INFORMATION ORGANIZATION AND EVIDENCE EFFICIENCY IN POLICY INDUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

How does information organization affect the amount of evidence foundation models need to induce policies from chronological state-action observations? We investigate whether organization inspired by interfaces designed to support human judgment improves this evidence efficiency. Using public market records and study-defined hidden policies, we construct chronological state-action observations in two text representations: flat tables (FLAT) and panels inspired by professional trading interfaces (PANELS). Models infer four policy settings from supplied candidates within a common rule structure. At each evidence budget, both representations present the same observations in the same chronological sequence. We introduce Evidence-to-Criterion (E2C), the first tested evidence budget at which an induced policy meets a prespecified action-reproduction criterion shared across representations on separate evaluation states. We summarize the cumulative criterion-attainment rate across evidence budgets using the area under the curve (AUC). On this task, FLAT achieves significantly higher E2C AUC than PANELS for both evaluated foundation models. The findings extend input-representation comparisons to evidence efficiency in policy induction and support evaluating information organization alongside content when designing inputs for foundation models.

1 INTRODUCTION

Professional interfaces group and arrange information to reflect domain knowledge and support users' observation and judgment. For example, professional trading interfaces present changing market information alongside users' own order activity. We examine whether information organization designed to support human work also helps foundation models interpret observations.

Our question is how information organization affects the amount of evidence foundation models need to induce policies from identical chronological state-action observations. In policy induction, a model infers behavioral rules from observations and submits an executable policy. We can evaluate that policy on separate states by comparing its actions with those of the policy that generated the observations. Policies induced under different conditions can meet the same action-reproduction criterion while requiring different numbers of observations. Comparing these observation requirements reveals differences in evidence efficiency beyond criterion attainment alone.

Related work examines input representation and the induction of rules from examples. Meaning-preserving prompt formatting can affect model accuracy (Sclar et al., 2024). Table Meets LLM evaluates structural understanding under different table serializations and input designs (Sui et al., 2024). Deng et al. (2024) compare text-based and image-based table representations, while TABVERSE compares understanding of identical table content across representations (Ahsan et al., 2026). CodeARC evaluates inductive program synthesis from input-output examples (Wei et al., 2025), while TwoLIP studies programmatic policy induction from demonstrations (Bashir et al., 2026). Our study examines how input representation affects evidence efficiency in policy induction, measured by the observations needed to meet an action-reproduction criterion.

We reconstruct market states from public event records and execute policies defined for this study to generate simulated order activity. Public market records allow us to construct reproducible observations, and the study-defined policies determine the target actions on evaluation states. The task requires connecting changes in the market with the status of existing orders and the time elapsed

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.

since their submission. Given a common rule structure and candidate values, models infer four hidden settings from the observations and submit an executable induced policy.

We present the same observations as flat tables (FLAT) and panels inspired by professional trading interfaces (PANELS), both supplied as text. At each evidence budget, the observations are provided in the same chronological sequence; their organization within each observation differs. We introduce Evidence-to-Criterion (E2C), which records the first tested evidence budget at which an induced policy meets a prespecified action-reproduction criterion shared across representations on separate evaluation states. E2C AUC summarizes the cumulative criterion-attainment rate across evidence budgets as an area under the curve.

Our task and E2C evaluation provide a reproducible setting for comparing how input representations affect the evidence needed for policy induction. On this task, FLAT yielded significantly higher E2C AUC than PANELS for both evaluated foundation models. This finding supports evaluating how model inputs are organized alongside what information they contain.

2 Task and experimental design

2.1 Policy induction task

We execute study-defined hidden policies (rules governing order submission, retention, and cancellation) to generate simulated order activity and chronological state-action observations. Models receive these observations, the common rule structure, and the candidate values for each setting, but not the values selected for the hidden policy. They infer the four hidden settings and submit an executable policy in a predefined domain-specific language (DSL) with a fixed rule structure. Policy induction uses the observations as input context without updating model weights. We evaluate the induced policy by comparing its actions with those of the hidden policy on separate evaluation states.

The policy can have at most one active order. With no active order, it submits a buy or sell order according to a predefined market-based rule (Appendix A.1), or takes no action. An active order can fill before its side-specific timeout. If it remains active, the policy cancels it when its age reaches that timeout and otherwise leaves it unchanged. This task requires relating changing market information to persistent order states and elapsed time. We compare how organizing these observations as flat tables or panels inspired by professional trading interfaces affects the number of observations needed for policy induction.

The four hidden settings are the BUY and SELL submission price levels and their respective cancellation timeouts. Submission levels specify price ranks among the bid (buy) or ask (sell) quotes listed in the order book. Submission prices are taken from the selected level in the pre-action order book at submission and remain fixed until cancellation or fill. Table 1 defines each setting and lists its five candidate values.

Table 1: Policy settings and candidate values supplied to the models.

Setting	**Definition**	**Candidate values**
BUY submission level	Bid-price rank; 1 = best bid	1, 2, 3, 4, 5
SELL submission level	Ask-price rank; 1 = best ask	1, 2, 3, 4, 5
BUY cancellation timeout	Order age (200 ms/tick)	5, 10, 15, 20, 25 ticks (1-5 s)
SELL cancellation timeout	Order age (200 ms/tick)	5, 10, 15, 20, 25 ticks (1-5 s)

2.2 Observation construction

We combine public market records with study-defined hidden policies to construct reproducible observations. For evaluation, the actions selected by the hidden policy on each evaluation state serve as ground-truth actions. We used public Hyperliquid Level 4 records for BTC, ETH, and SOL (Albers et al., 2026a;b). We reconstruct the order book by applying individual order changes in event order. The reconstructed order book and market trades provide the market information (Figure 1a).

At each 200 ms step, we update market information and apply a deterministic fill rule to any previously active simulated order (Appendix B.1). We then construct an observation containing market information and simulated order, fill, and position records, before applying the current policy action. This pre-action observation therefore reflects earlier policy actions and subsequent market changes and fills, but not the current action. The hidden policy then acts, producing the post-action state (Figure 1b). Simulated orders do not affect the market states reconstructed from the public records.

Each observation contains the 12 best price levels on each side of the order book, up to 24 recent market trades, up to 9 recent own fills and 9 own-order records, and the current position. The same observation scope applies to all four representations. Recent records can appear in successive observations, so the evidence budget counts snapshots rather than distinct events.

The model receives a chronological sequence of pre-action observations, not post-action states. We provide neither labels specifying the current policy action nor a separate ground-truth action log. Earlier submissions and cancellations can nevertheless appear in subsequent order records. Together with the changing market information, these records provide evidence for inferring the four hidden policy settings. Figure 2 illustrates how a past submission and cancellation appear in the same observation under the two representations.

