Researcher-Induced Estimation Uncertainty at Scale Using Agentic AI

Brett A. McCully

Collegio Carlo Alberto

April 10, 2026

Preliminary and Incomplete

Abstract

Reported standard errors capture sampling uncertainty conditional on one set of researcher decisions, but defensible alternatives in sample construction and specification can shift estimates substantially. I use repeated, independent AI-agent runs, implemented with large language models (LLMs), as stochastic replication agents that receive the same prompt and dataset but generate different research choices. Applying the method to the immigration policy–employment question studied by Huntington-Klein *et al.* (2025), AI agents generate 139 specifications with an interquartile range of $[-0.042, 0.046]$. The results support AI agents as scalable proxies for multi-analyst designs that characterize researcher-induced estimation uncertainty.

Keywords: Researcher degrees of freedom, replication, specification uncertainty, AI agents

JEL Codes: C12, C15, C18

1 Introduction

Empirical papers routinely report standard errors that quantify sampling uncertainty. These standard errors are conditional on a set of researcher decisions: which dataset to analyze, how to define the estimation sample, how to construct outcomes and treatments, and which specification and standard-error adjustments to use. A growing body of evidence shows that researchers making different decisions can produce meaningfully different estimates from the same underlying data (Silberzahn *et al.*, 2018; Huntington-Klein *et al.*, 2021; Breznau *et al.*, 2022). This implies an additional source of uncertainty beyond standard errors capturing sampling variation: researcher degrees of freedom.

Measuring parameter uncertainty due to researcher degrees of freedom at scale is challenging. Multi-analyst studies are expensive, requiring extensive recruitment and coordination. Pre-analysis plans constrain discretion but do not measure the sensitivity of results to alternative, still-defensible research choices (Olken, 2015). Specification-curve and multiverse analyses require the original analyst to enumerate and implement various alternatives (Simonsohn *et al.*, 2020; Steegen *et al.*, 2016).

I overcome these limitations using a novel approach: AI agents can perform empirical analyses at scale. I apply this approach to the DACA–employment question studied by many analysts in Huntington-Klein *et al.* (2025): how does eligibility for the Deferred Action for Childhood Arrivals program affect the probability of full-time employment? Across 139 agent-generated specifications, I obtain an interquartile range from −0.042 to 0.046. Sample sizes vary substantially (IQR 53,232 to 125,521). This combination—near-zero average point estimate, a wide estimate distribution, and sizable sample-size heterogeneity—broadly resembles the patterns from the benchmark multi-analyst study, in which human researchers produce an IQR from 0.014 to 0.051 under comparable high-discretion conditions.

Two distinctions are essential for interpretation. First, the distribution of executed estimates is not a sampling distribution: because all runs use the same realized dataset, dispersion reflects heterogeneity in research choices, not classical sampling error. Second, different research choices can create *estimand mismatch*—two defensible specifications may implicitly target different causal parameters by changing the treated population, timing window, or treatment definition (Lundberg *et al.*, 2021).

I make two main contributions. First, I provide an open-source, scalable approach to multi-analyst designs that lets researchers quantify researcher-induced uncertainty. Second, I show that repeated AI-agent runs provide a computationally tractable way to approximate the distribution of human research choices at scale: each run generates an interpretable, machine-readable specification that can be executed without human intervention.

Related literature. This paper builds on the researcher degrees of freedom literature, which documents how defensible analytic choices affect results (Simmons *et al.*, 2011; Gelman and Loken, 2013; Steegen *et al.*, 2016; Simonsohn *et al.*, 2020). Relative to analyst-enumerated multiverses, I introduce AI-agent-enumerated multiverses across the entire research pipeline, allowing quantification of analyst-induced estimate uncertainty at scale.

Repeated AI-agent runs play a role analogous to bootstrap resampling (Efron, 1979; Andrews and Buchinsky, 2000), approximating a distribution through repeated draws—here over research choices rather than data. Because the AI agents studied here are implemented with LLMs, the paper also connects to emerging work on LLMs as economic agents and research tools (Horton *et al.*, 2023; Brand *et al.*, 2023; Korinek, 2023; Novy-Marx and Velikov, 2026).

Section 2 formalizes the research-choice space and the AI-agent-induced distribution. Section 3 documents the AI-agent workflow. Section 4 reports the distribution of executed estimates, compares it to the human-analyst benchmark, and relates heterogeneity both to specification features and to model choice. The conclusion discusses the main limitations and implications.

2 Framework: researcher-induced uncertainty

I formalize the objects of interest in two steps. First, I define the research problem, the target estimand, and the set of defensible research choices. I then treat repeated AI-agent runs as a stochastic mechanism that samples from that set.

2.1 Scientific target and defensible research choices

A research question asks how treatment T affects outcome Y in a given population U. Define a research problem P as a tuple of defensible choice sets,

$$P = (\mathcal{U}, \mathcal{Y}, \mathcal{T}, \mathcal{S}),$$

where $\mathcal{U}$ is the set of defensible sample definitions, $\mathcal{Y}$ is the set of defensible outcome constructions, $\mathcal{T}$ is the set of defensible treatment constructions, and $\mathcal{S}$ is the set of defensible specifications. An element of $\mathcal{S}$ collects the control set, functional form, estimator, weighting, fixed effects, and variance-estimation choices such as clustering.

For example, in the empirical application below I consider the effect of the immigration policy, Deferred Action for Childhood Arrivals (DACA), on the likelihood of full-time work among affected immigrants. Researchers may make different defensible choices about age

bounds, full-time work definitions, eligibility proxies, controls, and standard-error adjustments. Different combinations of these choices imply different research designs and, in some cases, different causal parameters.

