Claude Code as an Empirical Economist: Like Humans but Without the Tails

Serafin Grundl*

February 9, 2026

Abstract

This paper compares Claude Code (Opus 4.5) and human economists on the same causal inference tasks following the same instructions. Human benchmark results and instructions come from Huntington-Klein, Pörtner, McCarthy, and the Many Economists Collaborative on Researcher Variation (2025). The main findings are that (1) the means and medians of the causal effect estimates are fairly similar between Claude Code and human researchers, and (2) the dispersion of estimates across Claude Code runs is substantial but dispersion among humans is 2–27 times larger. Thus Claude Code produces estimates that resemble those of human researchers in the middle of the distribution, while avoiding the tails. These results suggest that AI systems like Claude Code can serve as scalable tools for producing novel empirical research, large-scale replication, and robustness analysis of the existing literature. The reports and code written by Claude Code can be browsed at claude-code-economist.com.

1 Introduction

Large language models (LLMs) deployed in agentic coding environments can now write statistical code, execute it, interpret the output, debug errors, and iterate autonomously—performing many of the steps that a human economist would. But empirical economics

*Federal Reserve Board of Governors, 1801 K St. NW, 20006 Washington DC, serafin.j.grundl@frb.gov. The analysis and conclusions set forth are those of the authors and do not indicate concurrence by other members of the staff, by the Board of Governors or the Federal Reserve Banks. For this project Claude Code ran on my personal computer. Claude Code was also used extensively for writing, data analysis, and visualization. ChatGPT and Gemini were used as well. I thank Nick Huntington-Klein, Claus C. Pörtner, Ian McCarthy, and the Many Economists Collaborative on Researcher Variation for making their data and instructions publicly available.

requires more than coding: it requires reasoning about identification, choosing a research design, constructing appropriate samples, and writing up results. This raises a natural question: can AI agents perform the full pipeline of empirical research traditionally done by human economists? If so, AI-conducted analysis could dramatically increase the scalability and reduce the cost of empirical research.

This paper provides a direct answer by comparing AI and human researchers on the same causal inference task under identical instructions. I run 100 independent instances of Claude Code (Opus 4.5) on each of three progressively constrained tasks from Huntington-Klein et al. (2025), in which 146 human research teams estimated the effect of DACA eligibility on full-time employment using American Community Survey data. Each instance operates autonomously—with no memory of other instances' work—and produces a complete research report with code, tables, and written interpretation. Task 1 provides only raw data and a research question, Task 2 specifies the research design, and Task 3 additionally provides a pre-cleaned dataset. This progressive-constraint structure allows me to compare how AI and human analysts respond to the same reduction in analytical freedom.

The first main finding is that Claude Code's central estimates are close to those of human researchers. In Task 1, where analysts have full freedom, Claude Code's median estimate (2.7 percentage points) nearly matches the human weighted median (2.6 pp) and human unweighted median (3.0 pp).[1] The means differ more in Task 1—Claude Code's mean (2.9 pp) falls below the human means (4.4 pp weighted, 5.3 pp unweighted)—but this gap is driven by a handful of large human estimates in the right tail that inflate the human mean well above the human median. In Tasks 2 and 3, the mean gap narrows—Claude Code's means of 4.5 and 6.1 pp are within 0.1 pp of the human weighted means (4.6, 6.2 pp) and close to the human unweighted mean for Task 2 (4.4 pp), though the unweighted mean in Task 3 is notably lower (4.5 pp). The medians differ somewhat for Tasks 2 and 3, with Claude Code's medians (4.6, 6.1 pp) exceeding the human weighted medians (3.4, 5.1 pp) and human unweighted medians (3.2, 5.0 pp), though across all tasks and weighting schemes the median estimates differ by at most 1.4 percentage points and the weighted mean estimates are within 0.1 pp in Tasks 2 and 3 (the unweighted mean gap is larger in Task 3, at 1.6 pp).[2]

[1] Following Huntington-Klein et al. (2025), "human weighted" statistics use inverse-standard-error weights, reducing the influence of noisy estimates, with weights truncated at the 95th percentile; "human unweighted" statistics weight all teams equally.

[2] Prescribing the research design (Task 1 to Task 2) shifted Claude Code's median upward by 70% (from 2.7 to 4.6 pp), whereas the human weighted median rose 31% (from 2.6 to 3.4 pp) and the human unweighted median rose only 7% (from 3.0 to 3.2 pp). Huntington-Klein et al. (2025) attribute the limited human response to imperfect adherence to the prescribed design: the Task 2 distribution of human estimates is bimodal, with one mode near the Task 1 estimates and another near the Task 3 estimates. Only about 20–25% of human researchers implemented the prescribed treated-group definition exactly, and their estimates cluster near 5 pp—close to the Task 3 median.

Expressed as ratios of human to Claude Code central tendencies, the median ratios range from 0.70 to 1.11 across tasks and weighting schemes; the mean ratios range from 0.74 to 1.83.