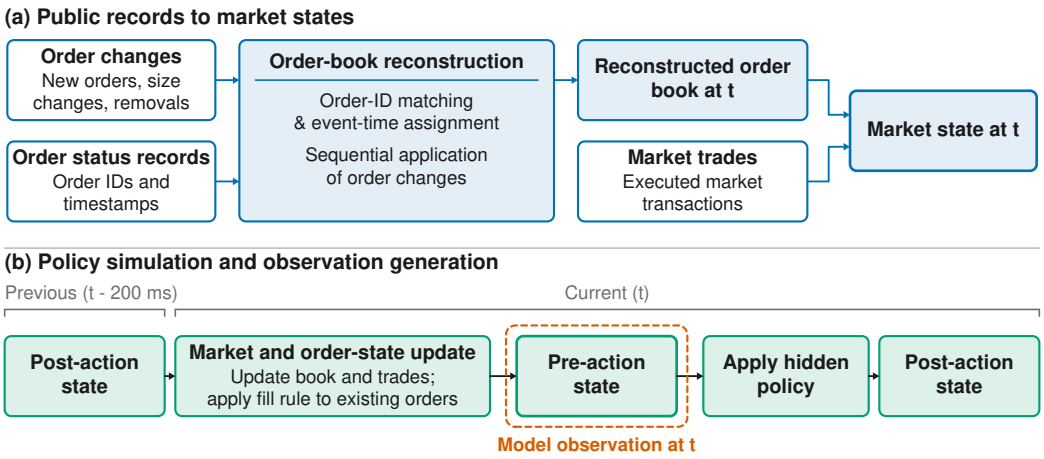

Figure 1: Observation construction: (a) reconstruction of market states from public order-change, order-status, and market-trade records; (b) one policy-simulation step. Each observation is constructed from the pre-action state, before the hidden policy acts.

2.3 Input representations and reading instructions

FLAT is a flat tabular representation in which each observation is a single table in tab-separated values (TSV) format, with a common column structure for market, order, fill, and position records. PANELS is a panel-organized representation: a fixed-width character grid with regions for the order book, market trades, positions, own fills, and order history. Its grouping and layout are inspired by professional trading interfaces that present market information alongside a user's own order activity.

Both representations present the same market and policy-generated information as text, with observations supplied in chronological order. They differ in how information is arranged and identified within each observation. For example, FLAT lists order-book price ranks in a LEVEL column; PANELS conveys those ranks through the positions of price rows. Market trades and own fills appear oldest first in FLAT and newest first in their respective PANELS regions. Each representation is accompanied by a format card explaining how to interpret its fields and conventions (Appendix A.1).

Supplementary comparisons use a message-style representation (RAW), inspired by exchange data feeds and order API records, and a panel image representation (IMG), a 1120×308 pixel PNG rendering of the same logical character grid used in PANELS. Both preserve the observation information provided in FLAT and PANELS. FLAT-RAW compares two text representations; PANELS-IMG compares text with an image of the same grid.

Figure 2 shows display excerpts of one observation from the chronological sequence. At 09:53:05.200, the observation includes a BUY order submitted at 09:53:00.000 and canceled at 09:53:05.000. The same canceled order appears in FLAT's OWN_ORDER row with status CXL and in PANELS' Order History with status CANCELED. An interactive HTML viewer in the supplementary package presents a separate example sequence in both RAW and PANELS.

(a) FLAT

	TYPE	FRAME TIME	EVENT TIME	SIDE	LEVEL	PX	MARKET QTY	OWN QTY	NET POS	STATUS	PLACED AT	CLOSED AT
	FRAME_HEADER	09:53:05.200										
A	OWN_ORDER			BUY		93059.0		1		CXL	09:53:00.000	09:53:05.000
	MKT_TRADE		09:53:00.794	BUY		93086.0	0.00319					
	MKT_TRADE		09:53:00.794	BUY		93086.0	0.00755					
	⋮											
B	MKT_TRADE		09:53:05.052	BUY		93061.0	0.00053					
C	BOOK_LV			BID	01	93060.0	6.69591					
	BOOK_LV			BID	02	93059.0	0.00013					
	⋮											
	BOOK_LV			ASK	01	93061.0	13.01644					
	BOOK_LV			ASK	02	93062.0	4.94830					
	⋮											
D	POS								0			

(b) PANELS

SNAPSHOT_TIME_UTC: 09:53:05.200 | INSTRUMENT: BTC-PERP | QUANTITY UNITS: MARKET=BTC | OWN=POLICY LOT

Depth of Market (DOM)

	OWN BID	BID SIZE	PRICE	ASK SIZE	OWN ASK
			93072.0	0.15811	
			93071.0	0.72260	
			93070.0	0.01074	
			93069.0	1.61121	
			⋮		
			93061.0	13.01644	
C		6.69591	93060.0		
		0.00013	93059.0		
		0.00013	93058.0		
		0.00013	93057.0		
			⋮		

Time & Sales

	TIME	PRICE	SIZE	AGGR SIDE
	09:53:05.052	93061.0	0.00053	BUY
B				
	09:53:03.986	93062.0	0.09197	BUY
	09:53:03.986	93061.0	0.00013	SELL
	09:53:03.986	93060.0	0.00096	SELL
	⋮			
	09:53:02.570	93085.0	0.00067	BUY
	09:53:02.570	93084.0	0.32228	SELL
	09:53:02.570	93083.0	0.00016	SELL
	09:53:02.570	93083.0	0.00011	SELL
	09:53:02.405	93085.0	0.00200	BUY
	⋮			

Positions

INSTRUMENT	NET POSITION
BTC-PERP	0

D

Own Fills

FILL TIME	INSTRUMENT	SIDE	FILL PRICE	FILL QTY
⋮				

Order History

PLACED AT	CLOSED AT	INSTRUMENT	SIDE	LIMIT PRICE	QTY	STATUS
09:53:00.000	09:53:05.000	BTC-PERP	BUY	93059.0	1	CANCELED
⋮						

A

Figure 2: Display excerpts of the same observation in FLAT and PANELS. Labels A-D identify corresponding order, trade, book-level, and position information, respectively. PANELS' side-by-side regions are stacked vertically for display; omissions, labels, and highlights are editorial.

2.4 Evaluation Units and Experimental Controls

An evaluation unit combines one hidden policy with one window set. We selected 12 distinct hidden policies from the 625 possible setting combinations to cover the candidate values approximately evenly. For each of the four settings, each candidate value appears in two or three selected policies.

To include market conditions with different degrees of price variation, we selected three window sets per asset, one from each of the low, medium, and high volatility groups. Across BTC, ETH, and SOL, this yields nine window sets. Each set comprises three market-data intervals: W supplies observations for policy induction, and the other two supply data for constructing the Natural hold-out and Controlled Probe evaluation states. All 27 selected intervals are nonoverlapping in time. Appendix B.1 describes the eligibility and selection rules. Applying all 12 policies to these

nine window sets yields 108 evaluation units, shared by both models and all four representations. Although market windows are shared across policies, simulated orders, positions, and ground-truth actions are generated separately for each policy. Table 2 summarizes the models, windows, and evidence budgets.