Not every logically possible combination is defensible. A sample definition that includes only U.S. citizens would be inconsistent with the research question which asks about impacts on affected *immigrants*, whereas a sample definition that restricts attention to plausible DACA-eligible immigrants is likely defensible. Similarly, a specification that controls for post-treatment variables is generally less defensible than one that conditions only on pretreatment covariates. Let $\mathcal{R}(P)$ denote the set of defensible research choices: combinations that are internally coherent, feasible in the data, and defensible ex ante given the stated question and standard econometric practice.

Let $\mathcal{P}(P)$ denote the underlying population distribution of observables, rich enough to construct any defensible sample restriction in $\mathcal{U}$ and any defensible outcome or treatment variable in $\mathcal{Y}$ and $\mathcal{T}$.

Definition 1. *A defensible set of research choices is an element*

$$r = (u_r, y_r, t_r, s_r) \in \mathcal{R}(P) \subseteq \mathcal{U} \times \mathcal{Y} \times \mathcal{T} \times \mathcal{S}.$$

Given a realized dataset ω, implementing research choices r produces an estimate $\hat{\theta}(r, \omega)$.

Definition 2. *When research problem P is sufficiently specific to determine a single scientific object of interest, its target estimand is a scalar $\theta^\star(P)$.*

If the research problem is itself too ambiguous to pin down a unique benchmark object, one can instead treat P as defining a set of defensible target estimands; throughout, I write $\theta^\star(P)$ for the benchmark scientific target the research problem is intended to describe.

Definition 3. *Each defensible set of research choices $r \in \mathcal{R}(P)$ induces an implied estimand $\theta(r; P)$, corresponding to the causal or statistical object actually targeted by that choice of population, outcome, treatment, and specification.*

Definition 4. *The* estimand mismatch *of research choices r relative to the target estimand is*

$$\delta(r; P) \equiv \theta(r; P) - \theta^\star(P).$$

When $\delta(r; P) \neq 0$, dispersion across executed research choices mixes differences in estimation strategies with differences in targeted causal parameters.

The broader causal-inference and statistics literatures distinguish the target estimand from the implied estimand and discuss mismatch between the intended question and the performed

analysis (Blair *et al.*, 2019; Choi *et al.*, 2023; Chang *et al.*, 2024; Barnard *et al.*, 2026). In this setting, mismatch arises when one specification defines full-time work as at least 35 hours per week while another uses 30 hours, annual weeks worked; similarly, alternative DACA eligibility proxies using different age and arrival cutoffs target different treated populations. Such choices differ not only in estimation strategy but in the object being estimated.

2.2 AI agents as empirical researchers

Fix a research problem P and a temperature parameter $\tau > 0$, the standard sampling parameter governing how concentrated the model's next-token probabilities are. Higher τ increases the probability of lower-ranked continuations and therefore generates more variation across runs. In this setting, $\tau > 0$ allows one fixed prompt to produce multiple non-identical sets of research choices.

Repeated independent runs of the same AI agent, holding fixed its underlying model and prompt, induce a distribution over defensible research choices, $\pi_{\text{AI}}(\cdot \mid P, \tau)$, supported on $\mathcal{R}(P)$. If $R \sim \pi_{\text{AI}}(\cdot \mid P, \tau)$ and the dataset is fixed at ω, the empirical object in this paper is the fixed-sample distribution of estimates $\hat{\theta}(R, \omega)$.

Why might AI agents produce plausible research choices? Their underlying models are trained on academic articles, codebases, and empirical templates (Gao *et al.*, 2020; Chen *et al.*, 2021)—the same materials from which researchers learn to define samples, write regression equations, and choose standard-error adjustments. AI agent outputs can therefore be interpreted as draws from a compressed, noisy representation of standard empirical social science practice. I seek to test whether $\pi_{\text{AI}}(\cdot \mid P, \tau)$ coincides with the distribution human researchers would generate.

3 Computing Pipeline

This section describes how I generate repeated draws $\hat{\theta}(R, \omega)$ from $\pi_{\text{AI}}(\cdot \mid P, \tau)$. Each run receives the same natural-language research prompt and the same dataset; only the AI agent's choices vary across runs. The pipeline records the full sequence from prompt to final numeric output, so each estimate can be traced back to the underlying specification and compared with the distribution generated by human analysts.

Phase 1: research choices. The AI agent receives a fixed description of the research question together with a data codebook. It returns a structured specification encoding four researcher choices: sample restrictions, outcome definition, treatment definition, and an

estimator command that embeds controls, fixed effects, weighting, and variance-estimation decisions. A machine-readable contract validates each specification before any code runs. Because the underlying large language model is called with positive randomness (temperature $\tau > 0$), each call can yield a different set of choices.

Phase 2: execution. The AI agent continues to write the analysis script inside an isolated workspace containing only the raw data and documentation files. Combining specification generation and execution in one session makes the workflow more reliable in practice: the model can immediately revise non-implementable choices after encountering null samples, missing variables, or runtime errors. The pipeline checks that each completed run produces a point estimate, standard error, and sample size.

Phase 3: aggregation. The final phase aggregates the full archived run directory structure, records whether each session preserves a recoverable specification, and classifies each usable specification's methodological features. Classification is deterministically accomplished via a Python program, so no additional stochasticity is introduced in this phase. The pipeline parses the estimator command from the structured specification using regular expressions: model type (OLS, WLS, Logit, etc.) is inferred from the estimator function name; control variables and fixed effects are read from the regression formula's right-hand side, distinguishing continuous regressors from categorical terms; sample weighting is detected from the estimator's weight argument; and the standard-error adjustment (clustered, heteroskedasticity-robust, or none) is identified from variance-estimation flags.

Accessing AI models. The implementation used here relies on GitHub Copilot, a subscription service which grants access to a range of frontier AI models. These include GPT models from OpenAI, Claude models from Anthropic, and Gemini models from Google. The empirical application uses only GPT models, but the pipeline can in principle use any of these model families. I use the GitHub Copilot CLI, a command-line interface that facilitates programmatic calls to Copilot models and scales research-choice generation and execution.[1]

[1] An alternative is to use API calls to the same models. Direct API calls to model providers, however, are much more expensive than personal or institutional subscriptions such as GitHub Education, Claude Code, or Codex. I also experimented with generating Phase 1 choices from the Mistral API using Devstral, with the advantage of explicit control over temperature and random seed. In practice, Devstral was not sufficiently reliable at tool use for Phase 2, so many of its Phase 1 specifications were not executable.

