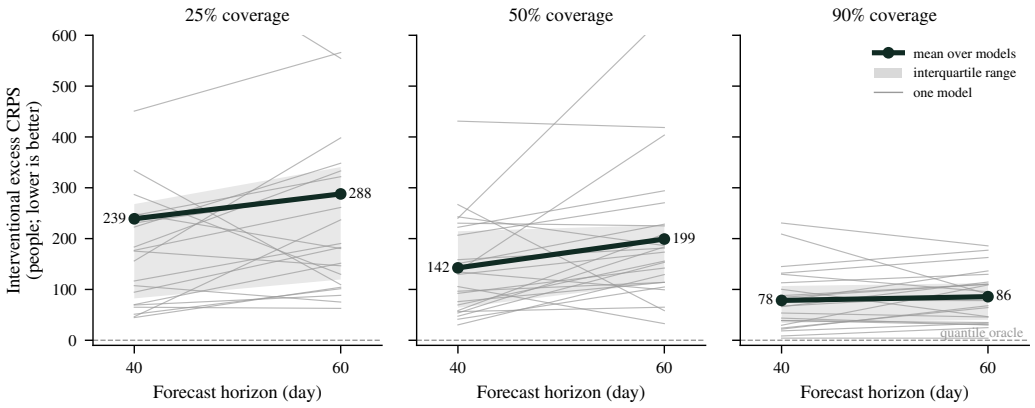


Figure 8: Interventional excess CRPS, in people, at day 40 and day 60 by vaccination coverage; lower is better. Thick line: mean over the 23 complete models; band: interquartile range; thin lines: one model each, three of which rise above the panel at 25 percent coverage and one at 50. Excess rises with horizon at every coverage: the mean goes from 239 to 288 at 25 percent, 142 to 199 at 50 and 78 to 86 at 90, and 16, 17 and 13 of 23 models do worse at day 60 than at day 40. DeepSeek V4 Flash is absent from all three panels.

Table 20 lists the 56 items, their transmission parameters, coverage, horizon, truth, empirical five-quantile floor and fixed normalization scale, plus the median, lowest and highest model excess on complete-item panels. Continuous truths and floors use the retrospective fixed-state rescore; binary values retain their original matched-seed construction. Median item difficulty is 0.55 at 25 percent coverage, 0.45 at 50 percent, 0.44 for unconditional continuous outcomes and 0.20 at 90 percent.

These losses measure predictions of outcomes under each condition; they do not isolate skill in estimating the intervention effect.

Table 20: Starsim items and their difficulty using saved September 16 forecasts. Truth: original matched-seed probability (binary, unrepaired) or median new infections from 1,000 fixed-state continuations (continuous rescore). Floor: the CRPS of the ground-truth distribution's own quantiles, in people; C: the horizon's constant that divides the excess. Excess: excess bits (binary) or excess CRPS divided by C (continuous), per item, over the models that answered every item of the set (24, or 23 where DeepSeek V4 Flash has missing items).

Set	Item	β (R_0) or β_R / β_S	Cov. %	Day	Truth	Floor	C	Excess med.	Min	Max
Binary, unconditional	w0_Rlead_d40	0.040 / 0.030	–	40	1.00	–	–	0.01	0.00	0.51
	w0_Rlead_d60	0.040 / 0.030	–	60	1.00	–	–	0.03	0.00	0.51
	w1_Rlead_d40	0.050 / 0.040	–	40	1.00	–	–	0.03	0.00	0.36
	w1_Rlead_d60	0.050 / 0.040	–	60	1.00	–	–	0.03	0.00	0.42
	w2_Slead_d40	0.050 / 0.065	–	40	0.00	–	–	0.05	0.00	0.32
	w2_Slead_d60	0.050 / 0.065	–	60	0.00	–	–	0.02	0.00	0.79
	w3_Slead_d40	0.030 / 0.040	–	40	0.00	–	–	0.01	0.00	0.32
	w3_Slead_d60	0.030 / 0.040	–	60	0.00	–	–	0.02	0.00	0.32
Continuous, unconditional	w0_d40	0.030 (1.8)	–	40	1669	52	200	0.68	0.20	6.77
	w0_d60	0.030 (1.8)	–	60	2824	45	300	0.52	0.07	7.54
	w1_d40	0.040 (2.4)	–	40	2845	26	200	0.86	0.08	9.28
	w1_d60	0.040 (2.4)	–	60	3128	21	300	0.37	0.16	5.85
	w2_d40	0.050 (3.0)	–	40	2276	13	200	0.51	0.10	5.69
	w2_d60	0.050 (3.0)	–	60	2308	12	300	0.37	0.07	2.33
	w3_d40	0.065 (3.9)	–	40	865	7	200	0.29	0.08	3.04
	w3_d60	0.065 (3.9)	–	60	866	7	300	0.22	0.04	2.32
Continuous, 25% coverage	w0_d40	0.030 (1.8)	25	40	863	37	200	0.32	0.01	3.26
	w0_d60	0.030 (1.8)	25	60	1383	52	300	0.44	0.04	3.29
	w1_d40	0.040 (2.4)	25	40	1813	30	200	1.50	0.11	6.68
	w1_d60	0.040 (2.4)	25	60	2051	30	300	0.69	0.10	4.44
	w2_d40	0.050 (3.0)	25	40	1629	19	200	0.96	0.11	4.64
	w2_d60	0.050 (3.0)	25	60	1656	19	300	0.65	0.13	4.33
	w3_d40	0.065 (3.9)	25	40	655	9	200	0.37	0.08	2.95
	w3_d60	0.065 (3.9)	25	60	656	9	300	0.30	0.04	2.14
Continuous, 50% coverage	w0_d40	0.030 (1.8)	50	40	388	21	200	0.24	0.06	2.06
	w0_d60	0.030 (1.8)	50	60	494	29	300	0.41	0.04	4.61
	w1_d40	0.040 (2.4)	50	40	964	26	200	0.74	0.15	3.60
	w1_d60	0.040 (2.4)	50	60	1065	29	300	0.71	0.08	2.65
	w2_d40	0.050 (3.0)	50	40	1023	19	200	0.54	0.19	3.59
	w2_d60	0.050 (3.0)	50	60	1039	20	300	0.49	0.07	3.08
	w3_d40	0.065 (3.9)	50	40	449	10	200	0.30	0.02	1.11
	w3_d60	0.065 (3.9)	50	60	450	10	300	0.19	0.03	1.34
Continuous, 90% coverage	w0_d40	0.030 (1.8)	90	40	55	5	200	0.11	0.01	0.90
	w0_d60	0.030 (1.8)	90	60	55	5	300	0.12	0.00	1.06
	w1_d40	0.040 (2.4)	90	40	161	8	200	0.35	0.02	1.53
	w1_d60	0.040 (2.4)	90	60	162	9	300	0.35	0.02	1.18
	w2_d40	0.050 (3.0)	90	40	242	9	200	0.38	0.04	1.76
	w2_d60	0.050 (3.0)	90	60	243	9	300	0.26	0.02	1.08
	w3_d40	0.065 (3.9)	90	40	139	6	200	0.14	0.00	2.82
	w3_d60	0.065 (3.9)	90	60	139	6	300	0.13	0.00	1.21
Binary, 90% coverage	w0_Rlead_d40	0.040 / 0.030	90	40	0.00	–	–	0.23	0.00	3.64
	w0_Rlead_d60	0.040 / 0.030	90	60	0.00	–	–	0.07	0.00	2.74
	w1_Rlead_d40	0.050 / 0.040	90	40	0.00	–	–	0.12	0.00	2.74
	w1_Rlead_d60	0.050 / 0.040	90	60	0.00	–	–	0.08	0.00	3.64
	w2_Slead_d40	0.050 / 0.065	90	40	1.00	–	–	0.23	0.00	2.74
	w2_Slead_d60	0.050 / 0.065	90	60	1.00	–	–	0.22	0.00	2.74
	w3_Slead_d40	0.030 / 0.040	90	40	1.00	–	–	0.19	0.00	4.64
	w3_Slead_d60	0.030 / 0.040	90	60	1.00	–	–	0.14	0.00	3.47
Binary, hold	w0_Rlead_d40	0.040 / 0.030	20	40	1.00	–	–	0.07	0.00	0.51
	w0_Rlead_d60	0.040 / 0.030	20	60	0.97	–	–	0.08	0.00	1.51
	w1_Rlead_d40	0.050 / 0.040	10	40	1.00	–	–	0.10	0.01	2.00
	w1_Rlead_d60	0.050 / 0.040	10	60	0.98	–	–	0.07	0.00	2.27
	w2_Slead_d40	0.050 / 0.065	15	40	0.00	–	–	0.35	0.02	3.63
	w2_Slead_d60	0.050 / 0.065	15	60	0.01	–	–	0.27	0.01	4.21
	w3_Slead_d40	0.030 / 0.040	20	40	0.00	–	–	0.07	0.00	0.47
	w3_Slead_d60	0.030 / 0.040	20	60	0.03	–	–	0.03	0.00	2.47

Table 21: Retrospective Starsim continuous rescore and current combined-score sensitivity. Positive ρ means better scores accompany higher external scores. Intervals from resampling models and replays are reported separately and hold different quantities fixed (Section A.3); neither combines both sources of uncertainty. The nine-cell combined fit excludes all binary Starsim scores.

Score	Comparator	n	ρ	Models, 95%	Replays, 95%
Continuous baseline	ECI	24	0.75	$[0.49, 0.88]$	$[0.75, 0.76]$
Continuous baseline	ForecastBench	17	0.24	$[-0.35, 0.82]$	$[0.24, 0.24]$
Interventional outcomes	ECI	23	0.67	$[0.34, 0.85]$	$[0.67, 0.68]$
Interventional outcomes	ForecastBench	17	0.45	$[-0.14, 0.88]$	$[0.45, 0.45]$
Combined nine cells	ECI	24	0.86	$[0.69, 0.93]$	$[0.86, 0.86]$
Combined nine cells	ForecastBench	17	0.37	$[-0.22, 0.81]$	$[0.37, 0.37]$

C.3 FreeCiv

FreeCiv uses the grouped run of September 20, 2026, and September 9 conditionals reused for 22 models, with two conditional reruns on September 20 (Section A).

Table 22 lists the four main-text sets for the 24 models, pooled over horizons. It adds the excess nCRPS, which subtracts the five-quantile score at the replay distribution's own quantiles, and the natural-conditional gain, the turn-1 excess Brier score against $p(Y \mid X)$ minus the turn-2 score. The parsed share counts the answers we could read over the mid-range set, tail, mirror, continuous and natural-conditional sets, 1,800 items per model; it is 99.1% or more for every model. The cost column is each model's OpenRouter charge for the calls behind its scored rows, $174.15 over the 24 models (Section A.5). The gain is positive for 7 of 24 models, with mean -0.002 and range -0.018 to 0.016. Gains compare forecasts against the same conditional truth; panel means weight models equally after averaging available cells. With forecasts and replay estimates fixed, 10,000 paired bootstrap resamples of the eight source worlds give a nominal 95% interval $[-0.016, 0.009]$ for mean gain; resampling whole model vectors separately gives $[-0.005, 0.002]$. These summaries describe variation across a few selected worlds and a deliberately chosen model panel. They do not combine replay and world uncertainty or measure variation across repeated model responses.

Mirrors. The mirror set is a check on the top end of the probability scale that the main text does not use. It holds 50 binary questions with $0.95 \leq q < 1$, ten per horizon, drawn from the binary templates; the mean q is 0.982. Table 23 gives each model's excess Brier score with a 95% percentile item-bootstrap interval (1,000 draws, seed 0), which is conditional on released truth, not world or replay uncertainty, and the score by horizon. Excess Brier runs from 0.263 (Gemini 3 Flash Preview) to 0.665 (GPT-5 Nano). A constant forecast of 0.5 has an excess of 0.233 on this set, and no model scores below it: models given grouped prompts assign low probabilities to these near-certain events, as they do on the mid-range set (Section D.1). The skill-oriented Spearman correlation $\rho(\text{ECI}, -\text{loss})$ is 0.10 [-0.37, 0.50], $p = 0.65$, with the model bootstrap of Section 6.