Second, the dispersion of estimates across Claude Code runs is economically meaningful but substantially smaller than across human teams. In Task 1, the 10th and 90th percentile Claude Code estimates span 0.9 to 5.3 pp—a 4.4 pp range from a near-zero to a substantial employment effect. For Tasks 1 and 2, the standard deviation of point estimates across Claude Code runs exceeds the average standard error, indicating that specification choices—not sampling variability—are the dominant source of dispersion. Yet human estimates are far more dispersed: the ratio of human to Claude Code standard deviations ranges from 4.6 to 12.9 across tasks, with IQR ratios of 2.1 to 8.6. The range (maximum minus minimum) of human estimates is 6 to 27 times wider than Claude Code's. The standard deviation gap widens as constraints increase: Claude Code's standard deviation falls monotonically from 2.0 pp in Task 1 to 1.3 pp in Task 2 to 0.8 pp in Task 3, while the human unweighted standard deviation remains roughly constant at 9.5–10.1 pp across all three tasks. Thus Claude Code produces estimates that resemble those of human researchers in the middle of the distribution, while avoiding the tails. It exercises its analytical discretion within a narrower range than humans.

Third, prescribing a shared research design (moving from Task 1 to Task 2) reduces the standard deviation across Claude Code runs by 35%, the IQR by 67%, and the range by 52%. Among humans, the IQR and range both increase for weighted and unweighted estimates; the unweighted standard deviation also increases slightly, though the weighted standard deviation drops.[3]

Fourth, a systematic audit of all 300 analysis scripts—comprising two independent verification rounds and a code quality review, each conducted by Claude Code agents (Section 4)—identified coding errors that affect the preferred point estimate in 10 of 300 replications (3.3%): 8 cases of integer overflow in a quadratic age control and 2 cases of inert education dummies due to a type mismatch. These figures are a lower bound, as additional undetected errors likely exist. At the same time, this rate must be weighed against the fact that human researchers also make mistakes—an observation well documented in the replication literature—and a rate below 4% falls within a range that many practitioners would consider acceptable for a first-pass analysis.

These results suggest that AI systems like Claude Code can serve as scalable tools for producing novel empirical research, large-scale replication, and robustness analysis of the

[3] As discussed above, Huntington-Klein et al. (2025) attribute the limited convergence among humans to imperfect adherence to the prescribed design.

existing literature. An important caveat, however, is that these systems can and do make mistakes. As noted above, some Claude Code replications contain discrepancies between the code and the written report—for instance, stating that robust standard errors were used when the code reveals they were not. Such errors are consequential and underscore the need for human oversight of AI-generated research. At the same time, this concern must be weighed against the well-documented fact that human researchers also make coding errors, misreport results, and produce analyses that do not fully match their written descriptions—mistakes that are often harder to detect because human code is less consistently documented. Moreover, the capabilities of frontier AI systems are improving rapidly, and the types of discrepancies observed here may diminish in future model generations.

This paper contributes to a growing literature comparing AI and human performance on professional tasks. Recent studies have benchmarked LLMs against lawyers (Katz et al., 2024), physicians (Goh et al., 2024; Kung et al., 2023), and economic forecasters (Halawi et al., 2024; Faria-e Castro and Leibovici, 2024). A separate strand examines how AI augments human productivity in professional settings (Dell'Acqua et al., 2023; Noy and Zhang, 2023; Brynjolfsson et al., 2025). Our study belongs to the first category but differs in that we compare AI and human performance on an open-ended empirical research task—one requiring the full pipeline from data cleaning to estimation to interpretation—rather than on a structured exam or narrowly defined task. Moreover, we focus not on whether AI matches human accuracy on a task with a known correct answer, but on how the *distribution* of AI outputs compares to the distribution of human outputs on a task where the "correct" answer is itself contested.

The remainder of the paper is organized as follows. Section 2 reviews the Huntington-Klein et al. (2025) study. Section 3 describes the experimental design. Section 4 verifies the reproducibility of Claude Code's output and conducts a systematic code quality review. Section 5 presents the distribution of Claude Code estimates across the three tasks. Section 6 compares these estimates to the human benchmark. Section 7 concludes. Appendix A provides illustrative examples of Claude Code's output, Appendix B contains additional figures, Appendix C surveys the many-analysts literature, and Appendices D–E provide replication instructions and the automation script.

2 The Huntington-Klein et al. (2025) Study

A growing literature documents that empirical research findings depend not only on data and theory but also on the analyst who conducts the work—a phenomenon variously described as "researcher degrees of freedom" (Simmons et al., 2011), "non-standard errors"

(Menkveld et al., 2024), or the "garden of forking paths" (see Appendix C for a review). Prior many-analysts studies have documented the existence and magnitude of this inter-analyst variation, but most have been limited in their ability to decompose it into component sources. Huntington-Klein et al. (2025) introduce the first large-scale many-analysts study in economics, with a design specifically engineered to decompose researcher variation into identifiable sources. They recruited 146 research teams that each completed the same causal inference task three times, under progressively tighter constraints on their analytical freedom.

The research question. All teams were asked to estimate the causal effect of eligibility for the Deferred Action for Childhood Arrivals (DACA) program on the probability of working full-time, among the population affected by the policy. DACA, implemented in August 2012, granted temporary legal work authorization and protection from deportation to undocumented immigrants who arrived in the United States as children, provided they met specific eligibility criteria: arrival before age 16, presence in the U.S. since June 15, 2007, no legal status as of June 15, 2012, age below 31 as of that date, and (in later task rounds) completion of at least high school or military service. The data source for all analyses was the American Community Survey (ACS) from IPUMS, covering years 2006–2016, with the sample focused on ethnically Hispanic-Mexican, Mexican-born individuals.