4 Empirical Application

The empirical object is the distribution of $\hat{\theta}(R, \omega)$ induced by $\pi_{\text{AI}}(\cdot \mid P, \tau)$ for a fixed raw dataset ω. This dispersion reflects variation in research choices, rather than sampling uncertainty. I apply the approach to the question answered by many analysts in Huntington-Klein *et al.* (2025): how the immigration policy, Deferred Action for Childhood Arrivals, DACA, affected full-time work among affected immigrants.

4.1 The benchmark multi-analyst study

Huntington-Klein *et al.* (2025) recruit a large team of economists and graduate students and ask each to independently answer the same causal question using the same instructions and raw data: how did eligibility for DACA affect the probability of full-time employment among eligible immigrants? Participants are given a data codebook and the research question but otherwise exercise broad discretion over sample construction, variable definitions, and specification choices.

The study organizes submissions into three tasks that vary the degree of researcher discretion. *Task 1*, the high-discretion condition, most closely resembles standard applied economics practice: analysts receive only the research question and codebook and make all methodological choices independently. *Tasks 2* and *3* impose progressively more structure, narrowing the choice set by prescribing sample restrictions and then estimation specification. The pipeline used here mimics the Task 1 environment because the prompt fixes only the research question and data documentation, leaving all remaining choices to the AI agent.

Across Task 1 submissions, the point-estimate distribution in Huntington-Klein *et al.* (2025) is wide: the unweighted mean is 0.053, the median is 0.030, and the IQR spans 0.014 to 0.051, with a maximum of 0.660. Sample sizes range considerably (median 179,960; IQR 61,600–356,787), reflecting a large divergence in how analysts bound their estimation samples even when given identical raw data. The human-analyst results therefore exhibit exactly the kind of researcher-induced heterogeneity this paper seeks to characterize and replicate computationally at scale.

Importantly for my purposes, Huntington-Klein *et al.* (2025) have publicized the instructions and codebook used for the exercise. Their instructions direct authors to download whichever variables they deem necessary from specific ACS extracts on IPUMS. To streamline the AI-agent workflow and to curb the pressure on the IPUMS API, I instead make available to the AI agents a large extract with a wide range of potentially relevant variables. I also obtain a set of state policy variables and an associated codebook from the Huntington-Klein *et al.* (2025) public repository and make it available to all AI-agent runs.

4.2 Summary statistics and distributions

The final analytic sample contains 139 AI agent runs.[2] The remaining 148 sessions preserve a machine-readable specification, of which 125 come from GPT 5.1 Codex Mini and 23 from GPT 5.4.

Among those 148 sessions, five produce specifications that are implementable but yield extreme or degenerate samples: three collapse to samples with no treatment variation, one retains no observations after filtering, and one hangs during execution without returning numeric output. This leaves 143 numeric estimates. Four of those are degenerate zero-effect, zero-standard-error outputs, so the final analytic sample used below contains $N = 139$ runs with recoverable specifications and positive standard errors.

Table 1 reports summary statistics for point estimates, standard errors, and sample sizes across AI agent executions. The pooled point-estimate distribution centers near zero and is wide, with a median of 0.004, IQR from −0.042 to 0.046, mean 0.006, and inverse-SE weighted mean −0.006. The full range spans −0.396 to 0.332. Sample sizes vary substantially (mean 105,957; median 82,264; IQR 53,232 to 125,521). Choices about year windows, age bounds, citizenship definitions, and immigration-timing rules continue to move the effective estimation sample considerably.

Table 1: Distribution of executed estimates across AI-agent-generated research choices

Label	N	Mean	SD	Min	Pctl. 25	Median	Pctl. 75	Max
Effect Size (unweighted)	139	0.006	0.081	-0.396	-0.042	0.004	0.046	0.332
Effect Size (weighted by inverse SE)	139	-0.006	0.062	-0.396	-0.045	-0.003	0.028	0.332
Standard Error	139	0.008	0.016	0.000	0.003	0.004	0.009	0.185
Sample Size	139	105,957	89,356	204	53,232	82,264	125,521	521,998

Notes: The sample includes runs with recoverable specifications; this excludes validation failures, interrupted runs with no results file, and degenerate outputs with non-positive standard errors. The inverse-SE weighted row uses weights $1/\max(SE, q_{0.25})$ where $q_{0.25}$ is the 25th percentile of SE. Standard errors are rounded for display, so very small positive values may appear as 0.000.

Sample sizes. Figure 1 displays the distribution of log sample sizes. The wide dispersion underscores how small differences in sample-definition logic—year windows, age bounds, eligibility timing rules—translate into large changes in the effective estimation sample.

Estimates and specification curve. Figure 2 shows the distribution of point estimates

[2] I have 177 attempted runs. Of these, 29 never preserve a recoverable specification and therefore cannot enter the analysis. Common causes include the session terminating before the agent completed its work—due to connection timeouts, authentication failures, or platform-imposed token limits—rather than substantive analytic failures.