By horizon. Table 24 gives the mid-range excess Brier score and the tail excess bits at each resolution turn, and Table 25 the continuous nCRPS and the natural-conditional turn-2 excess Brier score. Across the 24 models the mean mid-range score stays between 0.136 and 0.149 over the five horizons, and the mean natural-conditional score between 0.112 and 0.124 over its four. The mean tail bits rise from 0.28 at turn 90 to 0.41 at turn 210, and the mean nCRPS from 0.41 to 1.03. Figure 9 plots the excess nCRPS instead, which subtracts the floor and rises from 0.35 to 0.82 over the same turns.

Natural conditionals by structural group. Table 26 splits the 400 cells by the five structural groups of Section 4.1: A, irrelevant news (36 cells); B, related weak news (36); C1, mechanical same-series (56); C2, persistence across series (172); and D, inferential (100). The gain over not updating is negative for 23 of 24 models in group C1, where the mean gain is -0.053. In the other four groups the mean gain lies between 0.003 and 0.010, and it is negative for 3, 9, 9 and 4 models in groups A, B, C2 and D. The other 344 cells have mean gain $+0.006$. The mean gain in C1 remains negative when any one world is left out; the source-world interval for the non-C1 mean includes zero. Positive means are descriptive.

Table 22: FreeCiv scores per model, pooled over horizons; lower is better in every score column except gain. Mid-range: excess Brier score $(f - q)^2$ on 750 questions. Tails: excess bits $\mathrm{KL}(q \,\|\, f)$ on 300 questions with $0 < q \leq 0.05$. Continuous: nCRPS on 300 questions and its excess over the score at the replay distribution's own five quantiles. Natural conditional: turn-2 excess Brier score against $p(Y \mid X)$ on 400 cells, and the gain over keeping the turn-1 forecast. Parsed: readable answers over the five scored sets (1,800 items). Cost: OpenRouter USD for all of the model's calls.

Model	ECI	Mid-range	Tails	Continuous		Natural conditional		Parsed	Cost
		excess Brier	bits	nCRPS	excess	excess Brier	gain	(%)	($)
Fable	162.5	0.1659	0.206	0.792	0.641	0.1138	−0.0151	100.0	37.48
Opus 5	161.5	0.1527	0.223	0.789	0.638	0.1120	−0.0056	99.9	17.13
GPT-5.6 Sol	161.1	0.1171	0.398	0.552	0.402	0.1002	0.0159	100.0	7.24
GPT-5.5	158.6	0.1304	0.248	0.589	0.439	0.0944	0.0163	100.0	29.80
Gemini 3.7 Flash	156.8	0.1077	0.393	0.588	0.438	0.1214	−0.0021	100.0	3.40
GPT-5.6 Luna	156.2	0.1294	0.361	0.724	0.574	0.1058	−0.0023	99.9	0.66
Sonnet 5	155.9	0.1733	0.151	0.875	0.725	0.1128	−0.0018	100.0	5.23
DeepSeek V4 Flash	154.4	0.1614	0.237	0.688	0.537	0.1335	−0.0063	99.4	0.74
Gemini 3 Flash	150.7	0.1034	0.397	0.572	0.422	0.1138	−0.0095	100.0	2.24
GPT-5	150.0	0.1146	0.390	0.776	0.625	0.1063	−0.0006	99.9	6.64
o3	147.1	0.1169	0.317	0.715	0.565	0.0889	0.0067	99.9	12.32
GPT-5.4 Nano	146.5	0.1644	0.256	0.796	0.646	0.1123	−0.0071	100.0	0.97
o4-mini	146.5	0.1679	0.231	0.848	0.696	0.1455	0.0054	99.2	5.32
GPT-5 Mini	145.8	0.1525	0.289	1.040	0.889	0.1170	−0.0011	100.0	1.58
Gemini 3.1 Flash Lite	144.7	0.1316	0.303	0.743	0.593	0.1213	0.0009	100.0	0.99
Qwen3.5 Flash	144.1	0.1298	0.704	0.712	0.563	0.1544	−0.0106	99.8	1.15
Haiku 4.5	142.9	0.1110	0.442	0.833	0.683	0.1031	−0.0076	100.0	6.83
GPT-5 Nano	140.8	0.1917	0.180	1.280	1.130	0.1115	−0.0016	100.0	0.23
Kimi K2	140.4	0.1206	0.405	0.618	0.468	0.1124	−0.0054	99.7	6.82
Gemini 2.5 Flash	140.3	0.1542	0.453	0.827	0.676	0.1760	−0.0178	99.9	5.26
Qwen3 235B	139.6	0.1380	0.342	0.955	0.806	0.1115	0.0146	99.1	5.89
GPT-4.1	137.5	0.1361	0.413	0.866	0.716	0.1384	0.0011	100.0	12.92
DeepSeek V3	133.0	0.1310	0.384	0.928	0.777	0.1204	−0.0005	100.0	2.46
Llama 4 Scout	130.5	0.1933	0.237	1.060	0.909	0.1234	−0.0093	100.0	0.84

Table 23: FreeCiv mirror set: excess Brier score per model on 50 questions with $0.95 \leq q < 1$, pooled and by resolution turn (10 questions each); lower is better. The 95% interval is the item-level bootstrap interval. A constant 0.5 scores 0.233.

Model	ECI	Excess Brier	95% CI	By horizon (turn)				
				90	120	150	180	210
Fable	162.5	0.5080	[0.425, 0.590]	0.550	0.526	0.509	0.574	0.381
Opus 5	161.5	0.4441	[0.365, 0.519]	0.433	0.419	0.490	0.469	0.409
GPT-5.6 Sol	161.1	0.2684	[0.203, 0.330]	0.325	0.188	0.321	0.270	0.238
GPT-5.5	158.6	0.3810	[0.310, 0.457]	0.466	0.330	0.429	0.347	0.334
Gemini 3.7 Flash	156.8	0.2711	[0.210, 0.339]	0.330	0.239	0.300	0.275	0.211
GPT-5.6 Luna	156.2	0.3788	[0.311, 0.446]	0.427	0.235	0.418	0.350	0.464
Sonnet 5	155.9	0.5459	[0.481, 0.610]	0.391	0.532	0.565	0.597	0.644
DeepSeek V4 Flash	154.4	0.4194	[0.340, 0.498]	0.407	0.287	0.532	0.448	0.423
Gemini 3 Flash	150.7	0.2630	[0.202, 0.327]	0.352	0.185	0.272	0.262	0.244
GPT-5	150.0	0.3146	[0.257, 0.373]	0.346	0.253	0.368	0.309	0.298
o3	147.1	0.3354	[0.274, 0.397]	0.359	0.257	0.355	0.395	0.311
GPT-5.4 Nano	146.5	0.5527	[0.473, 0.625]	0.541	0.443	0.580	0.575	0.624
o4-mini	146.5	0.4981	[0.428, 0.572]	0.497	0.404	0.621	0.564	0.406
GPT-5 Mini	145.8	0.4242	[0.348, 0.497]	0.357	0.324	0.509	0.434	0.497
Gemini 3.1 Flash Lite	144.7	0.3913	[0.323, 0.466]	0.283	0.380	0.364	0.490	0.439
Qwen3.5 Flash	144.1	0.3146	[0.243, 0.382]	0.253	0.293	0.299	0.388	0.339
Haiku 4.5	142.9	0.3680	[0.306, 0.432]	0.294	0.310	0.434	0.440	0.361
GPT-5 Nano	140.8	0.6645	[0.606, 0.723]	0.684	0.662	0.572	0.721	0.684
Kimi K2	140.4	0.3402	[0.275, 0.405]	0.285	0.207	0.431	0.361	0.418
Gemini 2.5 Flash	140.3	0.3093	[0.234, 0.388]	0.259	0.226	0.378	0.307	0.376
Qwen3 235B	139.6	0.4712	[0.398, 0.541]	0.481	0.360	0.393	0.649	0.472
GPT-4.1	137.5	0.2866	[0.221, 0.360]	0.290	0.314	0.371	0.323	0.135
DeepSeek V3	133.0	0.3976	[0.322, 0.473]	0.353	0.259	0.511	0.523	0.342
Llama 4 Scout	130.5	0.6067	[0.525, 0.689]	0.581	0.763	0.431	0.623	0.635

Table 24: FreeCiv mid-range excess Brier score (150 questions per turn) and tail excess bits (60 per turn) per model by resolution turn; lower is better.

Model	ECI	Mid-range excess Brier, by horizon (turn)					Tail excess bits, by horizon (turn)				
		90	120	150	180	210	90	120	150	180	210
Fable	162.5	0.1671	0.1794	0.1643	0.1589	0.1595	0.175	0.208	0.186	0.171	0.288
Opus 5	161.5	0.1508	0.1674	0.1576	0.1467	0.1411	0.222	0.236	0.168	0.197	0.294
GPT-5.6 Sol	161.1	0.1212	0.1139	0.1163	0.1182	0.1157	0.273	0.328	0.317	0.413	0.659
GPT-5.5	158.6	0.1461	0.1378	0.1321	0.1216	0.1147	0.171	0.225	0.228	0.240	0.376
Gemini 3.7 Flash	156.8	0.1019	0.1168	0.1207	0.1013	0.0977	0.299	0.360	0.367	0.365	0.576
GPT-5.6 Luna	156.2	0.1353	0.1391	0.1384	0.1126	0.1218	0.259	0.256	0.390	0.413	0.486
Sonnet 5	155.9	0.1602	0.1779	0.1809	0.1719	0.1758	0.149	0.145	0.151	0.144	0.166
DeepSeek V4 Flash	154.4	0.1534	0.1677	0.1705	0.1510	0.1646	0.209	0.192	0.214	0.215	0.355
Gemini 3 Flash	150.7	0.1028	0.1016	0.1137	0.1013	0.0977	0.318	0.457	0.331	0.400	0.482
GPT-5	150.0	0.1152	0.1189	0.1237	0.1167	0.0986	0.283	0.326	0.370	0.451	0.519
o3	147.1	0.1119	0.1248	0.1221	0.1143	0.1116	0.284	0.290	0.277	0.290	0.443
GPT-5.4 Nano	146.5	0.1526	0.1661	0.1719	0.1639	0.1673	0.260	0.288	0.240	0.232	0.262
o4-mini	146.5	0.1680	0.1734	0.1875	0.1607	0.1496	0.215	0.166	0.273	0.233	0.269
GPT-5 Mini	145.8	0.1429	0.1567	0.1647	0.1478	0.1503	0.221	0.212	0.315	0.344	0.352
Gemini 3.1 Flash Lite	144.7	0.1184	0.1235	0.1396	0.1343	0.1421	0.237	0.346	0.272	0.301	0.360
Qwen3.5 Flash	144.1	0.1315	0.1457	0.1317	0.1124	0.1274	0.618	0.732	0.632	0.963	0.573
Haiku 4.5	142.9	0.1056	0.1155	0.1257	0.1063	0.1020	0.465	0.414	0.349	0.480	0.501
GPT-5 Nano	140.8	0.1679	0.1976	0.1843	0.2002	0.2085	0.229	0.165	0.195	0.139	0.171
Kimi K2	140.4	0.1078	0.1281	0.1279	0.1169	0.1224	0.297	0.398	0.424	0.413	0.495
Gemini 2.5 Flash	140.3	0.1453	0.1534	0.1567	0.1594	0.1560	0.233	0.401	0.484	0.594	0.554
Qwen3 235B	139.6	0.1249	0.1421	0.1486	0.1390	0.1355	0.379	0.294	0.292	0.371	0.374
GPT-4.1	137.5	0.1214	0.1576	0.1425	0.1249	0.1341	0.291	0.301	0.444	0.402	0.628
DeepSeek V3	133.0	0.1243	0.1464	0.1316	0.1246	0.1283	0.344	0.331	0.373	0.395	0.479
Llama 4 Scout	130.5	0.1915	0.2143	0.1778	0.1796	0.2032	0.238	0.146	0.334	0.302	0.166

Table 25: FreeCiv continuous nCRPS (60 questions per turn) and natural-conditional turn-2 excess Brier score (100 cells per turn) per model by resolution turn; lower is better. Natural conditionals exist only for turns 120 to 210.