An evidence budget is the number of observations supplied in chronological order from the start of W. Larger budgets include the observations at smaller budgets. Each tested budget uses a separate request, without previous answers, reasoning, or scoring feedback. Without early stopping, the main experiment would require 3,456 model requests. Section 3.3 describes the stopping procedure.

Table 2: Overview of the experimental design.

Item	**Design**
Primary analysis model	GPT-5.6 Luna, reasoning medium
Replication model	Claude Sonnet 5, adaptive thinking low
Evaluation units	12 policies $\times$ 9 window sets = 108
Market data windows	BTC/ETH/SOL $\times$ 3 volatility categories; 60 s, sampled every 200 ms
Evidence budgets	40, 78, 153, and 300 state-action observations

Within each model, all four representations use the same evaluation units and prespecified evidence budgets. The primary comparison is between FLAT and PANELS, with supplementary comparisons between FLAT and RAW and between PANELS and IMG. For each evaluation unit and budget, we matched and cross-checked the observation content, number of observations, and chronological sequence across representations before execution. Within each model, the common task instructions, answer DSL, model execution settings, evaluation states, scoring procedure, and recovery criterion were held fixed across representations. Input token counts were not matched, but each request was checked to fit within the model's context limit while reserving space for the maximum output length. Appendix B.2 reports execution settings and input-size summaries.

Before the main experiment, we used separate development data to check task feasibility and difficulty and to adjust candidate settings, task instructions, and the displayed observation scope.

3 Evaluation and evidence efficiency

3.1 Evaluation states and policy scoring

We score each submitted policy on two evaluation sets that are withheld from the model during policy induction. Natural hold-out uses the original pre-action states from its source window. Controlled Probe retains all pre-action states from a separate source window and adds a perturbed state for each original state containing an active order. Each perturbation removes that order while preserving market information and the position. This tests the policy's behavior without an active order under market conditions that originally occurred with one.

We apply the induced policy and hidden policy to the same evaluation state and compare their one-step actions. For each perturbed state, we recompute the ground-truth action by applying the hidden policy to that state. Appendix A.2 provides an example of the original and perturbed states.

We score action reproduction rather than agreement on action type alone. For BUY and SELL, both the action type and absolute order price must match; for CANCEL and NOOP, matching the action type is sufficient. A match contributes one true positive (TP) to the corresponding action class. A mismatch contributes one false negative (FN) to the hidden policy's action class and one false positive (FP) to the induced policy's action class. If both policies return BUY but at different prices, the mismatch contributes both an FN and an FP to BUY. We calculate Macro-F1 as the equally weighted mean of the F1 scores for BUY, SELL, CANCEL, and NOOP. All four ground-truth classes occur in every evaluation set. For each evaluation unit, we calculate separate Macro-F1 scores for Natural hold-out and Controlled Probe. For Controlled Probe, we pool the class-specific TP, FP, and FN counts across the original and perturbed states before calculating F1 and Macro-F1.

3.2 RECOVERY CRITERION

The recovery criterion requires an induced policy to reproduce the hidden policy's actions better than prespecified comparison policies. We constructed these policies by changing rules or settings of each hidden policy. For Natural hold-out, we use 6 comparison policies, including policies that always return NOOP or always cancel an active order. For Controlled Probe, we use 16 comparison policies, each changing exactly one of the four hidden settings to another candidate value while preserving the remaining settings and rules. Before the main experiment, we verified that, for every evaluation unit, each comparison policy produced a different action from the hidden policy on at least one state in its assigned evaluation set. Appendix A.3 provides the policy definitions and summarizes their verification.

Each evaluation unit has two thresholds. The Natural threshold is the highest Macro-F1 obtained by the 6 comparison policies on that unit's Natural hold-out, and the Probe threshold is the highest obtained by the 16 comparison policies on its Controlled Probe. At a given evidence budget, the induced policy meets the recovery criterion when its Natural hold-out Macro-F1 strictly exceeds that unit's Natural threshold and its Controlled Probe Macro-F1 strictly exceeds that unit's Probe threshold. It must therefore outperform the best-scoring comparison policy on each set. The thresholds were fixed before the main experiment without using its model responses or E2C results, and the same thresholds apply to all models and representations within each unit. Criterion attainment concerns action reproduction on the designated evaluation states, not complete policy recovery on every possible state. Table 3 summarizes the distribution of unit-specific thresholds across the 108 evaluation units.

Table 3: Summary of unit-specific recovery thresholds across 108 evaluation units.

Evaluation set	**Minimum**	**Median**	**Maximum**
Natural hold-out	0.73647	0.74631	0.74865
Controlled Probe	0.85525	0.93278	0.98923

3.3 EVIDENCE-TO-CRITERION AND AUC

Evidence-to-Criterion (E2C) records the first tested evidence budget at which a submitted policy meets the recovery criterion. For each evaluation unit and model-representation combination, we test the fixed, nested budgets of 40, 78, 153, and 300 observations in increasing order and stop further calls at the first attainment.

For each model-representation combination, the E2C curve shows the proportion of all 108 evaluation units that have attained the criterion by each budget. A unit first attaining at 78 observations remains counted at 153 and 300 without further requests. Units that do not attain the criterion after evaluation through 300 observations are still included among the 108 units used to calculate this proportion. Their attainment budgets are right-censored beyond 300.

We calculate E2C AUC by connecting cumulative attainment rates with straight lines on the $\log_2$ budget axis and normalizing the area over the range from 40 to 300. The same calculation applies to each unit. Let $x_j = \log_2 B_j$ for the tested budgets B_j. For each unit u with a completed evaluation path, set $R_{u,j} = 1$ if the criterion was attained by B_j, and 0 otherwise:

$$\mathrm{AUC}_u = \sum_{j=1}^{3} \frac{(x_{j+1} - x_j)(R_{u,j} + R_{u,j+1})}{2(x_4 - x_1)}.$$

Units evaluated through 300 without attainment have AUC 0 and remain in the average. The mean unit-level AUC equals the normalized AUC of the cumulative curve. Values range from 0 to 1, with higher values reflecting more units attaining the criterion with fewer observations.

A completed answer violating format or semantic requirements counts as non-attainment at that budget. Incomplete responses or evaluations are not counted as non-attainment. No evaluation paths remained unresolved in the final analysis.

3.4 PAIRED COMPARISONS AND STATISTICAL INFERENCE

For the primary comparison, we calculate the paired AUC differences and their equally weighted mean across the 108 evaluation units separately for each model:

$$D_u = \mathrm{AUC}_{u,\mathrm{PANELS}} - \mathrm{AUC}_{u,\mathrm{FLAT}}, \qquad \Delta = \frac{1}{108}\sum_{u=1}^{108} D_u.$$

Negative values indicate greater evidence efficiency for FLAT, and positive values indicate greater evidence efficiency for PANELS. We apply the same paired-difference calculation to the supplementary comparisons, using the subtraction order shown in Table 4.