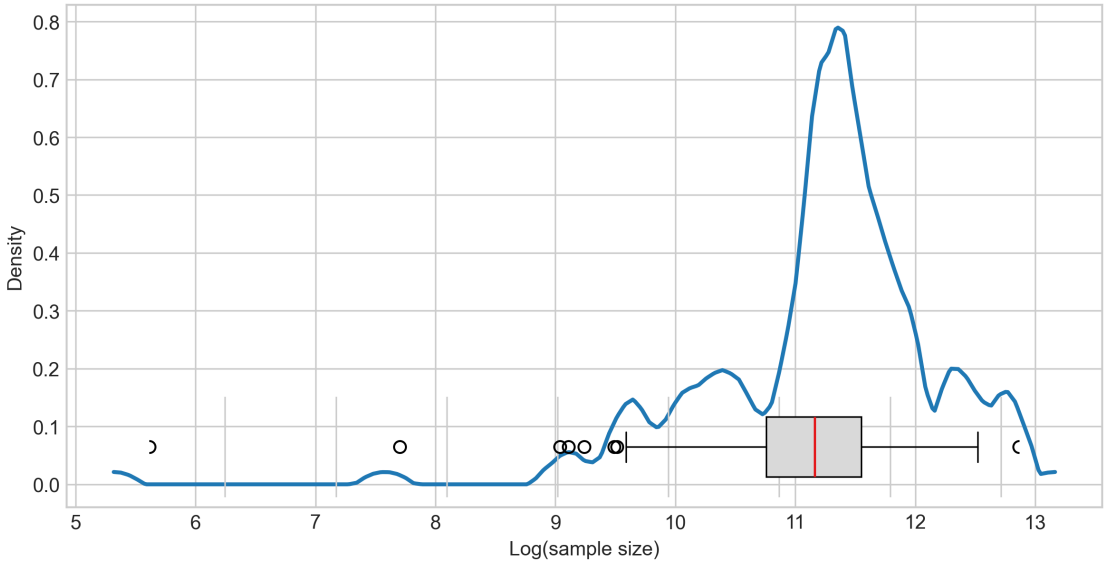


Figure 1: Distribution of executed sample sizes

Notes: Epanechnikov kernel density of $\log(n)$ across executed research choices, using Scott bandwidth, a standard rule-of-thumb bandwidth choice based on sample size and dispersion, with an overlaid box-and-whisker strip (median, IQR, min/max). Mirrors the benchmark study's descriptive sample-size distribution.

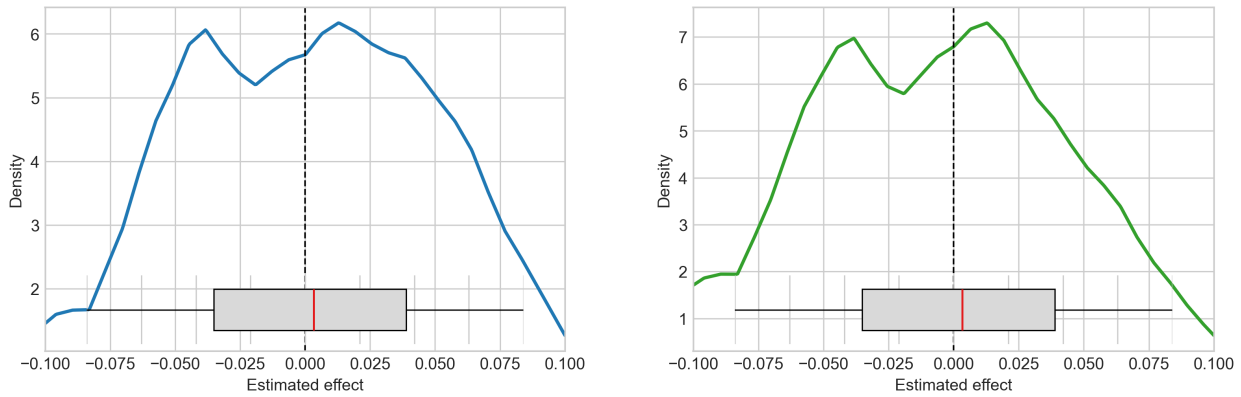
(unweighted and inverse-SE weighted); Figure 3 plots a specification curve ordered by magnitude.

Two patterns stand out. First, inverse-SE weighting leaves the center largely unchanged, suggesting the central mass is not driven by low-precision runs alone. Second, both panels truncate at $[-0.1, 0.1]$ for visual clarity, but Table 1 reveals that a small number of runs produce substantially larger magnitudes, particularly on the positive side. These outliers matter: they are precisely the estimates that selective reporting could emphasize if research choices were tried adaptively.

4.3 Comparison to the benchmark multi-analyst study

Huntington-Klein *et al.* (2025) study the same causal question and report dispersion across human researchers. I design the pipeline outputs as like-for-like counterparts: effect-size distributions (Figures 2 and 3), sample-size distributions (Figure 1), and summary statistics (Table 1).

The most directly comparable benchmark is Huntington-Klein et al.'s "Task 1," the high-discretion condition in which analysts answer the same DACA–employment question while retaining broad freedom over sample construction, variable definitions, and specification



(a) Unweighted

(b) Weighted by inverse SE

Figure 2: Distribution of executed DACA employment estimates

Notes: Each panel shows an Epanechnikov kernel density of point estimates, using Scott bandwidth, a standard rule-of-thumb bandwidth choice based on sample size and dispersion, with an overlaid box-and-whisker strip (median, IQR, min/max). For visual clarity, the x-axis is restricted to $[-0.1, 0.1]$, so tails beyond this range are not shown.

choices. That is the closest analogue to the pipeline used here, which also leaves those choices to the AI agent. In Table 3 of Huntington-Klein *et al.* (2025), the unweighted Task 1 distribution has mean 0.053, median 0.030, and IQR from 0.014 to 0.051 (max 0.660). The pooled AI agent distribution has mean 0.006, median 0.004, and IQR from −0.042 to 0.046 (max 0.332). Sample-size dispersion remains substantial in both settings: the benchmark median is 179,960 (IQR 61,600–356,787) versus 82,264 (IQR 53,232–125,521) in the retained AI agent sample.

Two caveats apply. The distribution here reflects (i) the AI-agent-induced support over research choices under the prompt design and command-line execution environment and (ii) selection into saved specifications and successful execution. The benchmark reports Task 1–3 objects under different constraints; the pipeline here mirrors the high-discretion Task 1 setting.