Model	ECI	Continuous nCRPS, by horizon (turn)					Natural-conditional excess Brier			
		90	120	150	180	210	120	150	180	210
Fable	162.5	0.389	0.781	0.760	0.950	1.079	0.1227	0.1224	0.1275	0.0826
Opus 5	161.5	0.375	0.747	0.783	0.986	1.053	0.1173	0.1172	0.1246	0.0888
GPT-5.6 Sol	161.1	0.272	0.475	0.504	0.681	0.830	0.0845	0.0995	0.1060	0.1108
GPT-5.5	158.6	0.292	0.528	0.550	0.714	0.861	0.0875	0.1022	0.0954	0.0926
Gemini 3.7 Flash	156.8	0.296	0.524	0.572	0.728	0.819	0.1207	0.1272	0.1441	0.0934
GPT-5.6 Luna	156.2	0.408	0.809	0.723	0.812	0.869	0.1030	0.1101	0.1041	0.1060
Sonnet 5	155.9	0.473	0.923	0.901	1.022	1.058	0.1079	0.1200	0.1206	0.1029
DeepSeek V4 Flash	154.4	0.363	0.737	0.644	0.810	0.883	0.1416	0.1289	0.1251	0.1377
Gemini 3 Flash	150.7	0.288	0.489	0.476	0.662	0.945	0.1302	0.1157	0.1248	0.0846
GPT-5	150.0	0.361	0.793	0.788	0.952	0.984	0.0995	0.1099	0.1089	0.1069
o3	147.1	0.392	0.777	0.748	0.879	0.781	0.0806	0.0919	0.0989	0.0843
GPT-5.4 Nano	146.5	0.413	0.894	0.790	0.930	0.953	0.1138	0.1213	0.1082	0.1058
o4-mini	146.5	0.443	0.876	0.813	0.983	1.138	0.1333	0.1487	0.1348	0.1651
GPT-5 Mini	145.8	0.530	1.064	1.052	1.254	1.299	0.1104	0.1208	0.1152	0.1218
Gemini 3.1 Flash Lite	144.7	0.391	0.715	0.783	0.879	0.946	0.1279	0.1200	0.1342	0.1030
Qwen3.5 Flash	144.1	0.393	0.698	0.694	0.939	0.845	0.1454	0.1546	0.1494	0.1681
Haiku 4.5	142.9	0.430	0.900	0.897	0.999	0.937	0.1110	0.1050	0.1007	0.0956
GPT-5 Nano	140.8	0.710	1.288	1.239	1.511	1.653	0.1032	0.1227	0.1017	0.1184
Kimi K2	140.4	0.313	0.614	0.563	0.710	0.891	0.1097	0.1215	0.1210	0.0977
Gemini 2.5 Flash	140.3	0.438	0.843	0.827	1.020	1.004	0.2003	0.1922	0.1672	0.1442
Qwen3 235B	139.6	0.509	1.036	0.930	1.136	1.182	0.1222	0.1042	0.1011	0.1185
GPT-4.1	137.5	0.448	0.883	0.858	1.031	1.110	0.1307	0.1440	0.1456	0.1334
DeepSeek V3	133.0	0.446	0.925	0.855	1.155	1.257	0.1277	0.1202	0.1210	0.1125
Llama 4 Scout	130.5	0.573	1.027	0.975	1.295	1.429	0.1111	0.1523	0.1090	0.1211

Using a private export of the aligned replays, we computed 95% percentile intervals from 100,000 resamples of 1,003 clusters defined by random-number-generator (RNG) tags shared across worlds. A sampled tag is counted the same number of times wherever it occurs. We match tags to each world's replay columns by identity, not by column position. The resampling includes all outcomes, including $X = \text{false}$; unconditional probabilities divide by the number of sampled replays in that world, and conditional probabilities divide by the number in which X is true. All conditional denominators stayed positive. We keep the 9,584 forecast pairs, 16 exclusions for missing pairs and score weights fixed: we average the available cells within each model and group, then give the 24 models equal weight. Replay-only gain intervals are $[-0.004, -0.000]$ overall, $[-0.059, -0.047]$ for C1 and $[+0.005, +0.008]$ for the other 344 cells. Independently resampling 1,000 columns within each world (also 100,000 draws) gives similar results. The overall upper endpoint is extremely close to zero; its nominal exclusion of zero should not determine the substantive conclusion. These intervals condition on the selected cells, worlds and observed forecasts; they do not account for variation across source worlds, model panels or repeated elicitation, or uncertainty from question selection. Distinct tags are treated as independent; dependence beyond shared tags is not established. We do not repeat the original selection by probability band or the substitutions used to construct the corpus.

Replay uncertainty further limits interpretation of movement. Reconstructing each cell's joint reveal/outcome counts and resampling that table 20,000 times preserves the dependence between its unconditional and conditional estimates. Cellwise paired 95% percentile intervals exclude zero for 228 of 400 shifts; 152 of the 174 estimated absolute shifts below 0.03 remain unresolved. These are unadjusted cellwise intervals, and comparisons based on the estimated shift or on whether its interval excludes zero are exploratory. C1 losses persist among cells whose intervals exclude zero. Aligned replay identities now permit the joint gain analysis above; the cellwise movement intervals remain a separate diagnostic. Movement relative to the estimated shift is also distinct from gain: writing $d = f_2 - f_1$ and $\Delta = q_c - p$, where $q_c = p(Y \mid X)$ and $p = p(Y)$, gain equals $2d(p - f_1) + 2d\Delta - d^2$. Thus a movement poorly matched to the shift can still improve the score by correcting initial forecast error. With one elicitation per condition, imprecision across cells does not tell us how much responses would vary if the model were asked again.

Table 26: FreeCiv natural conditionals per model by structural group: turn-2 excess Brier score against $p(Y \mid X)$ (lower is better) and gain over keeping the turn-1 forecast (higher is better). Groups: A irrelevant news (36 cells), B related weak news (36), C1 mechanical same-series (56), C2 persistence across series (172), D inferential (100).

Model	ECI	Turn-2 excess Brier, by block					Gain over not updating, by block				
		A	B	C1	C2	D	A	B	C1	C2	D
Fable	162.5	0.134	0.119	0.142	0.085	0.139	0.007	0.009	−0.073	−0.022	0.013
Opus 5	161.5	0.156	0.123	0.132	0.078	0.138	0.015	0.007	−0.082	−0.003	0.020
GPT-5.6 Sol	161.1	0.090	0.114	0.098	0.103	0.096	0.002	0.004	0.011	0.030	0.003
GPT-5.5	158.6	0.088	0.121	0.098	0.088	0.097	0.007	0.000	−0.014	0.035	0.010
Gemini 3.7 Flash	156.8	0.108	0.099	0.152	0.134	0.095	−0.000	0.005	−0.047	0.006	0.006
GPT-5.6 Luna	156.2	0.121	0.107	0.126	0.108	0.085	0.004	−0.007	−0.045	0.006	0.006
Sonnet 5	155.9	0.151	0.127	0.121	0.075	0.155	0.012	0.009	−0.068	0.002	0.020
DeepSeek V4 Flash	154.4	0.147	0.095	0.160	0.134	0.127	0.003	−0.002	−0.069	0.011	−0.008
Gemini 3 Flash	150.7	0.083	0.138	0.152	0.117	0.090	0.001	−0.018	−0.057	−0.001	0.001
GPT-5	150.0	0.123	0.138	0.125	0.100	0.089	0.001	−0.002	−0.039	0.010	0.002
o3	147.1	0.112	0.118	0.084	0.076	0.094	−0.009	0.002	−0.011	0.012	0.014
GPT-5.4 Nano	146.5	0.143	0.103	0.127	0.108	0.104	0.013	0.013	−0.072	−0.011	0.021
o4-mini	146.5	0.158	0.149	0.146	0.146	0.138	0.006	0.011	−0.019	0.006	0.015
GPT-5 Mini	145.8	0.166	0.146	0.142	0.105	0.095	0.007	−0.001	−0.047	0.009	0.005
Gemini 3.1 Flash Lite	144.7	0.117	0.161	0.164	0.105	0.113	0.015	−0.002	−0.058	0.016	0.004
Qwen3.5 Flash	144.1	0.231	0.178	0.180	0.155	0.103	0.006	0.011	−0.061	−0.011	0.005
Haiku 4.5	142.9	0.118	0.114	0.167	0.093	0.076	0.020	−0.010	−0.067	−0.005	0.012
GPT-5 Nano	140.8	0.141	0.156	0.139	0.086	0.114	0.014	0.004	−0.088	0.004	0.029
Kimi K2	140.4	0.130	0.128	0.129	0.104	0.106	0.019	−0.002	−0.058	−0.008	0.018
Gemini 2.5 Flash	140.3	0.165	0.124	0.232	0.163	0.190	0.005	0.002	−0.047	−0.013	−0.025
Qwen3 235B	139.6	0.124	0.119	0.141	0.106	0.097	0.026	0.016	−0.028	0.024	0.017
GPT-4.1	137.5	0.124	0.143	0.184	0.142	0.110	0.009	0.008	−0.053	0.019	−0.004
DeepSeek V3	133.0	0.103	0.167	0.153	0.108	0.113	0.014	−0.008	−0.044	0.012	−0.000
Llama 4 Scout	130.5	0.152	0.155	0.172	0.115	0.088	−0.000	0.029	−0.132	−0.010	0.044

Additional figures. Figure 9 shows each FreeCiv score by resolution turn and Figure 10 the turn-2 moves on the natural-conditional cells against the estimated replay shift.