Three-task progressive-constraint design. The study's central innovation is a sequential narrowing of researcher degrees of freedom across three tasks:

- **Task 1 (Full Freedom).** Teams received the raw ACS data and the research question, with minimal constraints on how to conduct the analysis. Researchers were free to choose their own sample restrictions, research design (e.g., difference-in-differences, regression discontinuity, matching), variable definitions, estimation method, and standard error treatment. The only requirements were to use ACS data from IPUMS (one-year files, harmonized variables), to restrict data to 2006–2016, and to estimate the causal effect of DACA eligibility on full-time employment.
- **Task 2 (Specified Design).** The research design was substantially constrained. Teams were instructed to define a "treated" group of ethnically Mexican, Mexican-born, non-citizen individuals aged 26–30 as of June 15, 2012, and an "untreated" comparison group of individuals who would have been eligible except that they were aged 31–35 on that date. The task specified a difference-in-differences logic: compare how

outcomes for the treated group changed from before DACA (2006–2011) to after (2013–2016) relative to the change for the untreated group. However, teams still made their own decisions about data cleaning, variable construction, control variables, estimation method, and standard error computation. An additional eligibility criterion—that eligible individuals must have completed high school or be military veterans—was added in this round.

- **Task 3 (Pre-cleaned Data + Specified Design).** The design constraints from Task 2 were retained, and teams were additionally provided with a pre-cleaned dataset of approximately 17,382 observations prepared by the study organizers. This dataset included a constructed treated/untreated indicator, limited the sample to only the treated and untreated groups, handled missing-data flags, merged state-level policy variables, and provided standardized recodings of demographic variables. Researchers were instructed not to further restrict the sample. The only remaining degrees of freedom were the choice of estimation method, functional form of controls, and standard error computation.

The rationale for this progressive design is that by sequentially removing categories of researcher freedom, the study can attribute variation to specific choice types. The reduction in dispersion from Task 1 to Task 2 captures the contribution of research design choices (holding data cleaning constant). The reduction from Task 2 to Task 3 captures the contribution of data cleaning and sample construction (holding the research design constant). Any remaining variation in Task 3 reflects differences in estimation and inference choices alone.

Recruitment. The 146 research teams that completed all three tasks were recruited from applied microeconomics through social media, professional organization emails, and outreach to U.S. economics department chairs. Eligible participants included academic faculty, graduate students with a published or forthcoming paper, and non-academic researchers holding a PhD who work in applied causal inference. The final sample was 87% PhD holders, 67% faculty, and approximately 40% working in labor or immigration economics. Participants were offered $2,000 and co-authorship upon completion of all three tasks, and a regression discontinuity analysis confirmed that guaranteed payment did not meaningfully affect completion rates.[4]

[4] Following each task, two-thirds of researchers were randomly assigned to peer review pairs, with each member reviewing the other's work. Reviews were conducted as though for a journal submission, and researchers had the option—but not the obligation—to revise their work in response. The majority chose not to revise.

Data. The underlying data are from the American Community Survey (ACS) 2006–2016, accessed via IPUMS (Ruggles et al., 2024). Key variables include employment status (specifically, usual hours worked per week), age, year of immigration, citizenship status, education, Hispanic/Mexican ethnicity, and birthplace. In addition to the ACS data, the study organizers provided a state-by-year dataset containing labor market variables (unemployment rate, labor force participation rate) and indicators for state-level immigration policies (driver's license access, E-Verify laws, Secure Communities participation, etc.), sourced from the Urban Institute (Urban Institute, 2022).

3 Experimental Design

We conduct parallel replications using Claude Code, an AI coding agent developed by Anthropic that can read files, write and execute code, and interact with the file system. We run 100 independent instances of Claude Code on each of the three tasks from Huntington-Klein et al. (2025), for a total of 300 replications.

Instructions. The replication instructions given to Claude Code were identical to those given to the 146 human research teams in Huntington-Klein et al. (2025), as described in Section 2. Each of the three instruction documents was renamed to `replication_instructions.docx` but otherwise left unchanged. The full instructions are reproduced in the Appendix.

Data. The only departure from the human researchers' experience is that Claude Code did not download the raw data itself. Instead, the data was pre-downloaded following the instructions and provided as a CSV file. This modification was necessary because Claude Code cannot navigate interactive websites or authenticate with data repositories, and because it ensures that all instances work with identical raw data.

For Tasks 1 and 2, each instance received a `data` folder containing three files: (1) `data.csv`, a rectangular extract of approximately 33.9 million person-level observations from the ACS one-year files for 2006–2016, containing 54 harmonized IPUMS variables; (2) `acs_data_dict.txt`, the IPUMS-generated data dictionary; and (3) `state_demo_policy.csv`, an optional supplemental file with 16 state-by-year variables covering demographics and immigration policy (driver's license access, in-state tuition, E-Verify, Secure Communities, etc.), provided by the Huntington-Klein et al. (2025) study organizers. The ACS extract includes the core variables needed to identify DACA eligibility and measure full-time employment: survey year (`YEAR`), birth year and quarter (`BIRTHYR`, `BIRTHQTR`), Hispanic/Mexican ethnicity

(HISPAN, HISPAND), birthplace (BPL, BPLD), citizenship status (CITIZEN), year of immigration (YRIMMIG), usual hours worked (UHRSWORK), employment status (EMPSTAT), and person-level survey weights (PERWT), along with demographic and socioeconomic variables (age, sex, marital status, education, income, health insurance, family composition, and geography).