Tables 2 and 3 report method shares and mean effects by control-variable functional form. In the benchmark, 82% of submissions use linear regression, 22% use no SE adjustment, and 25% use sample weights. The pooled AI agent sample places even more mass on linear models: 98.6% of retained runs classify as OLS or WLS, 74.1% use the ACS person weight, 38.1% use robust standard errors, and 14.4% state-clustered standard errors. Mean effects also move with control structure: specifications including year fixed effects average 0.012 versus 0.000 without them, and specifications including state fixed effects average 0.022 versus −0.003 without them. Linear age controls are associated with more positive estimates (0.038) than

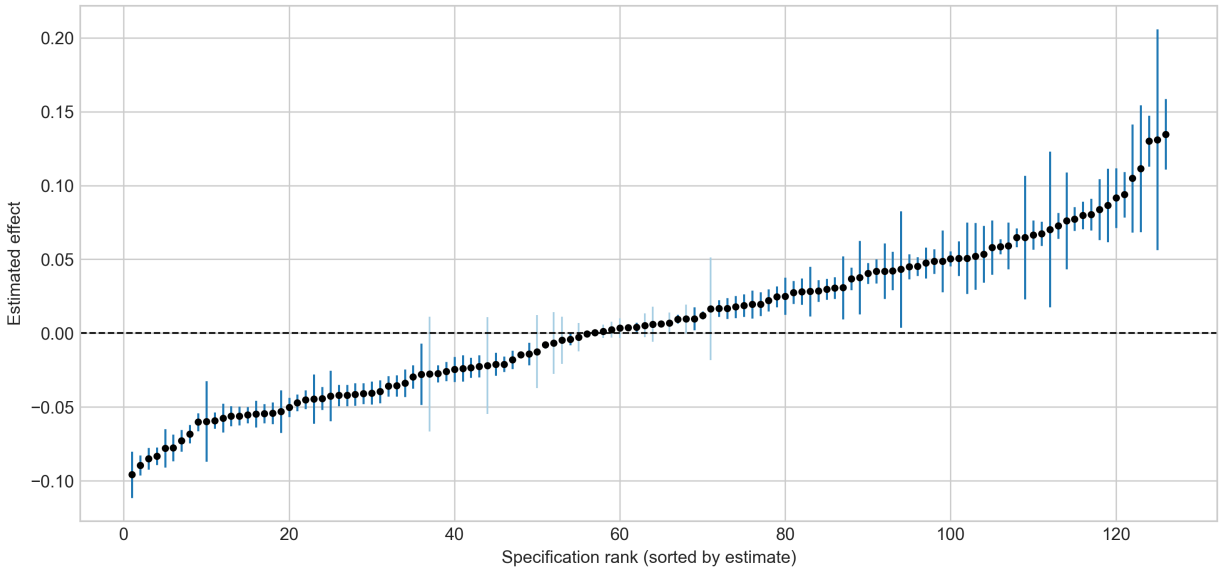


Figure 3: Specification curve of executed DACA employment estimates

Notes: Each point is one executed run, ordered by the point estimate. Vertical bars are 95% confidence intervals computed as $\hat{\theta} \pm 1.96\,SE$ using each run's reported standard error. For visual clarity, the figure omits specifications outside $[-0.1, 0.15]$.

specifications omitting age entirely (-0.006), although age controls still appear in a minority of retained runs. This is directionally consistent with the broader finding of Huntington-Klein *et al.* (2025) that routine modeling choices materially affect estimates.

Table 2: Shares of executed research choices by estimation method, weighting, and variance adjustment

Category	Choice	N	Share (%)
Estimation Method	OLS	28	20.1
	WLS	109	78.4
	Logit	2	1.4
Sample Weighting	None	29	20.9
	Other	7	5.0
	PERWT	103	74.1
SE Adjustment	Clustered	1	0.7
	Clustered (state)	20	14.4
	None	65	46.8
	Robust (HC)	53	38.1

Notes: Estimation methods are classified from the estimator code itself, including common closed-form implementations of linear regression. Logit is the only retained non-linear estimation method. "PERWT" refers to the ACS person weight. "Clustered (state)" indicates cluster-robust standard errors at the state level.

Table 3: Mean effects by functional form of common controls

Control Variable	Functional Form	N	Mean Effect	SD
Age	Linear	28	0.038	0.074
Age	Quadratic	2	0.187	0.205
Age	Not included	109	-0.006	0.075
Sex	Fixed effects	14	0.057	0.092
Sex	Not included	125	0.000	0.078
Education	Fixed effects	6	0.039	0.034
Education	Not included	133	0.004	0.083
Year FE	Included	67	0.012	0.061
Year FE	Not included	72	0.000	0.097
State FE	Included	48	0.022	0.063
State FE	Not included	91	-0.003	0.088

Notes: For each control variable, the table reports the mean and standard deviation of point estimates among executed research choices using that functional form.

I view these comparisons as validation targets. Agreement in broad qualitative patterns supports stochastic replication agents as a scalable approximation to multi-analyst exercises; discrepancies reveal differences in what kinds of specifications each approach generates and can execute.

4.4 Heterogeneity

There is substantial variation across specifications. Table 2 shows that AI agents disproportionately choose linear models: 78.4% of retained runs use WLS, 20.1% use OLS, and only 1.4% use Logit. The main contrast with the benchmark human sample is therefore the strong tilt toward WLS, and away from nonlinear estimators.

Table 3 shows that even within this narrower design space, routine specification choices continue to move estimates. Year fixed effects shift the mean from 0.000 to 0.012, state fixed effects shift it from −0.003 to 0.022, and linear age controls shift it from −0.006 to 0.038.