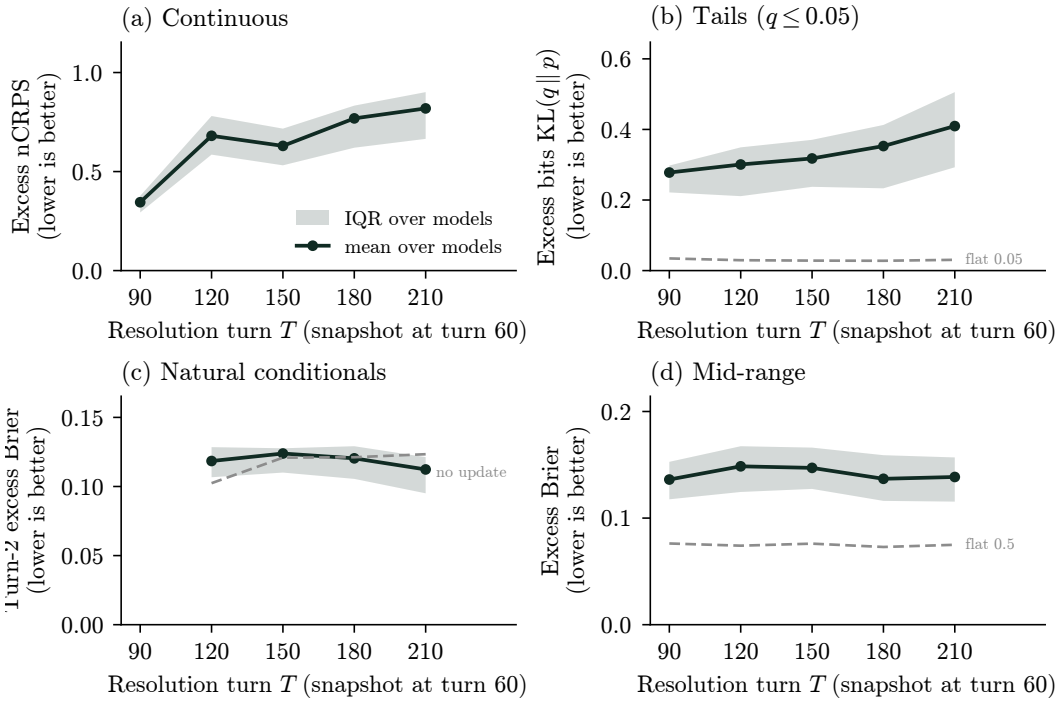


Figure 9: FreeCiv scores by resolution turn T from the turn-60 snapshot. The line is the mean over the 24 models and the band the interquartile range across models; lower is better. Dashed gray lines are reference forecasters: a constant 0.05 on tail questions, the unchanged turn-1 forecast on natural conditionals, and a constant 0.5 on the mid-range set. Mean excess nCRPS goes from 0.35 at $T = 90$ to 0.82 at $T = 210$, mean tail bits from 0.28 to 0.41, the mid-range excess Brier score from 0.136 to 0.139, and the natural-conditional excess Brier score from 0.118 at $T = 120$ to 0.112. Natural conditionals exist only for $T \geq 120$.

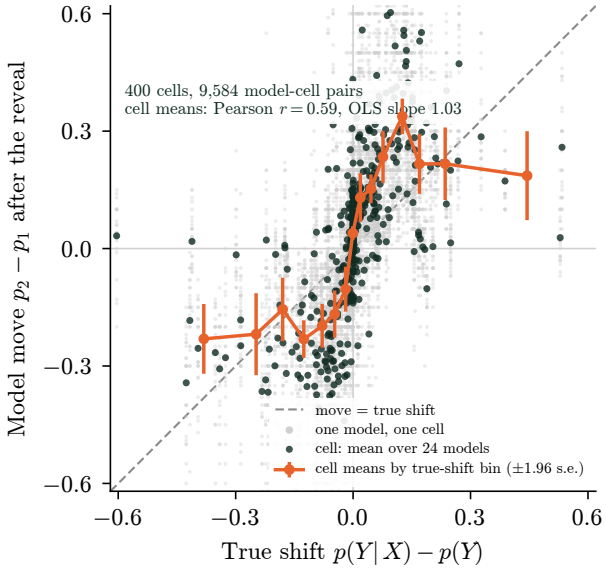


Figure 10: Natural conditionals: how forecasts move after one revealed fact. Each green point is one of the 400 cells: x is the estimated replay shift $p(Y \mid X) - p(Y)$, with the conditional probability using only replays satisfying the reveal, and y is the mean move $p_2 - p_1$ over available models. Gray points are the 9,584 model–cell pairs. Orange is the mean of cell means by bin of estimated shift, with 1.96 standard-error bars; these bars do not represent paired replay uncertainty. The dashed line marks a move equal to the estimated shift. Cell means track the estimated shift (Pearson $r = 0.59$, least-squares slope 1.03). Mean absolute movement over the 400 cell means is 0.17, 2.3 times the mean absolute estimated shift of 0.07; over model–cell pairs it is 0.18. The 2.3 ratio compares mean absolute magnitudes; it does not estimate a coefficient of proportional overshooting. Pooling the groups obscures their differences, and the replay shifts are themselves uncertain: C1 has negative mean gain, while the other groups have positive descriptive means (Table 26).

Truth-conditioned forecasts and updating. Figure 11 plots mean mid-range forecasts by replay-truth bin; it is not a forecast-binned reliability diagram, and Figure 10 above plots the move of each natural-conditional forecast against the estimated replay shift. Discrimination, the Pearson correlation between a model's forecasts and the truth over the 750 mid-range questions, runs from 0.09 (Llama 4 Scout) to 0.27 (Gemini 3.7 Flash) and correlates with ECI at $\rho = 0.72$ [0.48, 0.85] (model bootstrap, 10,000 resamples); the signed question-level mean of $f - q$ correlates with ECI at -0.07, and the excess Brier score correlates with the absolute bias at 0.87 and with discrimination at -0.39. Comparative questions, which ask whether one civilization exceeds another, show no bias (mean $f - q$ of $+0.00$) and the highest discrimination (0.48); event and threshold questions are under-forecast by 0.25 to 0.35 (Table 27). On the natural conditionals the turn-2 excess Brier score of a model correlates at 0.91 with the standard deviation of its turn-2 forecasts (0.21 for o3 to 0.35 for Gemini 2.5 Flash) and at -0.26 with its discrimination at turn 2 (0.16 to 0.33), which itself rises with ECI at 0.73.

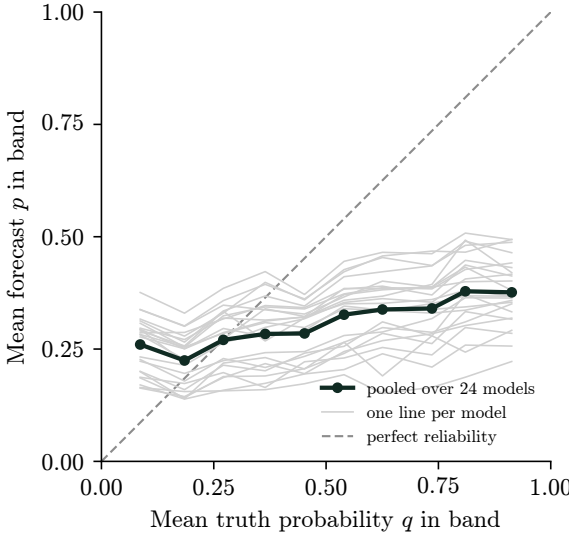


Figure 11: Mean FreeCiv mid-range forecast by replay-truth bin over 750 questions: mean forecast f against mean ground-truth probability q in ten equal-width intervals of q on $[0.05, 0.95]$. One gray line per model, green the pool over 24 models, dashed the identity line; its distance from a truth-binned curve is not a calibration decomposition. Pooled forecasts run from 0.26 in the lowest interval ($q = 0.09$) to 0.38 in the highest ($q = 0.91$); they lie above the truth in the two lowest intervals and below it in the eight highest. The question-level mean of $f - q$ is negative for 24 of 24 models (range -0.33 to -0.07).

Joint binary references and composition. Across the 1,100 selected binary questions, equally weighted per item (750 mid-range, 300 tail, 50 mirror), mean excess Brier is 0.135. Universal 0.5 scores 0.125, and nine of 24 models beat it. A leave-one-world-out ridge reference using only question-text features scores 0.095; it learns from replay labels in the other seven worlds, information the LLMs were not supplied. It is a supervised reference, not a matched zero-shot baseline or report-conditional oracle. All 24 model point estimates are worse under these archive weights. The fixed penalty is 10; no held-out band labels are inputs.

Under archive weights, grouping changes mean excess Brier from 0.138 to 0.135; equal mass for each band reverses the change (0.186 to 0.207). The two weighting schemes measure different averages. Neither describes the full population of candidate events before selection, and the improvement does not hold under both weighting schemes. Reused worlds, overlapping training folds, replay estimation and model generation limit uncertainty claims about the supervised reference.

Forecast-binned diagnostics on the selected mid-range set also differ from Figure 11: pooled rows with $f < 0.1$ average $f = 0.049$, $q = 0.436$ (3,189 rows), while $f \geq 0.9$ averages $f = 0.922$, $q = 0.699$ (261 rows). These repeated observations of models on the same questions describe

discrepancies in the selected archive; they are not independent evidence of miscalibration in deployment. Mean mid-range excess decomposes as $0.141 = 0.041 + 0.050 + 0.075 - 0.025$ (squared bias, forecast variance, truth variance, minus twice covariance); squared bias is about 29%, not all of the error.

Difficulty by question family. Table 27 gives the mean score of each family at each horizon, pooled over the 24 models. In the mid-range set the three worst families (any city destroyed, the number of government changes, and cities lost) hold 69 of 750 items and 14% of the pooled excess Brier score. In the tail set the three worst (a government in place at the horizon, a peace in effect, a wonder completed) hold 63 of 300 items and 36% of the pooled bits. In the continuous set the two technology-count families and world wonders hold 82 of 300 items and 58% of the pooled raw normalized CRPS. This is not their share of excess loss or evidence that these families drive its ECI association.

Table 27: FreeCiv difficulty by question family and horizon, pooled over the 24 models: mean excess Brier score (mid-range), mean excess bits (tail set) and mean normalized CRPS (continuous set); lower is easier. Families are sorted from hardest to easiest by the item-weighted mean over horizons.

Family	Items	T90	T120	T150	T180	T210	All
Mid-range: excess Brier score							
S7_any_destroyed	2	0.448	–	–	–	–	0.448
S4_gov_change_count	32	0.204	0.266	0.258	0.152	0.151	0.213
W4_lose_k	35	0.128	0.167	0.232	0.240	0.378	0.211
S3_wars_at_T	3	0.206	–	–	–	–	0.206
EX_government_at	46	0.085	0.177	0.278	0.199	0.194	0.191
NB1_threshold	48	0.207	0.194	0.170	0.190	0.180	0.187
W3_capture_k	38	0.071	0.165	0.214	0.234	0.244	0.180
NB6_event	7	0.091	0.393	–	–	–	0.177
W1_wonder_race	36	0.195	0.202	0.171	0.157	0.159	0.175
NW2_diplo	44	0.176	0.112	0.153	0.125	0.169	0.146
W2_directed_conquest	49	0.133	0.147	0.162	0.139	0.123	0.141
EX_tech_discovered	50	0.103	0.159	0.145	0.142	0.156	0.141
NB4_drawdown	44	0.092	0.116	0.162	0.133	0.153	0.139
NW1_war_at	49	0.126	0.124	0.098	0.156	0.142	0.130
NB1_value_threshold	51	0.121	0.164	0.118	0.143	0.103	0.129
W6_peace_at	12	0.101	0.096	0.335	–	0.077	0.117
NW5_wonder	34	0.112	0.109	0.109	0.111	0.102	0.108
S6_city_founding	34	0.135	0.112	0.087	0.078	0.092	0.102
S5_civ_wonders	43	0.155	0.112	0.098	0.082	0.059	0.101
W5_tech_lead	23	0.133	0.069	0.056	0.036	0.017	0.070
EX_comparative	53	0.068	0.110	0.038	0.066	0.052	0.067
NEW_civil_war	17	–	0.006	0.016	0.049	0.057	0.039
all families	750	0.136	0.149	0.147	0.137	0.139	0.141
Tail set: excess bits							
EX_government_at	22	0.770	0.567	0.704	0.482	0.614	0.624
W6_peace_at	20	0.523	0.572	0.505	0.564	0.556	0.546
NW5_wonder	21	0.178	0.250	0.419	0.665	0.974	0.543
NW1_war_at	15	0.543	0.367	0.287	0.384	0.411	0.395
NW4_survival	16	0.326	0.469	0.204	0.431	0.571	0.393
NW2_diplo	21	0.175	0.529	0.441	0.342	0.339	0.372
NB1_value_threshold	20	0.144	0.370	0.349	0.430	0.438	0.371
EX_tech_discovered	22	0.256	0.311	0.326	0.504	0.338	0.357
S6_city_founding	2	0.297	–	–	–	–	0.297
NB6_event	3	0.280	–	–	–	–	0.280
W4_lose_k	3	0.265	–	–	–	–	0.265
S4_gov_change_count	6	0.285	0.172	–	–	–	0.248
S5_civ_wonders	21	0.157	0.203	0.191	0.253	0.329	0.236
NB4_drawdown	20	0.173	0.207	0.318	0.184	0.260	0.233
W5_tech_lead	15	0.188	0.261	0.220	0.036	0.314	0.213
W3_capture_k	3	0.210	–	–	–	–	0.210
W2_directed_conquest	20	0.197	0.171	0.257	0.225	0.191	0.209
W1_wonder_race	21	0.158	0.160	0.151	0.231	0.250	0.193
EX_comparative	9	0.245	0.130	0.121	–	–	0.180
NEW_civil_war	20	0.065	0.090	0.039	0.047	0.049	0.057
all families	300	0.278	0.301	0.318	0.353	0.410	0.332
Continuous set: normalised CRPS							
P6_world_techs	20	2.037	3.931	3.962	4.029	4.736	3.463
P5_techs_at_T	31	0.781	1.357	1.728	2.029	1.795	1.546
NC14_world_wonders	31	0.156	0.338	0.706	1.063	1.504	0.763
NC5_world_captures	32	0.085	0.430	0.762	0.935	1.184	0.697
P2_civ_losses	31	0.103	0.280	0.289	0.860	0.704	0.456
S7_world_razings	33	0.154	0.317	0.500	0.574	0.648	0.451
NC1_value_at_T	60	0.225	0.395	0.435	0.489	0.691	0.447
P3_civ_founds	31	0.311	0.428	0.461	0.500	0.424	0.427
P1_civ_conquests	31	0.076	0.203	0.280	0.325	0.422	0.262
all families	300	0.415	0.807	0.784	0.964	1.036	0.801