To estimate uncertainty in the mean difference, we account for dependence among units sharing a policy or window set. For the primary analysis, we use restricted wild cluster bootstrap-t with weights applied along the policy axis (MacKinnon et al., 2021). We calculate standard errors using a two-term variance estimator (MacKinnon et al., 2021) that sums the CV1-adjusted policy and window-set variance components (Cameron et al., 2011). Confidence intervals (CIs) for the mean difference are obtained by inverting these tests (MacKinnon, 2015).

We selected the two-term estimator before conducting the main experiment, considering CI availability and performance in synthetic checks. In these checks, the alternative three-term estimator produced negative variance estimates in synthetic data and bootstrap replicates, limiting CI availability under the validity requirements used for the analysis. Appendix D.2 describes the primary procedure, the pre-experiment synthetic checks, and their limitations.

Before the main experiment, we set the smallest effect size of interest (SESOI) to $\delta \approx 0.05229$ as the margin for AUC differences. This margin is based on a 10% reduction in observations on the normalized log-budget scale (Appendix D.1); neither the margin nor an observed AUC difference directly represents percentage observation savings. For the primary comparison, a two-sided 95% CI excluding 0 indicates a statistically significant directional difference. If it includes 0, we assess equivalence. In both primary and supplementary equivalence assessments, the policy-axis two-sided 90% CI must lie entirely within the common margin $(-\delta, \delta)$ (Lakens, 2017).

We use additional analyses to examine whether the FLAT-PANELS difference recurs in a second model and across repeated runs, and whether it persists when we change the variance estimator or the bootstrap weighting axis. For cross-model replication, we analyze Claude Sonnet 5 separately on the same evaluation units. Post-hoc sensitivity analyses compare policy-axis and window-axis weights under two-term and three-term variance estimators, using the same paired differences. A valid-only analysis removes the 99% valid-replicate stopping rule for CI inversion while retaining replicate-level validity checks (Appendix D.3). To examine run-to-run variation, we perform three runs of the policy-induction procedure with GPT-5.6 Luna under identical conditions for each of 24 evaluation units selected before inspecting the main-experiment results, comparing first-attainment budgets and mean AUC differences. These comprise the main-experiment run and two additional runs excluded from the primary analysis. Appendix C.1 provides selection and repetition details.

4 RESULTS

4.1 FLAT-PANELS COMPARISON

For both models, FLAT achieved significantly higher E2C AUC than PANELS. The mean PANELS-FLAT AUC difference was -0.06946 for GPT-5.6 Luna (two-sided 95% CI: -0.10972, -0.03201) and -0.08184 for Claude Sonnet 5 (-0.14890, -0.02395) (Figure 3c). Both point estimates exceeded the prespecified SESOI in absolute value, but their CIs also included effect magnitudes smaller than this margin.

In both models, cumulative criterion attainment was higher with FLAT than with PANELS at every tested budget (Figure 3a-b). By 78 observations, 87 versus 75 units had attained the criterion with FLAT and PANELS for GPT-5.6 Luna, and 90 versus 79 for Claude Sonnet 5. By 300 observations, the corresponding counts were 108 versus 103 and 108 versus 100.

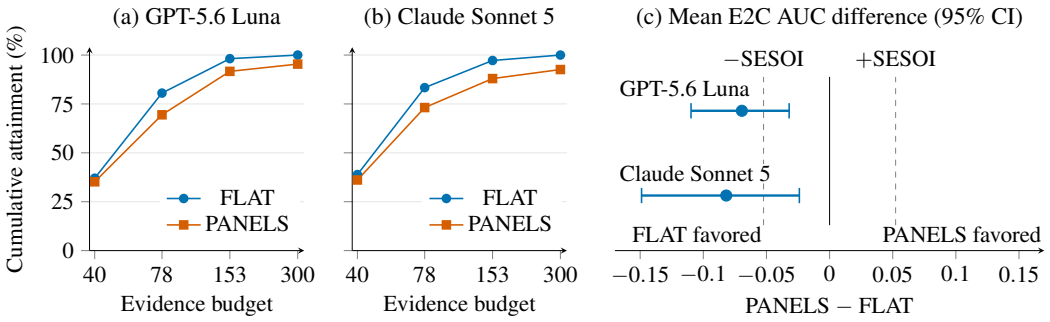

Figure 3: FLAT-PANELS comparison. (a-b) Cumulative attainment by evidence budget ($\log_2$ scale; 108 units per curve). (c) Policy-axis bootstrap 95% CIs; dashed lines: ±SESOI.

In the main experiment, all 419 answers that first attained the criterion with FLAT or PANELS across the two models achieved Macro-F1 = 1 on each of the two evaluation sets, Natural hold-out and Controlled Probe. Each answer therefore fully reproduced the hidden policy's actions on these evaluation states.

4.2 Statistical sensitivity and repeated runs

Statistical sensitivity. The statistically significant FLAT advantage was maintained when bootstrap weights were applied along the window axis instead of the policy axis and when the three-term variance estimator replaced the two-term estimator. These post-hoc analyses used the same 108 paired AUC differences per model. All reported 95% CIs lay below zero; the three-term CIs used valid-only inversion. Appendix C.2 reports the numerical results and CI availability conditions, while Appendix D.3 provides the additional procedures and synthetic checks.

Repeated runs. Across three runs of policy induction with GPT-5.6 Luna on the same 24 preselected evaluation units, mean E2C AUC was higher for FLAT than for PANELS in every run. The mean PANELS-FLAT AUC differences were -0.08333, -0.04869, and -0.01381, respectively, showing consistent direction but variable magnitude across runs. Individual first-attainment budgets also varied across runs; Appendix C.1 reports these changes and the selection and repetition details.

4.3 Supplementary representation comparisons

For both models, PANELS achieved significantly higher E2C AUC than IMG, with two-sided 95% CIs for the mean PANELS-IMG AUC difference entirely above 0 (Table 4). Presenting the same evaluated panel grid as text thus yielded greater evidence efficiency in policy induction than presenting it as an image.

Comparing flat tables (FLAT) with message-style records (RAW) of the same observations supported equivalence within the common SESOI margin for both models on the evaluated task. For each model, the two-sided 90% CI for the mean FLAT-RAW AUC difference lay entirely within the prespecified bounds (Table 4).

Table 4: Supplementary E2C AUC comparisons.

Comparison	Model	Mean difference	95% CI	90% CI
PANELS - IMG	Luna	0.50812	(0.34381, 0.67186)	(0.37384, 0.64486)
PANELS - IMG	Sonnet 5	0.19456	(0.08823, 0.30266)	(0.10666, 0.28216)
FLAT - RAW	Luna	0.01081	(−0.01773, 0.03938)	(−0.01172, 0.03448)
FLAT - RAW	Sonnet 5	−0.01696	(−0.03932, 0.00289)	(−0.03491, −0.00043)

5 DISCUSSION AND LIMITATIONS

Identical chronological state-action observations yielded different evidence efficiency in policy induction when organized differently. On this task, flat tables were more evidence-efficient than text panels inspired by professional trading interfaces for both evaluated models. Prior work has shown that input formatting and representation can affect model performance (Sclar et al., 2024; Ahsan et al., 2026). Our study extends these findings to evidence efficiency in policy induction by examining how input representations affect the number of observations needed to meet a common action-reproduction criterion. Researchers and practitioners adopting human-facing interfaces or existing storage formats as model inputs should therefore evaluate whether their information organization suits the model and task. Input design should address both what information to provide and how to organize it.