For Task 3, each instance received the pre-cleaned dataset prepared by the Huntington-Klein et al. (2025) study organizers (described in Section 2), along with the data dictionary. The run prompt explicitly told Claude Code that the data was already present and did not need to be downloaded.

Experimental Setup. Each of the 300 replications (100 per task) was executed as an independent instance of Claude Code in a separate working directory, with no shared memory or context between runs. A PowerShell script (reproduced in the Appendix) automated the process: for each of the 100 runs, the script created a new directory, copied in only the data file(s) and the replication instructions, then launched Claude Code with a prompt instructing it to read the instructions, perform the analysis, and produce a replication report. Up to four replications ran in parallel, with random startup delays (0–120 seconds) to avoid synchronized computation loads. The scripts for Tasks 2 and 3 were substantively identical, differing only in directory paths and data files.

Independence. Several features of the experimental design ensured independence across Claude Code instances. First, each instance ran in an isolated directory with no access to outputs from other runs. Second, the prompt explicitly framed each run as an "independent replication" to be treated as a "clean-room run." The stochastic nature of large language model outputs, combined with this isolation, means that each instance made its own decisions about data cleaning, variable construction, model specification, and estimation.

Outputs. Each Claude Code instance was instructed to produce two outputs: (1) a replication report of 20 pages in PDF format, documenting the analysis and results; and (2) a run log recording key decisions and commands. Most instances also saved their analysis code, though they were not explicitly instructed to do so: all 100 Task 2 replications saved their code, as did 98 of 100 for Task 1 and 83 of 100 for Task 3. For the remaining 19 replications, the code for the preferred specification was recovered from the run logs and replication reports. Appendix A provides illustrative excerpts from these outputs. Replication reports, log files, and code files are available at claude-code-economist.com.

Data extraction. From each replication's outputs, we extract the preferred point estimate, standard error, and sample size for comparison with the human replications. The point estimates and sample sizes were extracted from the run logs with the help of Claude Code agents. The standard error type (conventional, heteroskedasticity-robust, or clustered) was extracted from the analysis code rather than the log files, because Claude Code sometimes mislabels its standard errors in written output—for example, describing conventional standard errors as "robust" (see Section 4).

Model Version. All replications used Claude Opus 4.5 (model ID: `claude-opus-4-5-20251101`) via the Claude Code command-line interface.

4 Code Verification & Review

We audited all 300 replications in three passes, each conducted by Claude Code agents. First, two independent *verification rounds* re-executed every saved analysis script and compared the computed preferred coefficient, standard error, and sample size against the values reported in the run log and replication report; a replication passes if its coefficient and standard error each match within a tolerance of 0.001. The two rounds agreed on 297 of 300 replications (99.0%). Second, a *code quality review* combined an automated scanner for 12 known bug patterns with a manual review of each script's data filtering, variable construction, and estimation logic. For the 19 replications without a saved code file (2 in Task 1, 17 in Task 3), we used the code previously recovered from run logs and replication reports (see Section 3).

Reproducibility. All 300 replications reproduce their reported results from the saved code. One replication (Task 1, Rep 94) initially appeared to fail because its directory contains three script versions from the session; the saved `analysis.py` is an early iteration, but the later `analysis_final.py`—which was actually executed—reproduces the reported results. Approximately 10 replications across Tasks 2 and 3 describe their standard errors as "robust" in the replication report while the code produces conventional (homoskedastic) standard errors; point estimates are unaffected.

Integer overflow in the quadratic age control. Eight replications (6 in Task 1, 2 in Task 2) specify `int8` as the dtype for the `AGE` variable when loading the 6 GB raw data file. Because `int8` can only represent integers from -128 to 127, computing `age_sq = AGE ** 2` produces silent integer overflow—for example, $30^2 = 900$ wraps to -124. The resulting quadratic age control contains arbitrary wrapped values instead of a smooth function of age, so the regression cannot use it to absorb age–employment patterns and instead attributes this variation to the treatment effect, inflating the DiD coefficient. The `int8` specification reflects a deliberate memory optimization: all eight scripts specify compact dtypes for 8–15 columns simultaneously, and every script contains comments referencing memory constraints. The choice is technically correct for *storing* age values (all in the 0–95 range) but becomes a latent bug when squaring exceeds `int8`'s range and NumPy silently wraps rather than raising an error.

Table 1 quantifies the bias. The mean upward bias is 3.7 percentage points, representing 44–69% of the reported coefficient (range: 1.8–5.5 pp). The "No age^2" column shows that dropping the quadratic term entirely produces estimates nearly identical to the overflow values (mean absolute difference: 0.25 pp), confirming that the corrupted term contributes no explanatory power. The corrected estimates (2.0–4.2 pp in Task 1, 2.5 pp for Rep 97) fall within the normal range of the distribution. This finding parallels Huntington-Klein et al. (2025), who report that among human research teams "the choice of functional form had a greater impact than the selection of covariates," with the quadratic age specification producing substantially lower estimates than a linear-only age control.