C.4 CORRELATION WITH FORECASTBENCH

Table 28 repeats the nine correlations of Table 2 with the ForecastBench overall score (Karger et al., 2025a) in place of ECI. Seventeen of the 24 models have a ForecastBench score, ranging from 54.6 to 61.1. [2] Over the 17 models ECI and ForecastBench scores correlate at $\rho = 0.62$ [0.18, 0.88], $p = 0.008$. We note that for every of the 9 correlations, the point estimate is positive but the confidence interval does not exclude zero. Intervals are wider than in Table 2 because of the smaller number of models used.

Table 28: Spearman correlation ρ between the ForecastBench overall score and each of our 9 scores, over the 17 models covered by ForecastBench. The correlation is sign-adjusted so a positive value means higher ForecastBench score corresponds to higher scores. CI (models): 95% percentile bootstrap over models, with 2,000 resamples for the Starsim continuous score and 10,000 resamples for the remainder, at seed 2026; p is two-sided.

Simulation	Score	ρ	CI (models)	p
Micropolis	Excess Brier, mid-range	0.38	$[-0.20, 0.80]$	0.135
	Excess bits, tail	0.45	$[-0.03, 0.78]$	0.067
	Excess nCRPS	0.35	$[-0.22, 0.76]$	0.163
Starsim	Excess CRPS, interventional	0.45	$[-0.14, 0.88]$	0.071
	Excess CRPS, continuous	0.24	$[-0.35, 0.82]$	0.345
FreeCiv	Excess nCRPS	0.21	$[-0.32, 0.67]$	0.416
	Excess bits, tail	0.12	$[-0.38, 0.56]$	0.653
	Excess Brier, natural conditional	0.38	$[-0.18, 0.81]$	0.129
	Excess Brier, mid-range	0.28	$[-0.24, 0.67]$	0.279

C.5 COMBINED SCORE

Table 30 lists the latent skill θ_m of Section 6 for every model, and Figure 12 plots it against ECI. The logarithm lets cells with different loss scales share one axis. The correlation with ECI is 0.86 [0.69, 0.93]. ECI is itself a latent-ability fit across benchmarks (Ho et al., 2025), and this fit gives each observed log-loss cell equal nominal weight, not equal influence, so it is a descriptive summary of the nine score sets, not evidence of a single common latent skill. The fit is $\log \ell_{mc} = \delta_c - \theta_m + \varepsilon_{mc}$ by alternating least squares, omitting the one missing cell (DeepSeek V4 Flash on the Starsim interventional set) from the sums and θ_m centered at zero. Losses are floored at 10^{-4} before the logarithm; no cell reaches the floor. Every cell is an excess score. The residual standard deviation on the log scale is 0.35. The agreement between θ_m and each cell's own ranking, as a Spearman correlation, is 0.74 to 0.92 for the Micropolis and Starsim cells and 0.20 to 0.65 for the FreeCiv cells, across sets. Leaving out any one cell moves the correlation with ECI between 0.86 and 0.88. This fit excludes historical Starsim binary. Table 21 reports its ForecastBench association on 17 models, with separate model-panel and replay intervals.

Binary exclusion and panel sensitivity. The primary nine-cell fit keeps the 24-model roster and omits the one missing intervention cell from its sums. Using repaired continuous scores throughout, the ECI association is 0.863; refitting the 23 models with all cells gives 0.860, and restricting to the 22 models with all planned Starsim repetitions gives 0.874. The ten-cell historical-binary sensitivity gives 0.888 on 24 models. Exclusion is motivated by the visible fixed rule, not by the direction of these associations. These fits use different model panels and are not interchangeable; missing responses are not imputed. Paired bootstrap intervals over models for the change after removing binary scores include zero in the archived audit and do not include missing-response or replay uncertainty. Audit inputs and sensitivities are retained separately in the local supporting records.

Matched external panel and influence. On the same 17 models with ForecastBench scores, the ECI association is 0.794 rather than the 24-model 0.862. Six of the eight highest-ECI models lack

[2] The seven without are Opus 5, Fable, GPT-5.6 Sol, GPT-5.6 Luna, Gemini 3.7 Flash, DeepSeek V4 Flash and Qwen3.5 Flash

ForecastBench scores. Table 29 reports both external associations and their paired difference using 10,000 shared model resamples, seed 2026, with fitted scores held fixed. The difference interval includes zero; this is insufficient evidence of a difference, not equivalence.

Leaving one cell out keeps aggregate ECI correlation within 0.86–0.88 but moves individual ranks by up to five places. Cell log-loss standard deviations range from 0.149 to 0.892, so nominally equal weights do not give equal influence. Equal total weight per simulation gives ECI $\rho = 0.878$, rank agreement 0.986 with the primary fit and maximum movement of four places. These are exploratory aggregate sensitivities. FreeCiv mid-range and tail ranks correlate -0.703, underscoring the limits of a common-skill interpretation.

Table 29: Nine-cell combined score for the same 17 models with both external benchmark scores. Intervals resample the same model indices for both correlations; they omit replay and external-score measurement uncertainty.

Comparator / contrast	ρ or difference	Models, 95%
ECI	0.794	$[0.502, 0.939]$
ForecastBench	0.374	$[-0.216, 0.815]$
ECI minus ForecastBench	0.420	$[-0.028, 0.903]$

Table 30: Latent skill θ_m per model from the additive log-loss model, centered at zero, higher is better. Sorted by θ_m.

Model	ECI	θ
GPT-5.6 Sol	161.1	$+0.42$
GPT-5.5	158.6	$+0.42$
Opus 5	161.5	$+0.32$
DeepSeek V4 Flash	154.4	$+0.26$
Fable	162.5	$+0.25$
o3	147.1	$+0.24$
Gemini 3.7 Flash	156.8	$+0.22$
GPT-5.6 Luna	156.2	$+0.20$
Sonnet 5	155.9	$+0.18$
Gemini 3.1 Flash Lite	144.7	$+0.10$
Gemini 3 Flash	150.7	$+0.08$
GPT-5.4 Nano	146.5	$+0.07$
GPT-5	150.0	$+0.06$
Kimi K2	140.4	-0.07
GPT-4.1	137.5	-0.08
o4-mini	146.5	-0.09
GPT-5 Mini	145.8	-0.12
Haiku 4.5	142.9	-0.19
Gemini 2.5 Flash	140.3	-0.22
Llama 4 Scout	130.5	-0.36
Qwen3 235B	139.6	-0.39
DeepSeek V3	133.0	-0.42
GPT-5 Nano	140.8	-0.43
Qwen3.5 Flash	144.1	-0.46

D Ablations and diagnostics

This appendix reports prompt ablations, the Micropolis knowledge test and the FreeCiv no-news control. The effort check is summarized in Section A.5; the historical Starsim hold set is defined in Section C.2.

D.1 Questions per prompt

Micropolis. Eight models (Llama 4 Scout, DeepSeek V3, GPT-4.1, GPT-5 Mini, Gemini 3 Flash Preview, Haiku 4.5, Gemini 3.7 Flash and Fable) answered binary questions for five cities (bruce,

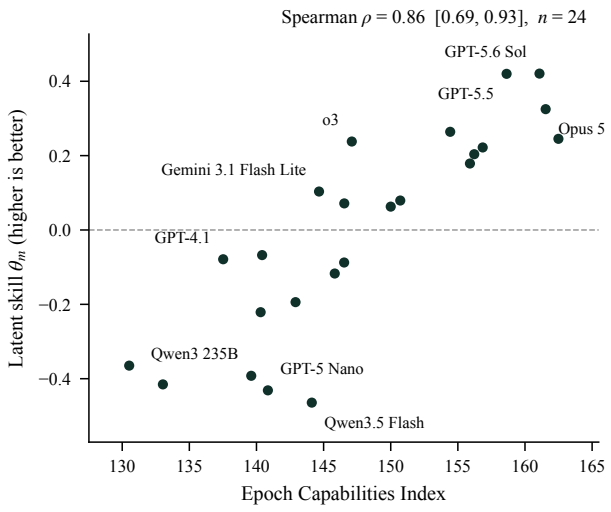


Figure 12: Latent skill θ_m against the Epoch Capabilities Index. Brackets: 95% percentile bootstrap over models, 10,000 resamples, seed 2026.

Table 31: Latent skill θ_m from the combined score and each model's rank on every cell, where rank 1 is the lowest loss and tied models share the average rank. "Avg Rank" ranks the mean of a model's cell ranks. Cell shading runs from green for the best value in a column, through white, to purple for the worst. DeepSeek V4 Flash has no Starsim interventional score; the other 23 models are ranked on that set. Its "Avg Rank" uses eight available cell ranks; its θ_m is fitted using eight available loss cells. Sorted by θ_m.

			Micropolis			FreeCiv				Starsim	
Model	θ	Avg Rank	Mid-range	Tail	Cont.	Mid-range	Tail	Cont.	Nat. cond.	Cont.	Interv.
GPT-5.6 Sol	+0.42	2	1	3	3	6	19	1	3	5	1
GPT-5.5	+0.42	1	4	1	2	10	8	4	2	7	3
Opus 5	+0.32	3	3	5	4	16	4	12	9	3	4
DeepSeek V4 Flash	+0.26	6	2	4	12	18	6	6	20	2	–
Fable	+0.25	5	8	6	1	20	3	13	13	4	8
o3	+0.24	4	6	10	9	5	12	8	1	6	5
Gemini 3.7 Flash	+0.22	7	5	2	6	2	17	3	18	9	17
GPT-5.6 Luna	+0.20	8	11	13	8	8	14	9	5	11	2
Sonnet 5	+0.18	11.5	9	12	5	22	1	19	12	12	6
Gemini 3.1 Flash Lite	+0.10	11.5	12	11	10	12	11	10	17	8	7
Gemini 3 Flash	+0.08	10	7	15	11	1	18	2	14	14	15
GPT-5.4 Nano	+0.07	13	15	16	15	19	9	14	10	1	18
GPT-5	+0.06	9	10	8	7	4	16	11	6	17	9
Kimi K2	−0.07	14	13	20	18	7	20	5	11	16	16
GPT-4.1	−0.08	17	20	9	17	13	21	18	21	15	13
o4-mini	−0.09	18	21	17	19	21	5	17	22	10	20
GPT-5 Mini	−0.12	16	17	7	20	15	10	22	15	19	12
Haiku 4.5	−0.19	15	14	19	16	3	22	16	4	20	14
Gemini 2.5 Flash	−0.22	21.5	23	22	14	17	23	15	24	13	10
Llama 4 Scout	−0.36	24	19	18	21	24	7	23	19	24	19
Qwen3 235B	−0.39	19	16	21	22	14	13	21	8	22	21
DeepSeek V3	−0.42	20	18	23	13	11	15	20	16	21	23
GPT-5 Nano	−0.43	21.5	22	14	24	23	2	24	7	23	22
Qwen3.5 Flash	−0.46	23	24	24	23	9	24	7	23	18	11

Table 32: Spearman correlation between the model rankings on θ_m and on each cell of Table 31, and between every pair of cells. Every ranking puts the best model first, so a positive value means two rankings agree. Correlations with the Starsim interventional set use the 23 models that have it. Cell shading runs from green at $+1$, through white at 0, to purple at -1.