4.5 Differences between GPT 5.1 Codex Mini and GPT 5.4

Table 4 compares the two GPT models used. OpenAI documents GPT-5.4 as its flagship model for complex reasoning and coding, with a 1.05 million-token context window, whereas GPT-5.1 Codex Mini is a smaller, cheaper, less-capable Codex model with a 400,000-token context window (OpenAI Developers, 2026, 2025). Beyond context window size, the models likely differ in their post-training. Post-training refers to the stage after broad pretraining when a model is further tuned on demonstrations, rankings, and reinforcement-learning objectives so that it better follows instructions and solves target tasks (Ouyang *et al.*, 2022). Differences in capacity, context length, and post-training emphasis can therefore show up as differences in planning, tool use, and error recovery in the empirical application. In this context, GPT 5.4 better matches the benchmark human Task 1 sample: its retained estimates are closer to the benchmark's positive center, and its specifications more often use the weighted regressions, fixed effects, and clustered standard errors common in the human submissions.

The GPT 5.1 Codex Mini sample remains centered near zero: across 116 retained runs, the mean estimate is −0.002, the median is −0.003, and the IQR is [−0.045, 0.030]. The 23 retained GPT 5.4 runs are notably more positive: the mean is 0.048, the median is 0.052, and even the 25th percentile remains positive at 0.031. Every retained GPT 5.4 run uses WLS and the ACS person weight, roughly 70% request state-clustered standard errors, 83% include year fixed effects, and 74% include state fixed effects. By contrast, GPT 5.1 Codex Mini explores a broader design space with much less clustering and many more unweighted or non-fixed-effects specifications.

These model differences line up with differences in the executed samples. The median retained GPT 5.4 sample contains 37,930 observations versus 91,173 for GPT 5.1 Codex Mini, and the median standard error is correspondingly larger (0.0127 versus 0.0039). That helps explain why GPT 5.4 shifts the unweighted pooled distribution upward while leaving

the inverse-SE weighted mean near −0.006. Substantively, the GPT 5.4 tranche appears to favor a more benchmark-like empirical template—weighted regressions with fixed effects and clustered standard errors—but on smaller, tighter samples that still produce noticeably positive point estimates. The within-CLI comparison therefore reinforces a broader lesson of the paper: even after fixing the interface and prompt, the particular frontier model used to instantiate the replication agent materially affects the induced distribution of estimates.

Table 4: Point-estimate distribution by Copilot model

Statistic	GPT 5.1 Codex Mini	GPT 5.4
N	116	23
Mean point_est	-0.002	0.048
SD point_est	0.070	0.118
IQR point_est	0.075	0.050

Notes: Statistics are computed over the same filtered sample used in Table 1. Labels refer to the Copilot model used to generate and execute the combined phase 1–2 session.

5 Conclusion

Standard errors measure sampling uncertainty but ignore the often-substantial variation that arises from defensible differences in how researchers construct samples, define variables, and specify models. This paper treats that variation as an estimable object. By leveraging AI agents, I obtain a scalable empirical distribution of researcher-induced uncertainty. In the DACA–employment application, the resulting 139 retained estimates have a pooled mean of 0.006, median 0.004, and IQR [−0.042, 0.046], with broad qualitative resemblance to the benchmark human-analyst study.

This approach has several limitations. First, the benchmark multi-analyst materials were public well before these runs on the Open Science Framework (OSF) and GitHub (Portner and Huntington-Klein, 2022; Huntington-Klein, 2023; Huntington-Klein *et al.*, 2025). I do not have direct evidence that OpenAI specifically scraped OSF, but OpenAI documents training on large public web corpora and, for Codex, on publicly available GitHub code (Brown *et al.*, 2020; Chen *et al.*, 2021). Second, different defensible specifications can imply different estimands by changing the treated population, timing window, or treatment definition. Third, model choice matters within the AI-agent design itself: GPT 5.4 comes closer to the benchmark human Task 1 sample than GPT 5.1 Codex Mini, but the two models explore meaningfully different empirical templates.

For empirical practice, the framework offers a diagnostic: researchers can run the pipeline on their own question and data to assess how sensitive results are to the space of defensible alternatives, flagging specifications that produce qualitatively different conclusions. The open-source pipeline makes this diagnostic reproducible.

References

Andrews, D. W. and Buchinsky, M. (2000). A three-step method for choosing the number of bootstrap repetitions. *Econometrica*, **68** (1), 23–51.

Barnard, M., Huling, J. D. and Wolfson, J. (2026). A framework for causal estimand selection under positivity violations. *Biometrics*, **82** (1), ujag014.

Blair, G., Cooper, J., Coppock, A. and Humphreys, M. (2019). Declaring and diagnosing research designs. *American Political Science Review*, **113** (3), 838–859.

Brand, J., Israeli, A. and Ngwe, D. (2023). *Using LLMs for Market Research*. HBS Working Paper 23-062, Harvard Business School, april 2023; revised October 2025.

Breznau, N., Rinke, E. M., Wuttke, A., Adem, M., Adriaans, J., Alvarez-Benjumea, A., Andersen, H. K. *et al.* (2022). Observing many researchers using the same data and hypothesis reveals a hidden universe of uncertainty. *Proceedings of the National Academy of Sciences*, **119** (44), e2203150119.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A. *et al.* (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165.

Chang, T.-H., Nguyen, T. Q. and Jackson, J. W. (2024). The importance of equity value judgments and estimator-estimand alignment in measuring disparity and identifying targets to reduce disparity. *American Journal of Epidemiology*, **193** (3), 536–547.

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. d. O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G. *et al.* (2021). Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374.

Choi, J., Dekkers, O. M. and le Cessie, S. (2023). Tying research question and analytical strategy when variables are affected by medication use. *Pharmacoepidemiology and Drug Safety*, **32** (6), 661–670.

Efron, B. (1979). Bootstrap methods: Another look at the jackknife. *Annals of Statistics*, **7** (1), 1–26.

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N. *et al.* (2020). The pile: An 800gb dataset of diverse text for language modeling. arXiv preprint arXiv:2101.00027.