Table 1: Effect of `int8` Overflow on the Preferred DiD Coefficient

Rep	Task	Reported (pp)	No age^2 (pp)	Corrected (pp)	Bias (pp)
07	1	8.7	8.8	4.2	4.5
10	1	6.5	6.6	2.4	4.1
24	1	5.2	5.2	2.0	3.2
45	1	7.0	7.1	2.7	4.3
76	1	7.3	7.4	3.0	4.3
97	1	8.0	8.0	2.5	5.5
40	2	4.2	4.9	2.3	1.9
90	2	3.6	4.3	1.8	1.8

Notes: All values in percentage points. "Reported" is the preferred DiD coefficient from the run log, computed with `AGE` stored as `int8`. "No age^2" drops the quadratic age term from the specification while keeping all other controls. "Corrected" re-runs the same script with `AGE` dtype changed from `int8` to `int32`. "Bias" is "Reported" minus "Corrected."

Inert control variables. Two Task 3 replications (16 and 52) construct education dummy variables by comparing the string-valued `EDUC_RECODE` column to integer values (e.g., `EDUC_RECODE == 2`). Because of the type mismatch, all comparisons evaluate to `False`, producing all-zero dummies that render the education controls inert. Re-running these scripts with correctly constructed dummies reduces the preferred coefficient from 6.4 to 6.1 pp in both cases—a difference of approximately 0.3 pp, or less than 5% of the reported estimate.

The error is understandable given the context. The data file is named `prepared_data_numeric_version.csv`, which strongly suggests that categorical variables have been integer-coded. The column name `EDUC_RECODE` reinforces this expectation: in IPUMS data, recoded variables typically use numeric codes. In fact, one of the two affected replications includes a comment documenting the assumed coding scheme (`# EDUC_RECODE: 1=Less than HS, 2=HS, ...`). However, `EDUC_RECODE` is stored as a string in both the "numeric" and "labelled" versions of the data file—the "numeric" label refers to other columns such as `STATEFIP`, which contains FIPS codes in one version and state names in the other. The bug goes undetected at runtime because `pandas` silently evaluates cross-type comparisons as `False` without raising a warning. The 98 replications that avoid this error either inspect the column values (e.g., via `.unique()`) before constructing dummies or use formula-based interfaces (`C(EDUC_RECODE)`, `pd.get_dummies`) that automatically detect the column type.

Summary. We identified coding errors that affect the preferred point estimate in 10 of 300 replications (3.3%): 8 integer overflow cases (Table 1) and 2 inert-control cases. Restricting to the 281 replications with original (non-reconstructed) code files, the rate is 10 of 281 (3.6%). These error rates should be regarded as a lower bound: it is possible, and indeed likely, that additional errors exist but were not detected. At the same time, the existence of coding errors in AI-generated analyses must be weighed against the fact that human researchers also make mistakes—an observation well documented in the replication literature—and a rate below 4% falls within a range that many practitioners would consider acceptable for a first-pass analysis. The estimates used in all comparisons in this paper (Sections 5–6) are the reported estimates drawn from the run logs, not corrected estimates. This ensures an apples-to-apples comparison with the human research teams, whose submitted estimates were likewise taken at face value.

Mitigations and design considerations. The coding errors documented above could likely be reduced through more explicit prompting or by deploying a separate audit agent to find errors. We deliberately chose not to implement such safeguards: Claude Code received the same instructions as the human research teams, and introducing automated auditing tools would compromise the apples-to-apples comparison.

5 Description of Claude Code Estimates

This section presents the estimates from 100 independent Claude Code replications for each of the three tasks. We examine the distribution of point estimates, sample sizes, and standard errors, and characterize the convergence pattern across tasks.

Point Estimates. Table 2, Panel A presents the summary statistics of Claude Code's point estimates across the three tasks, Figure 1 presents histograms of the point estimate distributions, and Figures 2–4 display forest plots of all 100 estimates per task with 95% confidence intervals. Several patterns are immediately apparent. First, the central tendency shifts upward as constraints increase: the median estimate rises from 2.7 pp in Task 1 to 4.6 pp in Task 2 to 6.1 pp in Task 3. Second, the dispersion declines: the standard deviation falls from 2.0 to 1.3 to 0.8 pp across the three tasks, and the interquartile range narrows from 1.5 to 0.5 pp in Task 2, widening slightly to 0.8 pp in Task 3. Even in the most dispersed task (Task 1), the standard deviation of estimates is only 2.0 pp. Although these standard deviations are small in absolute terms, they are economically meaningful: the difference between the 10th and 90th percentile estimates in Task 1 is 4.4 percentage points, spanning the range from a near-zero to a substantial employment effect. Moreover, the variation across Claude Code instances is large relative to the typical standard error of any single replication. In Task 1, the ratio of the standard deviation of point estimates to the average standard error is 4.32, indicating that analyst-driven variation far exceeds sampling uncertainty. This ratio falls to 1.21 in Task 2 and to 0.48 in Task 3. In other words, analytical choices are the dominant source of variation when Claude Code operates with full freedom, roughly comparable to sampling noise under a specified research design, and subordinate to sampling noise when the dataset is pre-cleaned.