E2C offers an operational way to assess foundation models' reasoning by measuring the first tested evidence budget at which an induced policy meets a common performance criterion. It can reveal differences in evidence use across models or input conditions when their induced policies meet the same criterion at different tested budgets. Researchers can construct task-appropriate observations and criteria to compare models, prompts, or input representations while controlling the remaining conditions. The reasoning assessed depends on the rules to be induced and the performance tested by the criterion.

Our finding that input representations affected evidence efficiency in both evaluated models means that evaluations using E2C should report how observations are represented. When comparing models or prompts, researchers should keep the input representation consistent to avoid mixing differences in models or prompts with differences in representation. Results obtained with a common representation should still be interpreted as performance under that representation.

The findings concern the evaluated models, market data, policies, and implemented representations. The comparison evaluates the implemented representations as a whole. Layout, record order within observations, field labels, reading instructions, and input length differ, and the analysis does not isolate their individual contributions. Evidence efficiency is measured in observation counts, not tokens or computation, and the PANELS-IMG comparison does not separate image interpretation from rule induction. We constructed observations by accumulating public market events into market states and combining those states with simulated order activity generated by policies defined for this study. This design was intended to reduce reliance on recalling individual source records, which alone do not reveal the hidden policy's rules. We did not measure pretraining overlap or contamination reduction, or separate memorization from reasoning in the observed performance.

Additional analyses of the same data supported the FLAT-favoring conclusion across the examined bootstrap weighting axes and variance estimators. In synthetic checks, the two-term procedure over-rejected when window dependence dominated with policy-axis weights or policy dependence dominated with window-axis weights. The valid-only three-term procedure also showed over-rejection under dominant policy dependence with window-axis weights. Agreement across inference settings on the observed data does not establish general error-rate control (Appendices D.2 and D.3). The primary comparison used 108 evaluation units per model, while the repeated-run analysis used one model to evaluate FLAT and PANELS three times each on 24 preselected evaluation units.

The findings motivate testing how information organization changes the evidence needed for policy induction in other professional domains, including healthcare and industrial operations, and which ways of organizing observations suit each model and task. A further question is how the organization of pretraining and post-training data affects learning efficiency and performance after training.

REPRODUCIBILITY STATEMENT

The supplementary package README introduces the task through a short GIF and an interactive HTML example showing how the same observations are organized as RAW records and PANELS. Reader aids are distinguished from the original model inputs. The P04/w3 walkthrough connects these observations to saved model requests, induced policies, and evaluation states, guiding readers from a concrete example to the representation comparisons and experimental results. The package provides prompts, model settings, saved answers and results, source code for market reconstruction, observation generation, policy simulation, and rendering all four representations, and scripts for re-scoring saved example policies on supplied evaluation states and recomputing statistical inference from saved paired data. REPRODUCING.md gives environment requirements and execution commands, while DATA_SOURCES.md identifies the external market data. Raw market archives, full intermediate market data, and the live model-calling harness are not bundled. Appendices A-D provide further methodological and statistical details.

AI USE STATEMENT

Generative AI tools assisted with literature retrieval and discovery, developing and reviewing the experimental methodology, implementing experimental methods in code and running experiments, interpreting results, and drafting, translating, and polishing manuscript text. AI-assisted code and analyses were checked through executable tests and cross-reviews. The author reviewed the methodology, code, results, and manuscript and takes responsibility for the final content.

REFERENCES

Momina Ahsan, Sarfraz Ahmad, Ming Shan Hee, Roy Ka-Wei Lee, and Preslav Nakov. TABVERSE: Benchmarking cross-format table understanding in LLMs and VLMs. arXiv preprint arXiv:2606.09578v1, 2026. URL https://arxiv.org/abs/2606.09578v1.

Jakob Albers, Mihai Cucuringu, Sam Howison, and Alexander Y. Shestopaloff. An open book: Level 4 order book data from the Hyperliquid exchange. SSRN working paper. doi: 10.2139/ssrn.6465720, 2026a. URL https://papers.ssrn.com/sol3/papers.cfm?abstract_id=6465720.

Jakob Albers, Mihai Cucuringu, Sam Howison, and Alexander Y. Shestopaloff. An open book: Level 4 order book data from the Hyperliquid exchange. Zenodo. Data set, version 1.0. doi: 10.5281/zenodo.18184441, 2026b. URL https://zenodo.org/records/18184441.

Zahra Bashir, Kelsey Allen, and Tom Silver. What to represent, how to act: Programmatic feature induction for few-shot Bayesian imitation. RL in Big Worlds Workshop at RLC, 2026. URL https://openreview.net/forum?id=zibtyst3sI.

A. Colin Cameron, Jonah B. Gelbach, and Douglas L. Miller. Robust inference with multiway clustering. *Journal of Business & Economic Statistics*, 29(2):238–249, 2011. doi: 10.1198/jbes.2010.07136. URL https://doi.org/10.1198/jbes.2010.07136.

Naihao Deng, Zhenjie Sun, Ruiqi He, Aman Sikka, Yulong Chen, Lin Ma, Yue Zhang, and Rada Mihalcea. Tables as texts or images: Evaluating the table reasoning ability of LLMs and MLLMs. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 407–426. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.23. URL https://aclanthology.org/2024.findings-acl.23/.

Daniël Lakens. Equivalence tests: A practical primer for t tests, correlations, and meta-analyses. *Social Psychological and Personality Science*, 8(4):355–362, 2017. doi: 10.1177/1948550617697177. URL https://doi.org/10.1177/1948550617697177.

James G. MacKinnon. Wild cluster bootstrap confidence intervals. *L'Actualité économique*, 91(1–2):11–33, 2015. doi: 10.7202/1036912ar. URL https://doi.org/10.7202/1036912ar.

James G. MacKinnon, Morten Ørregaard Nielsen, and Matthew D. Webb. Wild bootstrap and asymptotic inference with multiway clustering. *Journal of Business & Economic Statistics*, 39 (2):505–519, 2021. doi: 10.1080/07350015.2019.1677473. URL https://doi.org/10.1080/07350015.2019.1677473.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. Quantifying language models' sensitivity to spurious features in prompt design or: How I learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=RIu5lyNXjT.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. Table meets LLM: Can large language models understand structured table data? A benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pp. 645–654. Association for Computing Machinery, 2024. doi: 10.1145/3616855.3635752. URL https://doi.org/10.1145/3616855.3635752.

Anjiang Wei, Tarun Suresh, Jiannan Cao, Naveen Kannan, Yuheng Wu, Kai Yan, Thiago S. F. X. Teixeira, Ke Wang, and Alex Aiken. CodeARC: Benchmarking reasoning capabilities of LLM agents for inductive program synthesis. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=Q5pVZCrrKr.