Gelman, A. and Loken, E. (2013). The garden of forking paths: Why multiple comparisons can be a problem, even when there is no "fishing expedition" or "p-hacking" and the research hypothesis was posited ahead of time. *Unpublished manuscript*.

Horton, J. J., Filippas, A. and Manning, B. S. (2023). *Large Language Models as Simulated Economic Agents: What Can We Learn from Homo Silicus?* NBER Working Paper 31122, National Bureau of Economic Research.

Huntington-Klein, N. (2023). Nickch-k/repl_data. GitHub repository, public repository containing DACA prepared-data documentation and code, including Prepared Data Documentation.html and prepare_data.R; GitHub metadata report creation on 2023-08-14.

—, Arenas, A., Beam, E., Bertoni, M., Bloem, J. R., Burber, P., Chen, N., Grieco, P., Ekpe, G., Pugatch, T. *et al.* (2021). Influence of hidden researcher decisions in applied microeconomics. *Economic Inquiry*, **59** (3), 944–960.

—, P"ortner, C. C., Acharya, Y. *et al.* (2025). *The Sources of Researcher Variation in Economics*. I4R Discussion Paper 209, Institute for Replication (I4R), version accessed from replication-materials/I4R-DP209.pdf in this repository.

Korinek, A. (2023). Generative ai for economic research: Use cases and implications for economists. *Journal of Economic Literature*, **61** (4), 1281–1317.

Lundberg, I., Johnson, R. and Stewart, B. M. (2021). What is your estimand? defining the target quantity connects statistical evidence to theory. *American Sociological Review*, **86** (3), 532–565.

Novy-Marx, R. and Velikov, M. (2026). Artificial intelligence–powered (finance) scholarship. *Journal of Economic Literature*, **64** (1), 5–37.

Olken, B. A. (2015). Promises and perils of pre-analysis plans. *Journal of Economic Perspectives*, **29** (3), 61–80.

OpenAI Developers (2025). Gpt-5.1 codex mini. API model documentation, describes GPT-5.1 Codex Mini as a smaller, more cost-effective, less-capable version of GPT-5.1-Codex.

OpenAI Developers (2026). Gpt-5.4. API model documentation, describes GPT-5.4 as OpenAI's frontier model for complex professional work with a 1.05M-token context window.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A. *et al.* (2022). Training language models to follow instructions with human feedback. arXiv preprint arXiv:2203.02155.

Portner, C. C. and Huntington-Klein, N. (2022). Many economists. OSF project, public Many Economists project on OSF; project metadata report creation on 2022-10-05 and updates through 2025-02-24.

Silberzahn, R., Uhlmann, E. L., Martin, D. P., Anber, J., Aust, F., Awtrey, E., Bahník, v., Bai, F., Bannard, C., Bonnier, E. *et al.* (2018). Many analysts, one data set: Making transparent how variations in analytic choices affect results. *Advances in Methods and Practices in Psychological Science*, **1** (3), 337–356.

Simmons, J. P., Nelson, L. D. and Simonsohn, U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, **22** (11), 1359–1366.

Simonsohn, U., Simmons, J. P. and Nelson, L. D. (2020). Specification curve analysis. *Nature Human Behaviour*, **4**, 1208–1214.

Steegen, S., Tuerlinckx, F., Gelman, A. and Vanpaemel, W. (2016). Increasing transparency through a multiverse analysis. *Perspectives on Psychological Science*, **11** (5), 702–712.

A Pipeline implementation details

This appendix provides technical detail on the four-stage workflow summarized in Section 3. Figure 4 presents the core algorithm alongside a schematic of the pipeline architecture.

Three design principles guide the architecture: (i) *auditability*—every run stores a full provenance chain from the research-problem description through the final numeric output; (ii) *isolation*—each execution runs in its own directory with no shared state, so that research choices are independent draws; and (iii) *portability*—resource locations resolve dynamically,

Algorithm Replication at scale with stochastic replication agents

Input: Research problem P; runs K; stochastic LLM behavior; dataset ω

for $k = 1, \ldots, K$ **do**

1. Sample research choices $r_k \sim \pi_{\text{AI}}(\cdot \mid P, \tau)$ Phase 1
2. Execute r_k on ω in an isolated workspace Phase 2
3. Record $(\hat{\theta}_k, \widehat{SE}_k, n_k)$

end for

4. Summarize $\{\hat{\theta}_k\}_{k=1}^{K}$ and relate to research-choice features Phase 3

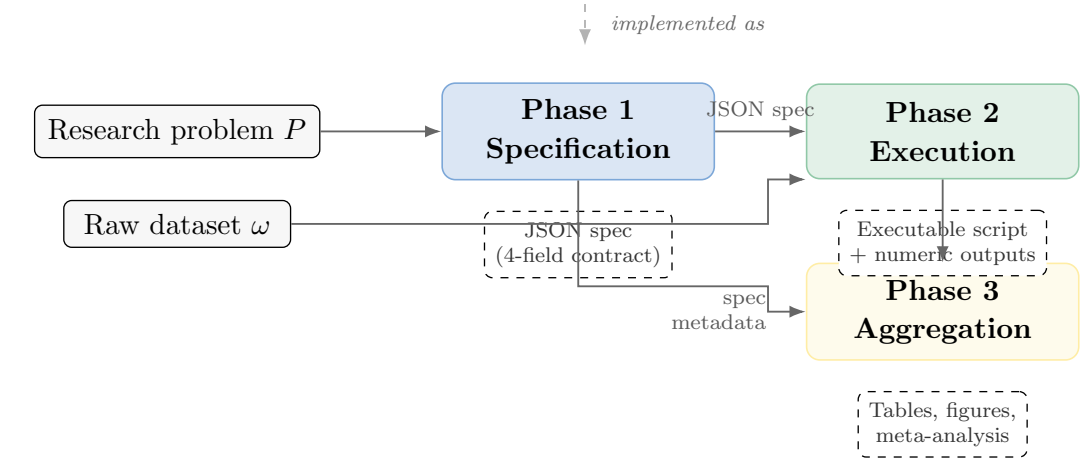


Figure 4: Replication-at-scale pipeline: algorithm and architecture

Notes: The top panel states the core algorithm; color annotations map each step to the pipeline phase below. The bottom panel compresses the workflow into three visual blocks: Phase 1 generates a structured specification via a stochastic LLM call, Phase 2 executes each specification in an isolated sandbox, and the final block summarizes downstream aggregation and meta-analysis. In the workflow used here, that final block corresponds to Phase 3 aggregation into `runs_complete.csv` and production of the manuscript tables and figures. Only the AI agent's stochastic research choices vary across runs.

allowing the pipeline to run across operating systems and local environments without code changes.