Sample Sizes. Table 2, Panel B reports the sample size statistics, and Figure 9 in the Appendix displays the distribution of sample sizes across tasks. Both median sample sizes and their dispersion decline sharply across tasks.

In Task 1, where analysts freely construct their samples, the median is approximately 551,000, but sample sizes range widely from 43,238 to 771,888 (SD = 173,884), reflecting different decisions about age ranges, citizenship restrictions, and DACA eligibility pre-filters. In Task 2, where the research design is specified, the median drops to 43,238 and the dispersion narrows considerably (SD = 20,471), with nearly all replications producing samples between 42,558 and 53,490. In Task 3, where the dataset is pre-cleaned, the median falls further to 17,382 and the dispersion effectively vanishes (SD = 1): 89 replications use the full sample of 17,382 observations, and the remaining 11 report 17,379. This convergence pattern mirrors that of the point estimates—as constraints increase, analyst discretion over sample construction diminishes, and the resulting samples become nearly identical.

The declining sample sizes across tasks reflect the progressive application of filters to a common underlying population. The modal Task 1 sample of 561,470 observations covers non-citizen, Hispanic, foreign-born individuals aged 16–64 surveyed from 2005 to 2016. Moving to Task 2, the research design restricts this population to birth cohorts 1977–1986 (ages 26–35 as of June 2012) and to individuals who arrived before age 16—a DACA eligibility criterion—reducing the sample to approximately 44,000. The Task 3 pre-cleaned dataset applies four additional filters to this Task 2 sample: dropping survey years 2006–2007, requiring a high school diploma or above, computing a fractional age using birth quarter (which excludes boundary cases), and restricting to the non-institutionalized household population. Together, these filters reduce the sample from 44,000 to 17,382.

Standard Errors. Table 2, Panels C and D report standard error statistics. Standard errors increase mechanically from Task 1 to Task 3 as sample sizes decline. Task 1 standard errors are small (median 0.42 pp) because of the large samples, while Task 3 standard errors are roughly four times larger (median 1.67 pp) due to the fixed sample of approximately 17,400 observations.

Robust standard errors (predominantly HC1) are the modal choice across all three tasks, used in 55–65% of replications. Clustered standard errors are the second most common choice (23–35%), with clustering exclusively by state—a natural choice given that DACA implementation varied across states. Only a single replication (Task 1, replication 94) clusters by household. No replication uses two-way clustering. Conventional (homoskedastic) standard errors are used in 7–12% of replications.

Table 2: Summary Statistics of Claude Code Replications by Task (Percentage Points)

	Task 1	**Task 2**	**Task 3**
Panel A: Point Estimates			
N (replications)	100	100	100
Mean	2.9	4.5	6.1
Median	2.7	4.6	6.1
Std. Dev.	2.0	1.3	0.8
IQR	1.5	0.5	0.8
10th Percentile	0.9	2.0	5.2
25th Percentile	1.9	4.3	5.6
75th Percentile	3.4	4.9	6.4
90th Percentile	5.3	6.0	7.5
Min	−2.1	1.4	2.0
Max	9.2	6.8	7.5
Range	11.3	5.4	5.5
Panel B: Sample Sizes			
Mean	493,036	47,537	17,382
Median	550,898	43,238	17,382
Std. Dev.	173,884	20,471	1
Min	43,238	42,558	17,379
Max	771,888	164,874	17,382
Range	728,650	122,316	3
Panel C: Standard Errors			
Mean	0.46	1.08	1.72
Median	0.42	1.07	1.67
Std. Dev.	0.13	0.21	0.27
Min	0.33	0.40	1.40
Max	1.19	1.60	2.47
Range	0.86	1.20	1.07
Panel D: Standard Error Type			
Conventional	7	12	10
Robust (HC1/HC3)	57	65	55
Clustered by state	35	23	35
Clustered by household	1	0	0