		Micropolis			FreeCiv				Starsim	
	θ_m	Mid-range	Tail	Cont.	Mid-range	Tail	Cont.	Nat. cond.	Cont.	Interv.
θ_m		.92	.82	.87	.20	.27	.65	.39	.85	.74
Micropolis mid-range	.92		.76	.82	.37	.21	.66	.50	.66	.61
Micropolis tail	.82	.76		.68	.11	.35	.40	.30	.60	.58
Micropolis cont.	.87	.82	.68		.27	.18	.55	.38	.68	.71
FreeCiv mid-range	.20	.37	.11	.27		−.70	.66	.35	.00	.24
FreeCiv tail	.27	.21	.35	.18	−.70		−.35	.13	.27	.00
FreeCiv cont.	.65	.66	.40	.55	.66	−.35		.18	.57	.53
FreeCiv nat. cond.	.39	.50	.30	.38	.35	.13	.18		.12	.40
Starsim cont.	.85	.66	.60	.68	.00	.27	.57	.12		.60
Starsim interv.	.74	.61	.58	.71	.24	.00	.53	.40	.60	

haight, freds, happisle and joffburg), with 1, 2, 4, 7–8, 14–15, 25, 50 or 100 questions per prompt. By 7–8 (resp 14–15) we mean some prompts in those groups had 7 (resp 14) question while others had 8 (resp 15) — this was necessary because the 100 questions could not be evenly divided at this size. Table 33 reports the scores for each batch size, parse rates, costs and ECI associations; 50 is the main-run setting.

Tail loss falls from 0.241 excess bits at one question per prompt to 0.154 at two and 0.137 at four; the paired one-to-four change is -0.104 $[-0.20, -0.04]$. Mid-range changes are smaller and their paired-model intervals mostly include zero; the largest is -0.0153 $[-0.03, -0.00]$ at 7–8 questions. The tail loss with four questions minus that with 50 is $+0.038$ bits, paired-model 95% interval $[0.010, 0.070]$, versus mid-range $+0.00049$ $[-0.0058, 0.0070]$. Costs fall approximately as $1/n$. The macro-reported comparisons use 10,000 paired model resamples; the additional four-versus-50 comparisons use 3,000, seed 20260923. Each draw is shared across settings. These exploratory comparisons do not establish equivalence across batch sizes. Positive ECI associations do not imply stable rankings: rank correlations between settings reach 0.43, with movements up to four places.

The answer-format examples are 0.65 and 0.03 (Section B.1). GPT-5 Mini answers exactly 0.65 on 24.5% of one-question prompts and 1.9% of two-question prompts; GPT-4.1 also frequently reproduces the example block. These are finished answers, not parsing errors. Excluding 0.65 changes the pooled one-question tail score from 0.241 to 0.208 bits; a batching difference remains. Excluding both examples gives one-/two-question means 0.215/0.151, versus 0.208/0.139 after excluding only 0.65 and 0.241/0.154 without exclusions. Deletion changes the sample, can remove useful low forecasts, and does not identify copying or a causal mechanism. Tail forecast-minus-truth bias falls from $+0.151$ at one question to $+0.096$ at four and $+0.072$ at 50, while forecast–truth correlations range from 0.60 to 0.65.

FreeCiv. Table 34 compares one-question prompts on September 9 with the reported grouped run of September 20. Tail loss falls while mid-range, mirror and continuous losses rise. ECI correlations change by at most 0.06, but per-model rank correlations are 0.68–0.85 and maximum rank movements are 13, 14, 7 and 12 places respectively for mid-range, tail, mirror and continuous. On 231 overlapping mid-range questions, mean forecasts fall from 0.42 to 0.33 against mean truth 0.49. The dates, prompt context and two requested hosts also differ, so the comparison does not isolate an effect of batching or a mechanism specific to rare events.

A first grouped run on the same day shuffled question sets and used 50 questions for both kinds. Scores were within 0.013 and rank correlations 0.90–0.95, but o4-mini declined four of eight continuous prompts containing 38–48 questions; the reported run therefore caps continuous prompts at 20. Binary prompts are capped at 50 rather than 100 to limit long-context demands; these comparisons do not establish an optimal batch size.

Table 33: The questions-per-prompt ablation by setting: the number of questions per prompt and the number of prompts needed per model; the pooled mid-range excess Brier score and tail excess bits (means of per-model means); the lowest parse rate over the 8 models; the total cost in dollars over all 8 models; and the sign-adjusted Spearman correlation of each score with ECI over the 8 models. †: the paper's own run, whose prompts this setting reproduced.

Per model		Pooled score		Parsed	Total cost	ρ with ECI	
Q/prompt	Prompts	Mid-range	Tail	(%)	($)	Mid-range	Tail
1	1,000	0.0776	0.241	97.9	52.53	0.60	0.60
2	500	0.0758	0.154	96.4	28.70	0.67	0.69
4	250	0.0675	0.137	90.4	16.26	0.86	0.52
7–8	130	0.0623	0.138	86.2	10.02	0.95	0.74
14–15	70	0.0648	0.131	94.2	6.70	0.86	0.67
25	40	0.0654	0.095	100.0	4.75	0.79	0.90
50†	20	0.0670	0.100	100.0	2.83	0.81	0.60
100	10	0.0667	0.134	100.0	2.17	0.81	0.88

Table 34: FreeCiv scores under one question per prompt (September 9, 2026) and under the grouped prompts of September 20, mean over the 24 models, with the sign-adjusted Spearman correlation with ECI and the rank correlation of the per-model scores between the two runs. Lower scores are better.

	One per prompt		Grouped		
Set	Mean	ρ	Mean	ρ	Rank corr.
Mid-range, excess Brier	0.129	0.12	0.141	0.18	0.72
Tail, excess bits	0.556	0.35	0.332	0.34	0.68
Mirror, excess Brier	0.294	0.08	0.406	0.10	0.85
Continuous, excess nCRPS	0.524	0.61	0.648	0.59	0.70

D.2 World report and prompt variants (Micropolis)

We compared 11 continuous-prompt configurations using 24 models at the main reasoning settings and 15 cities (600 questions, 30 prompts per model). Table 35 specifies the history, snapshot ages, preamble, labels, order and batch size; `20yhist` is the main run. Scores use the common normalization described in Section A.1. Some contrasts change several features.

The variants change five things. The *history table* of the report is sampled every 4 turns (`long`), every 48 turns, one row per game year (`yfreq`), cut to its last two yearly rows (`last-2y`), or removed, so that the model sees the current state and the events alone (`snapshot-only`). The *preamble* is either preamble 1, which assigns the forecaster's role and asks for calibrated percentiles, or preamble 2b, which adds that service funding is fixed at 100% and never decays, that CRPS scores the answer so that intervals too narrow and too wide both lose, and that the five percentiles must be non-decreasing (preamble2b, shown in Section B.1). A question is *labeled* by a number, the questions ordered by turn, or by its metric and turn, the questions grouped by metric (`semantic`). One variant puts one metric's four horizons in a *prompt of its own*, five prompts per snapshot instead of one (`smallbatch`). And the *snapshot* moves from game years 5 and 10 to years 20 and 30 (`20yhist`), 30 and 40 (`semlonger`, `numlonger`) or 100 and 118 (`100yhist`), with the history sampled every 24 turns to keep the history table to a manageable length. The four variants built on preamble 1, and `preamble2` itself, also print the report's three funding-effectiveness lines, which the later variants drop once the preamble states that funding is fixed.

Among same-age configurations, removing history produces the largest loss increase; two yearly rows do not recover it. Yearly versus 4-turn history sampling is unresolved. Preamble 2b improves on preamble 1, but bundles several instructions. Semantic labels increase parsing failures, and small batches triple costs without a resolved score improvement (Table 36). These contrasts compare whole configurations; the shared-answer checks below address availability, not causal identification.

Table 35: The 11 prompt variants of the Micropolis continuous set. "vs" is the comparison configuration, which may differ in several features. "Snapshot" is in game years and "History" the sampling interval of the report's history table, with $*$ where the report also prints the road, police and fire funding-effectiveness lines. Preamble 1 assigns the role and asks for calibrated percentiles; preamble 2b adds that funding is fixed at 100% and that CRPS scores the answer. "Label, order" says how a question is named and whether the questions are ordered by turn or grouped by metric; "Per prompt" is the number of questions one prompt carries. †: the main run's prompts. Every variant runs 15 cities with disasters on, 24 models and horizons of 3, 5, 7 and 10 game years. The `100yhist` snapshots are turns 4,800 and 5,670.

Variant	vs	Snapshot	History	Preamble	Label, order	Per prompt
`snapshot-only`	`long`	5, 10	none*	1 (snapshot)	number, turn	20
`last-2y`	`yfreq`	5, 10	48 turns, last 2 rows*	1	number, turn	20
`yfreq`	`long`	5, 10	48 turns*	1	number, turn	20
`long`	–	5, 10	4 turns*	1	number, turn	20
`preamble2`	`long`	5, 10	4 turns*	2b	number, turn	20
`semantic`	`preamble2`	5, 10	4 turns	2b	metric@turn, metric	20
`smallbatch`	`preamble2`	5, 10	4 turns	2b	number, metric	4
`20yhist`†	`numlonger`	20, 30	24 turns	2b	number, metric	20
`semlonger`	`semantic`	30, 40	24 turns	2b	metric@turn, metric	20
`numlonger`	`semlonger`	30, 40	24 turns	2b	number, metric	20
`100yhist`	`numlonger`	100, 118	24 turns	2b	number, metric	20

Table 36: Results for each prompt variant: the reference configuration it is compared against (Table 35 gives the settings and snapshots); the pooled excess nCRPS (mean of per-model means); its difference from that reference, paired across the 24 models, with the 95% interval of that paired bootstrap; the share (%) of the variant's forecasts that did not parse, over all models; the sign-adjusted Spearman correlation of the per-model means with ECI, with a 95% bootstrap interval over models; and the cost of the run in dollars over all models. †: the main run, using its cached answers.