A POLICY, INPUT, AND EVALUATION EXAMPLES

A.1 POLICY P04 AND MODEL INPUTS

We use P04, one of the 12 hidden policies, to illustrate the model inputs and the evaluation of submitted policies. Its BUY and SELL submission levels are 2 and 5, and its BUY and SELL cancellation timeouts are 25 ticks (5 s) and 10 ticks (2 s), respectively. These values are given here for the reader; they were hidden from the models. Models received the common rules, the candidate values in Table 1, chronological observations, and instructions for submitting an induced policy.

Under the common rules, order-book imbalance (OIB) is the best-bid quantity minus the best-ask quantity, divided by their sum. With no active order, the policy submits a BUY when OIB is at least 0.10, a SELL when it is at most -0.10, and otherwise returns NOOP. Models received this definition, not precomputed OIB values. At most one order can be active, and each order has a fixed size of one synthetic policy lot. For P04, a BUY uses bid level 2 and a SELL uses ask level 5 in the pre-action order book at submission; that absolute price remains fixed until cancellation or fill. Order age is measured from submission in 200 ms ticks. An order can fill before its timeout; if it remains active, P04 cancels it when its age reaches 25 ticks for a BUY or 10 ticks for a SELL, and otherwise returns NOOP.

At an evidence budget of 40, each model's FLAT and PANELS requests for this P04 example contained one user message with instructions followed by 40 chronological pre-action observations. No current-action labels or separate ground-truth action log were supplied. Earlier submissions and cancellations can appear in later order records, as illustrated by the single observation excerpted in Figure 2. The common reading instructions explained how to infer earlier policy actions from these records and distinguish policy cancellations from market-driven fills. They also instructed models to match a submission price to the order book at submission, rather than its price rank in a later observation.

Representation-specific format cards explained how to locate and interpret fields, empty values, order status, and price ranks. Table 5 presents selected verbatim excerpts; DOM denotes the order-book display labeled "Depth of Market" in Figure 2. The cards explained each representation's conventions without revealing P04's hidden settings.

Table 5: Selected reading instructions from the FLAT and PANELS format cards.

Format	Instruction excerpt
Interpreting rows and columns	
FLAT	Rows are not grouped by kind; decode each row from its TYPE and columns, not from its position in the table.
PANELS	A value belongs to the column whose label it sits under.
Interpreting blank fields	
FLAT	An empty field means not applicable, never zero.
PANELS	A blank cell has no value and is not zero.
Identifying the active order	
FLAT	The single OWN_ORDER row with STATUS OPEN is the working order.
PANELS	A working order may therefore be absent from the DOM; identify it from the OPEN row in Order History.
Interpreting price levels	
FLAT	Background price rank on its side: 01..12, with 01 best. A display label, not the policy's submission level.
PANELS	The best ask (ask level 1) is the lowest-priced ASK SIZE row, immediately above the bid rows.

The shared response requirement was: "Return only one valid policy JSON object. Do not output analysis, calculations, explanations, copied evidence, or code fences."

A.2 Natural hold-out and Controlled Probe

The policy induced from P04's observations is evaluated on states from two source windows that do not overlap each other or the observation window W. Neither evaluation set is provided to the model during policy induction. Natural hold-out uses the original pre-action states from one evaluation window. Controlled Probe retains all original pre-action states from the other and adds an order-removed counterpart for each state containing an active order. Removing the order's side, price, and age while preserving market information and the position tests behavior without an active order under market conditions originally encountered with one.

Table 6 shows a P04 Controlled Probe state for BTC-PERP at 11:50:00.200 UTC on 3 December 2025, from a different window than the observation in Figure 2. OIB is calculated from the stored order book and rounded for the reader; it was not supplied as a computed input value.

Table 6: An evaluation state before and after active-order removal.

Item	**Original evaluation state**	**After active-order removal**
Order-book imbalance	0.97637	Unchanged
Second-best bid price	92,932.0	Unchanged
Active order	BUY at 92,932.0	None
Active-order age	200 ms	Not applicable
Position	0 policy lots	Unchanged
Hidden-policy action	NOOP	Submit BUY at 92,932.0

In the original state, the active BUY order's age is 200 ms, below P04's 5 s timeout, so the policy returns NOOP. After order removal, the unchanged OIB satisfies the BUY condition, and P04 submits a new order at its configured level 2 price. Order removal need not change the action in every state.

For each original and perturbed state, we apply the hidden policy to determine the ground-truth action and compare it with the induced policy's one-step action on that same state. Controlled Probe is scored jointly over the original and perturbed states, while Natural hold-out is scored separately, following Section 3.1. This comparison does not simulate a new trading trajectory.

A.3 Comparison policies and recovery thresholds

To judge whether the submitted policy reproduces actions better than specified incorrect alternatives, we constructed comparison policies by modifying each hidden policy's rules or settings. For Natural hold-out, the six alternatives always return NOOP; replace SELL submissions with NOOP; replace BUY submissions with NOOP; cancel whenever an order is active; return NOOP whenever an order is active; or swap BUY and SELL entry conditions. All other rules and settings remain unchanged.

For Controlled Probe, each alternative changes exactly one hidden setting to one of its other four candidate values, preserving the remaining settings and rules. For example, replacing P04's BUY submission level 2 with each other candidate produces four alternatives. Applying this construction to all four settings gives 16 comparison policies per hidden policy. These policies are evaluated on the original and perturbed states together.

This construction applies to all 12 hidden policies. Before the main experiment, we verified that every assigned comparison policy differed from the hidden policy on at least one state in its designated evaluation set across all 108 evaluation units, covering 2,376 policy-unit pairs.

For each evaluation unit, the highest comparison-policy Macro-F1 on each evaluation set defines that set's threshold. The induced policy meets the recovery criterion only when its Natural hold-out and Controlled Probe scores each strictly exceed their respective thresholds, as specified in Section 3.2. Table 3 summarizes the unit-specific threshold distributions.

B Observation construction and model execution

B.1 Market reconstruction, interval selection, and policy simulation

The observation construction in Section 2.2 combines market states reconstructed from public events with simulated policy activity (Figure 1). Timestamps are assigned to order changes using order-status records and record-type-specific rules before the changes are replayed to reconstruct the order book.

To provide opportunities for policy activity in both directions, we screened candidate one-minute intervals aligned to UTC minute boundaries for sufficiently positive and negative order-book imbalance (OIB). Using the reconstructed background book at 200 ms intervals, each candidate had to reach OIB values of at least $+0.40$ and at most -0.40 using the best bid and ask quantities, and at least $+0.20$ and at most -0.20 using quantities summed over the top 20 levels on each side. Each threshold condition had to be met at least once within the candidate interval. The same OIB conditions applied to observation, Natural hold-out, and Controlled Probe source intervals. To avoid reusing data from preliminary model tests, we excluded BTC candidates whose one-minute intervals or preceding 10-minute periods overlapped the intervals used in those tests.