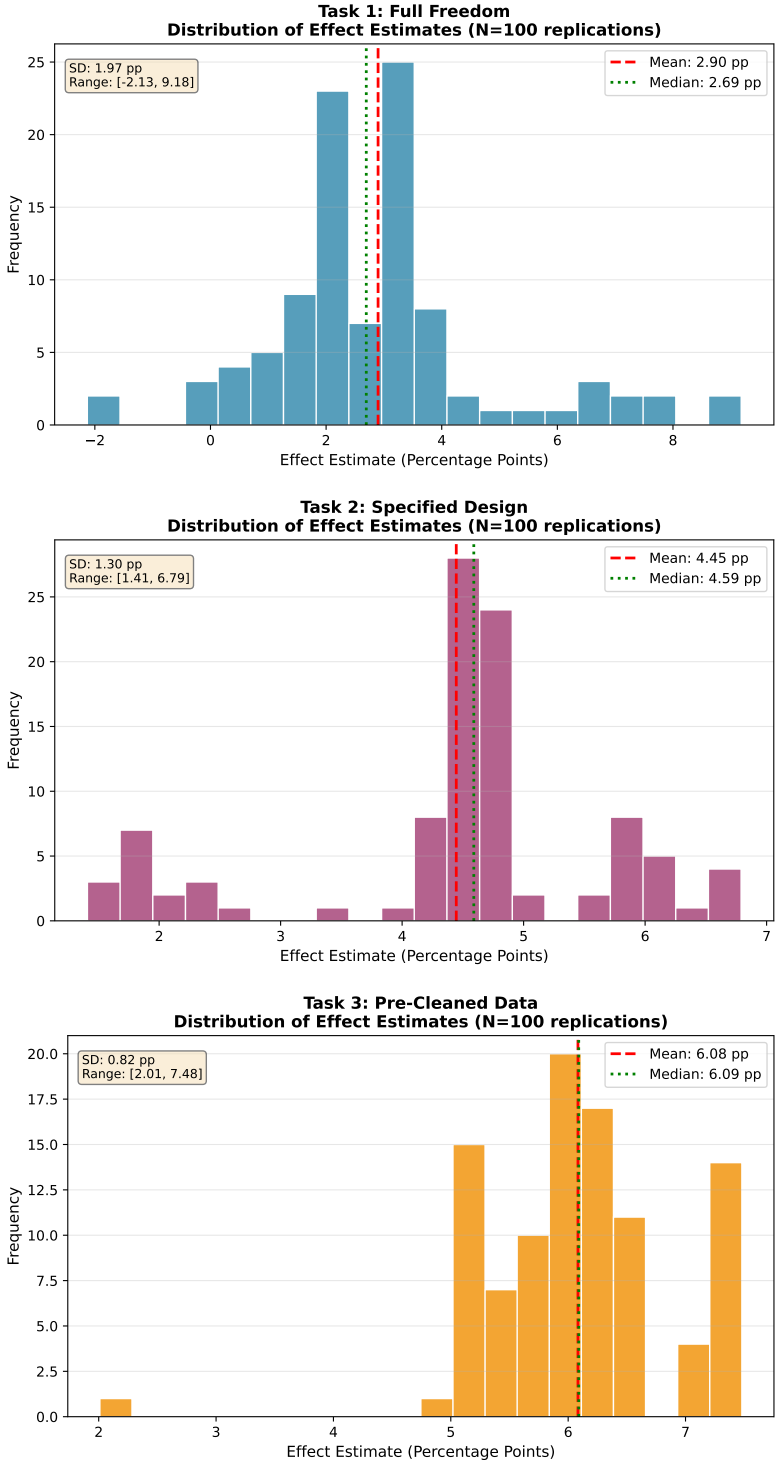

Figure 1: Distribution of Point Estimates by Task. Top: Task 1 (Full Freedom). Middle: Task 2 (Specified Research Design). Bottom: Task 3 (Pre-Cleaned Data).