Variant	vs	Excess nCRPS	Δ vs parent	Unparsed	ρ with ECI	Cost ($)
`snapshot-only`	`long`	0.115	$+0.051$ [0.030, 0.074]	0.4	0.68 [0.36, 0.85]	7.30
`last-2y`	`yfreq`	0.119	$+0.054$ [0.040, 0.070]	0.8	0.66 [0.32, 0.83]	7.36
`yfreq`	`long`	0.065	0.000 [-0.003, 0.004]	0.1	0.76 [0.47, 0.90]	7.56
`long`	–	0.065	–	1.1	0.83 [0.61, 0.92]	8.89
`preamble2`	`long`	0.055	-0.009 [-0.015, -0.004]	0.4	0.86 [0.63, 0.96]	9.29
`semantic`	`preamble2`	0.054	-0.002 [-0.005, 0.002]	1.8	0.78 [0.51, 0.90]	9.15
`smallbatch`	`preamble2`	0.058	$+0.003$ [-0.001, 0.007]	0.3	0.77 [0.48, 0.91]	28.25
`20yhist`†	`numlonger`	0.039	$+0.005$ [0.003, 0.006]	0.6	0.82 [0.56, 0.92]	8.47
`semlonger`	`semantic`	0.034	-0.020 [-0.024, -0.016]	2.7	0.79 [0.51, 0.91]	9.36
`numlonger`	`semlonger`	0.034	0.000 [-0.003, 0.003]	0.6	0.80 [0.57, 0.91]	8.81
`100yhist`	`numlonger`	0.020	-0.014 [-0.021, -0.002]	0.6	0.64 [0.26, 0.85]	11.34

Comparisons across snapshot ages change the prediction targets as well as the prompts: losses fall from 0.054 at years 5/10 to 0.034 at years 30/40 and 0.020 at years 100/118. All variants use the main run's city-and-metric scales through year 20 (Section A.1); lower normalized losses do not isolate a dynamical explanation or prompt effect.

The correlation with ECI is present in every variant, between 0.64 (`100yhist`) and 0.86 (`preamble2`), and no interval reaches zero (Table 36). It is weakest where the report says least (`snapshot-only`, `last-2y`) and where the questions are easiest (`100yhist`), and strongest in the 4-turn-history variants at years 5 and 10. Positive ECI associations do not imply stable rankings: pairwise variant rank correlations reach 0.607, with a maximum 13-place shift. Comparisons of semantic/preamble2 and smallbatch/preamble2 at the same snapshot ages, restricted to answers valid in both configurations, give 0.870 and 0.842.

The post-hoc median-range rule in Section A.1 was motivated by three pathological responses among 10,800 calls (158,400 question-answer opportunities), including incorrect traffic units and population values repeated as answers for other metrics. Before the rule, those replies moved one variant's mean from 0.065 to 216. The rule removes 53 answer entries within 1,353 total invalid opportunities, leaving 157,047 valid answers; it removes none from the main run. These are filtering choices, not evidence that missing answers are random. The selected main configuration uses preamble 2b, numbered questions grouped by metric, one prompt per snapshot, 24-turn history sampling and years 20/30.

D.3 What models know about the Micropolis engine

We test whether a score that reflects both knowledge and abstention is associated with Micropolis forecasting. To probe how much the models know about the engine, we wrote a test of 206 statements about MicropolisCore, each to be answered True, False or Unknown: 94 are true and 112 false. Most come in 84 pairs, a true statement and a false twin that changes one thing about it, a number, a direction or a mechanism, so that the false statements are as specific as the true ones. A further 20 statements describe mechanics of later SimCity titles that Micropolis does not have, such as water pipes, subways or city ordinances (we refer to these as *honeypot* questions), and the share a model marks True measures how much of a neighboring game it brings along. Each statement carries a difficulty: level 0 (28 statements) for facts anyone who has seen the game would know, level 1 (91) for facts from playing it or reading its manual, and level 2 (87) for facts known only from the source code. Each also carries a topic. *Engine* (152) covers disasters, zoning, power, the budget and the internals; *cities* (28) covers which sample cities ship with the game and what some of them are like when loaded; *dynamics* (26) covers what moves the metrics the world report shows: that the four averages are means over byte-valued maps, that the population is a fixed formula over zone densities and can grow only by densification once nobody zones, how often the census runs, that unpowered zones lose people, that rubble stays and traffic decays. The dynamics statements are the knowledge that could help a model forecast the report. Every statement was checked against the fork's source or the bundled cities' own files.

Each model receives two prompts of 103 statements, built so that the two members of a pair never share a prompt: shown "one in eight" beside "one in two", a model would know that exactly one is true. A prompt names the engine and its version, lists its statements in one numbered list and asks for one answer per line with no explanation; the exact preamble and both statement lists are preserved in the private local supplement (Section E). It states the scoring rule, $+1$ for a correct answer, -2 for a wrong one and 0 for Unknown, so that answering pays only above 67% confidence. An answer that does not parse counts as wrong. A model's score on a set of statements is its points divided by their number: 1 for all correct, 0 for all Unknown and -2 for all wrong, on every subset alike. The test was run on September 22, 2026, on the 24 models of the main run at the main run's reasoning settings, one call per prompt, for $0.93 in all.

Tables 37 and 38 report model scores and associations. Engine, dynamics and honeypot scores track ECI; city knowledge does not. Most models score below zero on the city subset, answering rather than abstaining despite frequent errors. Separate intervals do not establish equivalence among difficulty tiers.

The knowledge–forecasting association persists after adjusting for ECI on mid-range questions, but not on tail questions (Table 38). Continuous results depend on the inferential procedure, as detailed

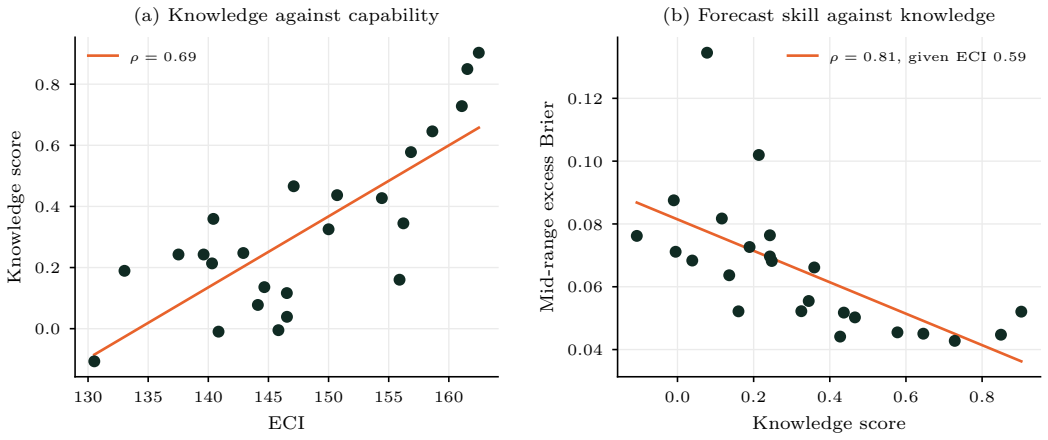


Figure 13: The knowledge test against capability and against forecast skill, one point per model (24). (a) Knowledge score over all 206 statements against ECI, Spearman ρ over the models. (b) The main run's mid-range excess Brier score against the knowledge score; ρ is sign-adjusted so that a positive value means more knowledge, better forecasts, and "given ECI" is the partial rank correlation with ECI removed from both. Lines are least-squares fits.

below. These associations do not identify a causal effect of engine knowledge or establish that tail errors reflect calibration rather than understanding.

Two response patterns help explain the scores. First, 3 models answered Unknown to more than half the statements (GPT-5 Nano, Claude Haiku 4.5, GPT-5.4 Nano), GPT-5 Nano to 98% of them. The rule scores an abstention 0, so their scores near zero measure a refusal to guess as much as a lack of knowledge; GPT-5 Nano behaves the same way in the forecasting runs. Second, the honeypots: on average a model marked 16% of them True, 7 models marked none, and GPT-5 Mini marked 60%. The mechanics of SimCity 2000 and 3000 are part of what the smaller models believe about Micropolis.

The test has three limitations. The statements were drafted by Claude Fable with references into the source code, and that model, which is in the panel, scores highest (0.90, rank 1); without it the correlation with ECI is 0.65. The source checks do not eliminate wording ambiguity or cross-cues, and a test written by one model may favor what that model finds natural to state. The difficulty levels are subjective. And the test is one exchange per prompt with no repeats, so a model's score carries the noise of a single reply. An earlier test was replaced before these results were used; its details remain in the preserved source.

Abstention, cues and inferential sensitivity. The $+1/-2/0$ score combines knowledge, abstention and response strategy. Partial rank associations given ECI are 0.591, 0.120 and 0.488 for mid-range, tail and continuous skill; additionally adjusting for abstention fraction gives 0.597, 0.134 and 0.517. These are exploratory associations, not a causal adjustment or a pure knowledge measure. The continuous partial correlation's model-percentile interval includes zero ($[-0.04, 0.78]$), whereas the reported $p = 0.018$ comes from a t approximation with $n - 3$ degrees of freedom; the two methods do not agree on statistical significance.

Putting each true/false pair in separate prompts does not remove every clue from related statements within a prompt. Ten identified cross-cue items yield partial associations 0.565, 0.139 and 0.473 when deleted from the existing answers. Census wording is also ambiguous: the engine updates the short-term census every four city-time units and the long-term census every 48, while prompt1 item13 and prompt2 item86 do not specify which series. Their historical labels and answers are preserved. Excluding the two ambiguous items gives 0.588, 0.133 and 0.487; excluding them plus the cross-cues (11 unique items) gives 0.564, 0.155 and 0.476. Deleting these items provides limited sensitivity checks, not a complete cue audit or a causal experiment, since models already saw the full prompts.

Table 37: The knowledge test by model, ordered by ECI. Scores follow the prompt's rule, $+1$, -2 and 0 over the statements of the subset, so 1 is every statement right and 0 is every statement Unknown: over all 206 statements, over the engine, cities and dynamics topics, and over the 20 honeypots. "Unknown" is the share of all statements the model declined to answer and "Wrong" counts its wrong and unparsed answers.

Model	ECI	All	Engine	Cities	Dynamics	Honeypot	Unknown (%)	Wrong
Claude Fable	162.5	0.90	0.96	0.54	0.96	1.00	8	1
Claude Opus 5	161.5	0.85	0.86	0.71	0.96	1.00	5	7
GPT-5.6 Sol	161.1	0.73	0.93	-0.43	0.77	1.00	2	17
GPT-5.5	158.6	0.65	0.87	-0.36	0.42	1.00	9	18
Gemini 3.7 Flash	156.8	0.58	0.78	-0.50	0.54	1.00	0	29
GPT-5.6 Luna	156.2	0.34	0.47	-0.29	0.31	0.25	0	45
Claude Sonnet 5	155.9	0.16	0.24	-0.18	0.04	0.25	11	50
DeepSeek V4 Flash 0731	154.4	0.43	0.52	-0.18	0.54	0.40	12	31
Gemini 3 Flash Preview	150.7	0.44	0.56	-0.18	0.38	1.00	13	30
GPT-5	150.0	0.33	0.48	-0.50	0.31	0.55	0	46
o3	147.1	0.47	0.57	0.00	0.35	0.70	29	17
GPT-5.4 Nano	146.5	0.04	0.09	-0.14	-0.04	-0.85	51	31
o4 Mini	146.5	0.12	0.22	-0.14	-0.19	0.25	10	54
GPT-5 Mini	145.8	-0.00	0.07	-0.50	0.08	-0.80	1	68
Gemini 3.1 Flash Lite Preview	144.7	0.14	0.25	-0.43	0.08	0.25	0	59
Qwen3.5-Flash	144.1	0.08	0.12	-0.18	0.08	0.70	0	63
Claude Haiku 4.5	142.9	0.25	0.30	0.00	0.19	0.60	56	13
GPT-5 Nano	140.8	-0.01	-0.01	0.00	0.00	-0.10	98	2
Kimi K2	140.4	0.36	0.47	-0.14	0.23	0.55	17	32
Gemini 2.5 Flash	140.3	0.21	0.31	-0.29	0.19	0.85	0	54
Qwen3 235B A22B	139.6	0.24	0.30	0.00	0.19	0.25	36	27
GPT-4.1	137.5	0.24	0.41	-0.71	0.31	1.00	0	52
DeepSeek V3	133.0	0.19	0.14	0.21	0.42	0.25	11	48
Llama 4 Scout	130.5	-0.11	-0.03	-0.71	0.08	-0.35	0	76

Table 38: Correlations of the knowledge score by statement subset over the 24 models: Spearman ρ with ECI, with a 95% bootstrap interval over the models; with each of the main run's three forecast scores, sign-adjusted so that a positive value means more knowledge, better forecasts; and the same three with the ECI rank partialled out of both sides.