We grouped eligible intervals by price variation during the 10 minutes immediately before each interval began. Volatility was calculated as the square root of the sum of squared one-minute log returns of the midpoint between the best bid and ask prices over that preceding period. Within each asset, eligible candidates were sorted by volatility and divided into low, medium, and high volatility groups at the available boundaries closest to one-third and two-thirds of the candidate count. Intervals with the same volatility score were kept in the same group. These categories therefore describe relative volatility within the eligible candidate pool.

Within each asset-volatility group, we selected the lower median of the remaining candidates ordered by volatility, first for W, then for Natural hold-out, and finally for the Controlled Probe source. We used a fixed ordering of groups and a deterministic rule to resolve score ties. Volatility-group assignments remained fixed during selection. After selecting an interval, we excluded all remaining candidates whose start times were at most 10 minutes from its start time, across all assets and groups. This kept the information periods comprising each selected minute and its preceding 10-minute volatility-estimation period from overlapping. The resulting 27 intervals form the nine window sets described in Section 2.4.

At each 200 ms step, the deterministic fill rule evaluates any previously active simulated order using all market trades since the preceding step and the updated order book. A BUY order fills fully if a sell-aggressor trade occurs strictly below its limit price or the endpoint best ask is at or below that price. A SELL order fills fully if a buy-aggressor trade occurs strictly above its limit price or the endpoint best bid is at or above that price. After market updates and fill processing, we construct the pre-action observation, and the hidden policy then acts. A newly submitted order is first evaluated for fills at the next step, using trades after submission and that next step's order book.

Simulated orders do not alter the reconstructed market, and submissions and cancellations are applied without separate execution delays. The reconstruction and fill rules make the evaluation data reproducible from the same records and processing rules, without exactly reproducing the exchange's full historical order book or actual execution. The evaluation concerns policy induction and action reproduction, not long-term trading performance or profitability.

B.2 Model execution and input sizes

Main-experiment requests used GPT-5.6 Luna (gpt-5.6-luna, reasoning.effort=medium, store=false) and Claude Sonnet 5 (claude-sonnet-5, thinking.type=adaptive, output_config.effort=low), each with a 20,000-token output limit. Table 7 abbreviates these model names as Luna and Sonnet 5.

For each evaluation unit and evidence budget, all representations contained the same observations in the same chronological sequence; file sizes and token counts were not matched. Before execution, requests were checked against each model's context limit while reserving space for the maximum output length.

Table 7 summarizes maximum input sizes across the main-experiment requests that were actually executed. Because requests stopped at first criterion attainment, the budgets actually tested for a given evaluation unit could differ across models and representations. Input bytes include all text (UTF-8) and images (PNG), excluding transport encoding; input token counts are reported by the APIs. Byte and token maxima are calculated independently and may come from different requests, so the table does not compare input sizes for matched requests.

Table 7: Maximum input sizes across main-experiment requests.

Representation	**Model**	**Maximum input content (bytes)**	**Maximum input tokens**
FLAT	Luna	905,892	427,019
FLAT	Sonnet 5	926,150	566,079
PANELS	Luna	1,837,738	575,605
PANELS	Sonnet 5	1,837,738	637,312
RAW	Luna	539,721	346,149
RAW	Sonnet 5	563,299	373,897
IMG	Luna	1,929,093	134,866
IMG	Sonnet 5	1,929,093	149,756

One Claude Sonnet 5 IMG request reached the output limit without producing a scorable policy answer. It was retried once with the same input, settings, and output limit, and the resulting answer was scored. No automatic retry rule was used.

C Supplementary results

C.1 Repeated-run details

We evaluated FLAT and PANELS with GPT-5.6 Luna on the same 24 evaluation units selected before inspecting the main-experiment results. Selection used a fixed seed and constrained allocation: two units per policy from different assets, with eight units per asset and eight per volatility category.

For each selected unit and representation, we performed three runs of the policy-induction procedure under identical input and execution conditions. Each run started at an evidence budget of 40 observations and followed the same budget grid and recovery criterion, stopping at first attainment as described in Section 3.3. The first run was the main-experiment run; the two additional runs were not pooled into the primary analysis.

The mean PANELS-FLAT AUC differences across these 24 units were -0.08333, -0.04869, and -0.01381 in runs 1, 2, and 3, respectively. The direction favored FLAT in every run, while the magnitude varied. Individual first-attainment budgets were identical across all three runs for 13 of the 24 units with FLAT and 8 of the same 24 units with PANELS. These results describe repeated policy induction on the selected units, rather than additional independent evaluation units.

C.2 Statistical sensitivity results

Table 8 compares the prespecified two-term policy-axis analysis with post-hoc sensitivity analyses using window-axis weights, a three-term variance estimator, or both. All analyses use the same 108 paired PANELS-FLAT AUC differences per model, so the estimated mean differences are unchanged. The statistically significant FLAT advantage was maintained across these settings. All four three-term tests of zero mean difference rejected the null, with all 9,999 bootstrap replicates valid at that null.

CI inversion initially required at least 99% valid bootstrap replicates at each tested null. Under this requirement, three-term CIs were available for Claude Sonnet 5 but not GPT-5.6 Luna. A subsequent valid-only analysis removed only this stopping requirement, retaining the replicate-level validity checks and p-value calculation based on valid replicates. Across the null values evaluated during

valid-only CI inversion on the observed data, 97.3-100% of bootstrap replicates were valid. Luna's CIs then became available, while Sonnet 5's CIs and all four p-values at zero were unchanged.

Table 8 reports the two-term CIs calculated under the 99% requirement and the valid-only three-term CIs; all lie below zero. All reported intervals have open endpoints and describe confidence sets obtained within the searched region.

Table 8: Statistical sensitivity of inference for the mean E2C AUC difference (PANELS - FLAT). CIs and p-values are two-sided. Three-term CIs use the valid-only procedure.

Model	**Estimator**	**Axis**	**Nominal 95% CI**	**p-value**
GPT-5.6 Luna	Two-term	Policy	$(-0.10972, -0.03201)$	0.0006
GPT-5.6 Luna	Two-term	Window	$(-0.11184, -0.02308)$	0.0010
GPT-5.6 Luna	Three-term	Policy	$(-0.11183, -0.02997)$	0.0018
GPT-5.6 Luna	Three-term	Window	$(-0.11187, -0.02224)$	0.0020
Claude Sonnet 5	Two-term	Policy	$(-0.14890, -0.02395)$	0.0032
Claude Sonnet 5	Two-term	Window	$(-0.14863, -0.01795)$	0.0174
Claude Sonnet 5	Three-term	Policy	$(-0.15573, -0.01988)$	0.0054
Claude Sonnet 5	Three-term	Window	$(-0.15177, -0.01865)$	0.0176

These results provide sensitivity evidence from the same data, not independent replication. CI availability does not establish general error-rate control. Appendix D.2 describes the primary procedure and the pre-experiment synthetic checks; Appendix D.3 provides the three-term and valid-only procedures, diagnostics, and subsequent synthetic checks.