Specification contract

Phase 1 requires the AI agent to return a JSON object with four fields:

```
{
  "sample_selection":        [<filter condition>, ...],
  "outcome_definition":      "<Python expression>",
  "treatment_definition":    "<Python expression>",
```

```
  "model_specification_line": "<executable Python line>"
}
```

The contract is deliberately minimal: it pins down what every set of research choices must produce (an estimator applied to an outcome, treatment, and sample) while leaving maximum freedom in how those objects are defined. The `model_specification_line` must be a single executable Python statement calling an estimator (e.g., `smf.ols(...).fit()`) and may embed controls, fixed effects, sample weights, and clustering choices. The contract is enforced by a JSON Schema validated before any code executes.

In this implementation, Phase 1 is executed through GitHub Copilot CLI. The system extracts the JSON object from the model response, validates it against the schema, and stores metadata for auditing.

Execution sandbox

Phase 2 takes each validated specification and produces an executable analysis script plus numeric results. The steps are:

1. **Create an isolated workspace.** Each run receives only whitelisted input data and documentation files.

2. **Link large inputs.** Large data assets are shared across runs via platform-appropriate linking, with safe fallbacks.

3. **Invoke a coding agent.** The agent receives the structured specification, generates analysis code, and executes it with bounded self-correction for runtime errors.

4. **Validate output.** The pipeline verifies the result object exists, parses correctly, and contains numeric fields for point estimate, standard error, and sample size. Runs that fail validation, never write a results file, or produce degenerate outputs with non-positive standard errors are flagged explicitly in the archive.

A combined mode allows a single agent session to both propose and implement a specification, useful for agents that benefit from end-to-end context.

Aggregation and classification

Phase 3 aggregates the full session archive, not just the subset that already has clean final outputs. Each archived session receives a specification-status label (recoverable JSON

specification versus missing specification) and an execution-status label (successful result, validation failure, interrupted run with no results file, or degenerate numeric output with non-positive standard error). For sessions with recoverable specifications, the aggregator then infers derived metadata from `model_specification_line` using regular-expression classifiers:

- **Model type**: OLS, WLS, or Logit, inferred from estimator function names, explicit weights, and common closed-form implementations of linear regression.
- **Controls and fixed effects**: parsed from the formula's right-hand side, distinguishing continuous regressors from categorical (`C(...)`) terms.
- **Sample weighting**: detected from `weights=` arguments (typically the ACS person weight `PERWT`).
- **SE adjustment**: classified as clustered (state), heteroskedasticity-robust (HC), or none, from `cov_type` and `cov_kwds` arguments.

This classification operates on structured text (Python code), avoiding reliance on self-reported AI-agent metadata. The paper's quantitative analysis then keeps only sessions with recoverable specifications and `execution_status = success`.

The reporting step produces:

- **Summary statistics** (Table 1): Unweighted and inverse-SE-weighted moments and quantiles of the point-estimate distribution, plus standard-error and sample-size distributions.
- **Density plots** (Figures 2–1): Epanechnikov kernel densities with box-and-whisker overlays.
- **Specification curve** (Figure 3): Estimates ordered by magnitude with 95% confidence intervals.
- **Decision-share tables** (Tables 2–3): Cross-tabulations of estimation methods, weighting, SE adjustments, and control-variable functional forms.

All tables and figures are generated from the same integrated analysis dataset so that manuscript objects update automatically when the pipeline is re-run.

B Prompt excerpt and design

This appendix reproduces the key portion of the prompt used to elicit structured specifications of research choices. The full prompt also included extensive variable definitions and codebook material to support independent implementation; I omit that bulk for space and instead present the instructions that govern the required outputs and the feasible choice set.
Prompt text (governing instructions and required output).

```
Research question:
Among ethnically Hispanic-Mexican Mexican-born people living in the United States,
what was the causal impact of eligibility for the Deferred Action for Childhood
Arrivals (DACA) program on the probability that the eligible person is employed
full-time, defined as usually working 35 hours per week or more?

DACA was implemented in 2012. Estimate effects on full-time employment in 2013{2016.

KEY DELIVERABLES | ONLY RETURN THE FOLLOWING USING PYTHON:
Your answer MUST contain ONLY a JSON block with the following structure:

{
  "sample_selection": ["<filter condition 1>", "<filter condition 2>", "..."],
  "outcome_definition": "<Python expression for outcome variable>",
  "treatment_definition": "<Python expression for DACA eligibility>",
  "model_specification_line": "<exact Python line calling the estimator>"
}

Rules:
- Provide implementable filters and expressions consistent with the available data.
- The model_specification_line must be a single executable Python line that:
  (i) estimates the effect of treatment_definition on outcome_definition; and
  (ii) encodes any controls, fixed effects, weights, and SE/clustering choices.
- Be explicit in sample_selection about years, age bounds, nativity/origin criteria,
  and any restrictions used to approximate eligibility rules.
- Do NOT include any commentary outside the JSON block.
```

The prompt fixes the research question and the required output contract but allows discretion over sample restrictions, functional form, controls, fixed effects, weighting, and variance estimation. This discretion is the mechanism that generates dispersion across executed estimates when the prompt is held fixed and the generator is stochastic.