Subset	n	ρ with ECI	ρ with forecast score			ρ with forecast score, given ECI		
			Mid-range	Tail	Continuous	Mid-range	Tail	Continuous
All	24	0.69 [0.34, 0.90]	0.81	0.58	0.77	0.59	0.12	0.49
Engine	24	0.71 [0.38, 0.89]	0.80	0.63	0.79	0.55	0.22	0.52
Cities	24	0.07 [−0.48, 0.54]	0.03	-0.30	-0.01	-0.05	-0.53	-0.11
Dynamics	24	0.53 [0.08, 0.83]	0.72	0.55	0.68	0.58	0.28	0.51
Honeypot	24	0.46 [0.03, 0.76]	0.47	0.41	0.56	0.20	0.11	0.37
Difficulty 0	24	0.56 [0.18, 0.83]	0.61	0.50	0.69	0.31	0.14	0.48
Difficulty 1	24	0.76 [0.45, 0.91]	0.82	0.71	0.84	0.55	0.32	0.59
Difficulty 2	24	0.60 [0.18, 0.86]	0.71	0.45	0.55	0.48	0.01	0.13

Administered-item examples and preservation. Original prompt 1 includes "A separate water-coverage value is tracked per map location" and "Each pass of the simulator is divided into 16 phases." The first concerns a mechanic absent from the engine; the second concerns a fact from the source code; the complete administered lists, preamble and historical ambiguous census wording are preserved unchanged in the local supplement's original tables. Those private records are required for exact reproduction; the examples above do not replace them.

D.4 Answering again with nothing revealed (FreeCiv)

We score natural conditionals by turn-2 excess Brier loss against $p(Y \mid X)$, and define gain relative to the unchanged turn-1 forecast. The no-news control measures forecast changes when the question is asked again without a revealed fact. For 100 of the 400 cells, stratified over the five structural groups of Section 4, we continued the turn-1 exchange with a statement that the game had continued and that no new information was available. These 100 selected cells correspond to 99 distinct question prompts: two cells share one no-news answer. The analysis continues to weight cells, rather than distinct prompts. We then asked the question again. We call the change in the forecast the *drift*. Table 39 lists it by model. The mean absolute drift over the 24 models is 0.027, from below 0.001 (GPT-5.5) to 0.081 (Kimi K2). Six models drift by less than 0.005 on average. GPT-5.5 and Sonnet 5 return exactly their turn-1 forecast on 96% and 94% of cells; GPT-5 Nano, Gemini 3 Flash Preview and Kimi K2 return it on 2%. The drift is mostly downward: its signed mean is -0.010 over models, -0.064 for Kimi K2 and -0.038 for Haiku 4.5. On the same 100 cells the mean absolute move after a revealed fact is 0.187, about 7 times the drift, so the moves in Figure 10 are not produced by asking again alone. Drift falls with capability: the Spearman correlation between ECI and mean absolute drift is -0.75 [-0.89, -0.48], $p < 0.001$, $n = 24$, with a percentile bootstrap over models (10,000 samples). The mean observed accuracy change was small. The excess Brier score of the no-news answer against $p(Y)$ differs from that of the turn-1 answer by less than 0.001 on average over models, and by at most $+0.014$ (Gemini 3 Flash Preview). The control was asked one question per prompt, as the natural-conditional arm was; the rows of DeepSeek V3 and DeepSeek V4 Flash come from their rerun of September 20 (Section A.2).

D.5 Starsim reasoning and aggregation robustness

Historical target only. The following checks used the original matched-seed continuous truths and have not been rescored against fixed-state continuations; they do not establish robustness of the revised results. The same items elicited on September 8 at vendor-default reasoning, before the lowest-effort setting of Section 5 was adopted, give, under the same scoring rule, a pooled interventional $\rho(\mathrm{ECI}, -\mathrm{loss})$ of 0.66 with the same five models at the top and a mean excess of 0.47 rather than 0.65. Resampling one answer per item in place of the median gives correlations of 0.72 [0.64, 0.80] for the September 8 run and 0.73 [0.62, 0.82] for the September 16 run. These intervals resample answers, not models. The runs are separate elicitations, so their contrast does not isolate a causal effect of reasoning effort. The largest observed difference is at 50 percent coverage, where the mid-sized reasoning models (GPT-5 Mini, o4-mini) give more variable forecasts without a thinking budget. At this coverage, ρ moves from 0.70 to 0.46.

E Release and documentation

The intended release covers questions and truths, reports, prompts, raw outputs, code and documentation.

E.1 Artifacts, provenance and remaining dependencies

FreeCiv. Draw v1.8 has six item files: 750 mid-range, 300 tail, 50 mirror and 300 continuous questions, 400 natural-conditional cells and 124 extra turn-1 questions. They specify text, generated resolution criteria, replay truths and normalization constants; eight turn-60 reports and prompt builders define the elicitation (Sections A.2 and B.2). A three-file export of aligned natural-conditional replay data is stored privately with manifests of its SHA-256 checksums: `natcond_replays.npz`, `natcond_replays_cells.csv` and `natcond_replay_tags.csv`. Its boolean X/Y ar-

Table 39: The no-news control on 100 FreeCiv natural-conditional cells. Drift is the turn-2 forecast under the no-news statement minus the turn-1 forecast; the table gives its mean absolute value and its signed mean. "Unchanged" is the share of cells on which the two forecasts are equal, and "> 0.05" the share on which they differ by more than 0.05. "|move|" is the mean absolute move on the same cells when the fact is revealed. Models are ordered by ECI. Llama 4 Scout has 99 cells; every other model has 100. Recomputed from the released per-question scores.

Model	ECI	|drift|	drift	Unchanged	> 0.05	|move|
Fable	162.5	0.003	−0.003	80%	0%	0.164
Opus 5	161.5	0.018	+0.003	39%	3%	0.187
GPT-5.6 Sol	161.1	0.005	−0.002	78%	1%	0.141
GPT-5.5	158.6	0.001	−0.001	96%	0%	0.167
Gemini 3.7 Flash	156.8	0.003	+0.000	77%	0%	0.170
GPT-5.6 Luna	156.2	0.015	−0.005	45%	3%	0.139
Sonnet 5	155.9	0.001	−0.001	94%	0%	0.157
DeepSeek V4 Flash	154.4	0.019	−0.016	67%	9%	0.203
Gemini 3 Flash Preview	150.7	0.034	+0.013	2%	13%	0.214
GPT-5	150.0	0.004	−0.001	77%	0%	0.123
o3	147.1	0.036	−0.002	30%	16%	0.159
GPT-5.4 Nano	146.5	0.026	−0.014	26%	12%	0.202
o4-mini	146.5	0.019	−0.017	55%	4%	0.164
GPT-5 Mini	145.8	0.010	−0.009	80%	6%	0.197
Gemini 3.1 Flash Lite	144.7	0.031	+0.001	43%	13%	0.225
Qwen3.5 Flash	144.1	0.027	+0.004	51%	8%	0.209
Haiku 4.5	142.9	0.053	−0.038	4%	36%	0.193
GPT-5 Nano	140.8	0.046	−0.025	2%	27%	0.155
Kimi K2	140.4	0.081	−0.064	2%	48%	0.215
Gemini 2.5 Flash	140.3	0.040	−0.030	58%	13%	0.252
Qwen3 235B	139.6	0.027	−0.017	55%	8%	0.175
GPT-4.1	137.5	0.015	−0.009	46%	4%	0.212
DeepSeek V3	133.0	0.053	−0.029	12%	19%	0.225
Llama 4 Scout	130.5	0.074	+0.012	18%	42%	0.239

rays, metadata and tag maps reproduce all 400 cells, accepted probabilities/counts and 9,584 paired forecast scores (Section C.3). The original game-save extraction was not independently re-audited.

Some local rows reconstructed from score artifacts are not original raw replies. The 92 GB replay archive is documented as available on request; the derived truth files are 1.1 GB and the compressed corpus 311 MB. The simulator uses CivRealm (Qi et al., 2024) and freeciv-web (The Freeciv Project, 2026b).

Micropolis. The paper repository contains scored binary/continuous forecasts, usage summaries, `versions.txt` and `data/micropolis/continuous_replays.tar.gz`. The verified archive has 30 JSONL files (15 cities, years 20/30), four horizons and 1,000 outcomes per metric: 840,000 values for seven metrics. All 840 means, 3,000 binary Yes-count probabilities and 600 continuous identifiers match exported summaries. It records engine `4fe3a72` and producer `2239fb1 (dirty)`, distinct from analysis pin `d70eba9`; the uncommitted changes to the code that produced the archive have not been supplied. The archive records a range of seeds without identifying the seed for each array element. Binary Yes totals do not preserve the outcome of each aligned replay. Original model responses and predicted quantiles, other variant ages and trunk histories used for scales are absent. The horizon records share snapshots and do not represent independent sets of continuations. The numerical agreement verifies the available archive. It does not establish that the run can be reproduced from source or allow joint resampling of binary outcomes.

Starsim. The headline uses 32 continuous items per model; 24 historical binary items remain appendix-only. Five historical item files support seven scored sets, and their reference-distribution file belongs to older skill scoring. Draw v4 does not contain repaired continuous truths. The corrected continuous run under `data/starsim/fixed_state/` retains original scores, new truth samples, score/uncertainty outputs, coverage records and hashes. The seven archived call files match byte for byte the 5,045 successful scored September 16 responses; they do not cover every failed

attempt or establish the numerical precision used to serve the models. Original forecasts are reused, normalization is unchanged, and binary truths are unrepaired (Section A.3).

Scripts and exact supporting material. Micropolis's `worlds/micropolis/scripts/analyze_paper.py` produces its paper figures and tables. FreeCiv's `score_v2.py`, `paper/make_freeciv_figs.py` and `paper/make_freeciv_tables.py` supply scoring and paper outputs. Starsim's paper entry point is `data/starsim/scripts/integrate_fixed_state.py`; combined-score and rank outputs use `data/scripts/combined_score.py`, `plot_combined_score.py` and `make_rank_table.py`. Reproducing the reported outputs requires the corresponding truth samples, forecasts and normalization scales; missing raw responses and quantiles limit reconstruction.

Exact removed prompt lists, report excerpts, tables and diagnostic detail are preserved in a separate private local supplement preserving source commit `ed953586`, with a complete source archive, original 81-page PDF and SHA-256 manifest. Its original appendix and table files include the administered knowledge-test preamble and both 103-statement lists, unchanged ambiguous items, roster/host/billing tables and historical diagnostics. Full forecasting instructions and question templates remain in Section B. Access to the exact files is necessary to reproduce the knowledge test and some ablations. Limitations of the cost and host records are in Section A.5.

Licenses and notices. The repository's root license does not determine the licenses of its dependencies. Starsim 3.3.4 is MIT; MicropolisCore `4fe3a72` includes GPL and name/trademark terms. The intended data license is CC BY 4.0, and raw outputs remain subject to provider terms. SimCity and Civilization are trademarks of their owners; this work is not affiliated with either.

E.2 Refresh and versioning limits

Selecting questions from existing replays is inexpensive; fresh simulations and model calls cost money. New seeds and draws can reduce answer reuse, but cannot guarantee harder questions, contamination immunity or freedom from saturation: simulator knowledge, released information and other exposure remain relevant. Preserve old draws and version new ones with source and run identifiers; check score comparability across draws. Recorded run costs (Section A.5) are not full-refresh prices.