D Statistical details and synthetic validation

D.1 SESOI definition

We defined the smallest effect size of interest (SESOI) before the main experiment using a 10% reduction in the number of observations as a reference. For an evidence budget of B observations, this reduction corresponds to $0.9B$. Normalizing the log-budget difference by the evaluated range of 40 to 300 observations gives

$$\delta = \frac{\ln B - \ln(0.9B)}{\ln(300/40)} = \frac{\ln(10/9)}{\ln(300/40)} \approx 0.05229.$$

We use δ as the margin for mean E2C AUC differences in Section 3.4. E2C AUC summarizes cumulative criterion attainment across evidence budgets, so an observed AUC difference does not directly translate into a percentage reduction in observations.

D.2 Statistical inference and synthetic validation

For each model, each evaluation unit corresponds to a policy p and window set w. Let D_{pw} denote its PANELS-FLAT AUC difference, equivalent to D_u in Section 3.4, with mean Δ, residuals $e_{pw} = D_{pw} - \Delta$, and $N = 108$. The two-term variance (MacKinnon et al., 2021) uses component-specific CV1 factors (Cameron et al., 2011):

$$\widehat{V}_2 = \frac{1}{N^2}\left[\frac{12}{11}\sum_{p=1}^{12}\left(\sum_{w=1}^{9} e_{pw}\right)^2 + \frac{9}{8}\sum_{w=1}^{9}\left(\sum_{p=1}^{12} e_{pw}\right)^2\right], \qquad \mathrm{SE}(\Delta) = \sqrt{\widehat{V}_2}.$$

For the primary analysis, restricted wild cluster bootstrap-t (MacKinnon et al., 2021) imposed each null using 9,999 replicates with equiprobable Webb weights $\left\{\pm\sqrt{3/2}, \pm 1, \pm\sqrt{1/2}\right\}$, shared within policy clusters. The variance was recalculated for each bootstrap replicate. CIs were obtained by

test inversion (MacKinnon, 2015). Tests used two-sided equal-tailed p-values calculated from valid replicates, with ties included and a +1 correction. CIs for supplementary comparisons are contrast-specific and unadjusted for multiple comparisons.

Valid replicates required finite variance components and test statistics, with variance above the fixed numerical tolerance. Invalid replicates were excluded without replacement. Our prespecified, project-specific rule required at least 99% valid replicates at each tested null. If this requirement was not met during CI inversion, the search stopped and the CI was reported as unavailable. For the three-term estimator, negative bootstrap variance estimates could therefore limit CI availability through this validity requirement.

Before observing the main-experiment results, we evaluated computational availability and test performance on synthetic data. Each condition used 2,000 Gaussian random-effects datasets with 12 policies, 9 window sets, and a true mean difference of zero. Each test used 999 bootstrap replicates. The policy, window-set, and residual components contributed equally to the variance of the mean in the balanced condition. In each dominant condition, the named component contributed 70%, and the other two contributed 15% each.
Table 9 reports rejection rates for two-sided tests at the nominal 5% level using the two-term estimator under each weighting axis. All 2,000 tests per condition were available under both procedures, so each rejection rate uses all 2,000 datasets.

Table 9: Two-term bootstrap null rejection rates at nominal 5%.

Synthetic condition	**Policy-axis weights**	**Window-axis weights**
Balanced	5.60%	5.45%
Policy-dominant	5.15%	8.00%
Window-dominant	8.45%	4.70%
Residual-dominant	5.65%	5.55%

The policy-axis three-term procedure yielded 95% CIs for 61.95-99.80% of synthetic datasets across conditions, compared with 100% for the policy-axis two-term procedure. These availability rates reflect our requirement that at least 99% of bootstrap replicates be valid at each tested null during CI inversion. We selected the two-term estimator for the primary analysis considering both CI availability and synthetic test performance. This choice does not establish superior statistical accuracy or error-rate control across dependence structures. Appendix D.3 reports additional post-hoc sensitivity analyses on the observed data and subsequent synthetic checks.

D.3 Additional statistical sensitivity analyses

We assessed sensitivity to the bootstrap weighting axis and variance estimator in post-hoc analyses of the same 108 PANELS-FLAT paired AUC differences per model. Table 8 compares policy-axis and window-axis weights with two-term and three-term variance estimators. Using the notation in Appendix D.2, the three-term estimator (MacKinnon et al., 2021) subtracts the CV1-adjusted policy-by-window intersection component. Each intersection contains one evaluation unit, giving:

$$\widehat{V}_3 = \widehat{V}_2 - \frac{1}{N^2}\frac{N}{N-1}\sum_{p=1}^{12}\sum_{w=1}^{9} e_{pw}^2.$$

We applied this estimator to both the observed statistic and every bootstrap replicate.

Appendix C.2 reports the observed-data results and CI availability. For GPT-5.6 Luna, CI inversion under the 99% requirement stopped at the estimated mean difference, the search's starting point: 97.35% of policy-axis replicates and 98.25% of window-axis replicates were valid. The valid-only analysis removed this stopping requirement while retaining the replicate-level validity checks and p-value calculation. All excluded replicates during valid-only CI inversion on the observed data had negative bootstrap variances. The data, weights, and remaining search settings were unchanged.

We evaluated valid-only three-term tests using the synthetic setup described in Appendix D.2, with two-sided tests at the nominal 5% level. All datasets were testable except the same 13 residual-dominant datasets on both axes, whose observed three-term variances were negative. Table 10 re-

ports rejections among testable datasets, excluding untestable cases from the denominators rather than counting them as non-rejections. Wilson 95% confidence intervals quantify Monte Carlo uncertainty in these rejection rates.

Table 10: Synthetic rejection rates for valid-only three-term tests at nominal 5%.

Synthetic condition	**Axis**	**Rejections / testable**	**Rejection rate**	**Wilson 95% CI**
Balanced	Policy	90/2000	4.50%	(3.68%, 5.50%)
	Window	78/2000	3.90%	(3.14%, 4.84%)
Policy-dominant	Policy	96/2000	4.80%	(3.95%, 5.83%)
	Window	123/2000	6.15%	(5.18%, 7.29%)
Window-dominant	Policy	110/2000	5.50%	(4.58%, 6.59%)
	Window	84/2000	4.20%	(3.41%, 5.17%)
Residual-dominant	Policy	101/1987	5.08%	(4.20%, 6.14%)
	Window	100/1987	5.03%	(4.16%, 6.08%)

The observed-data analyses support the FLAT-favoring conclusion under the examined inference settings, without providing independent replication. Synthetic performance depended on the dependence structure: the window-axis rejection rate under dominant policy dependence was 6.15%, with its interval entirely above 5%, whereas the policy-axis rate under dominant window dependence was 5.50%, with its interval including 5%. These checks evaluated rejection rates at the true mean under the specified synthetic conditions, not the full CI construction procedure, and do not establish general error-rate control.