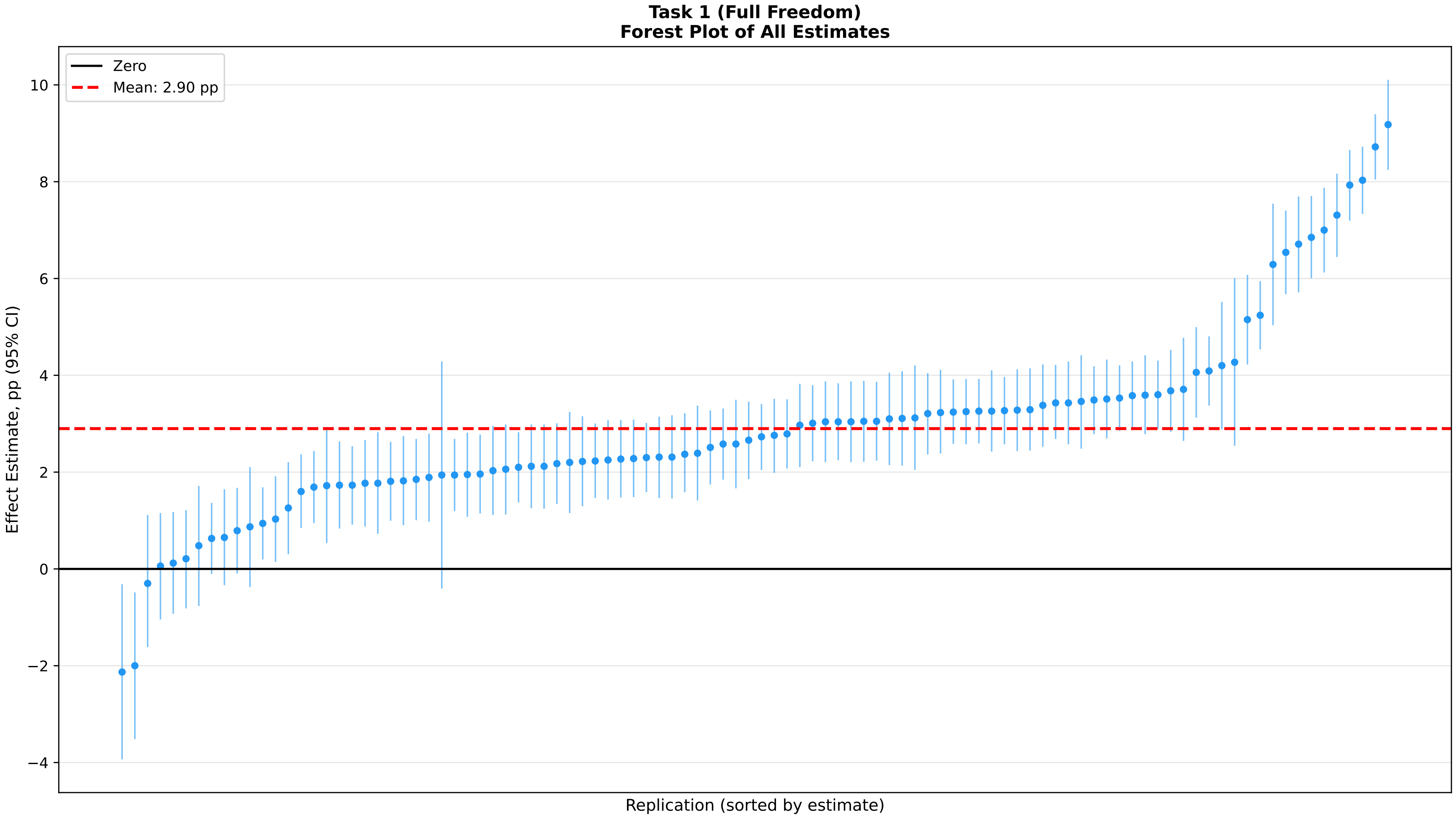

Figure 2: Forest Plot of Point Estimates and 95% Confidence Intervals: Task 1 (Full Freedom)

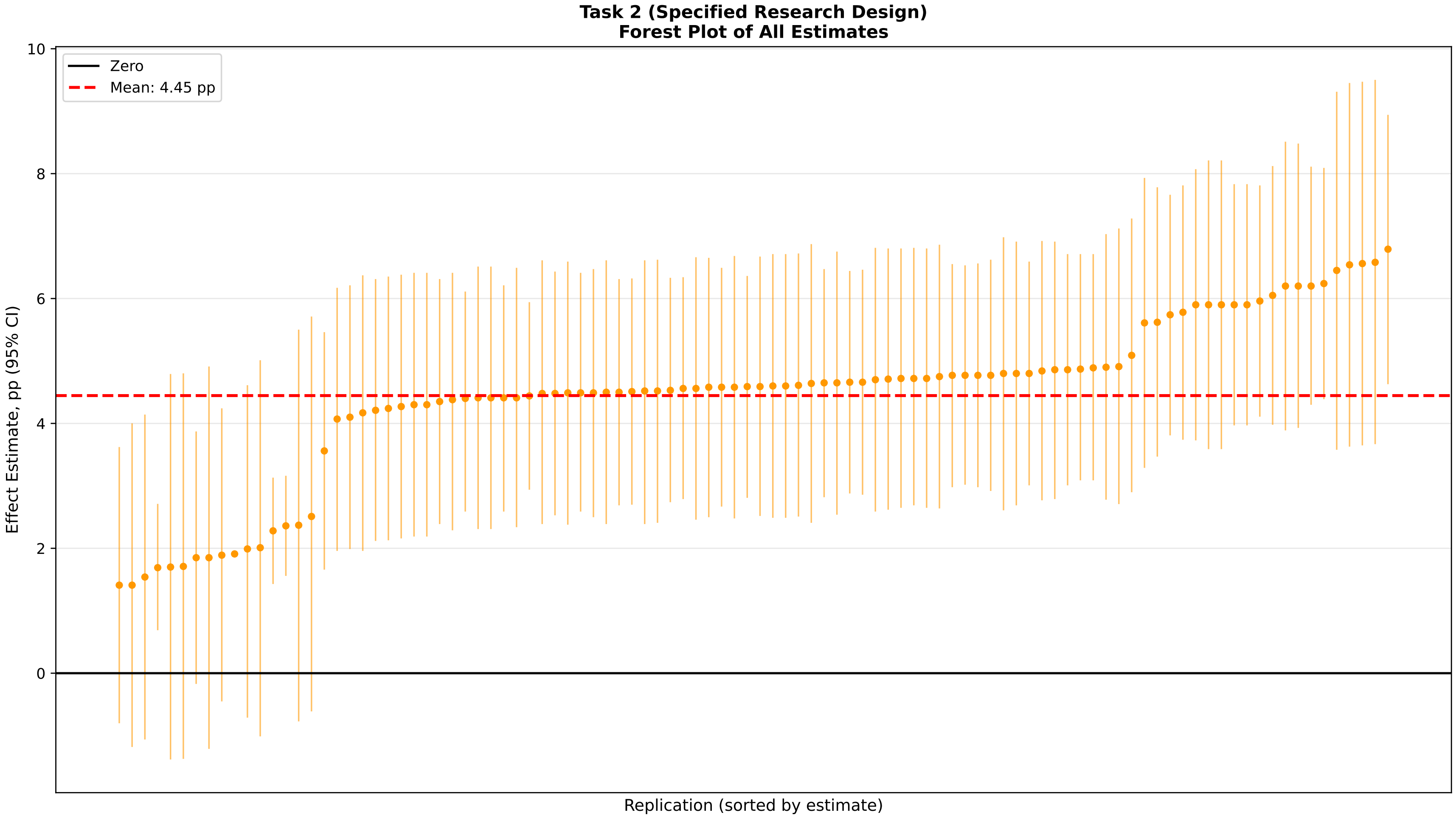

Figure 3: Forest Plot of Point Estimates and 95% Confidence Intervals: Task 2 (Specified Research Design)

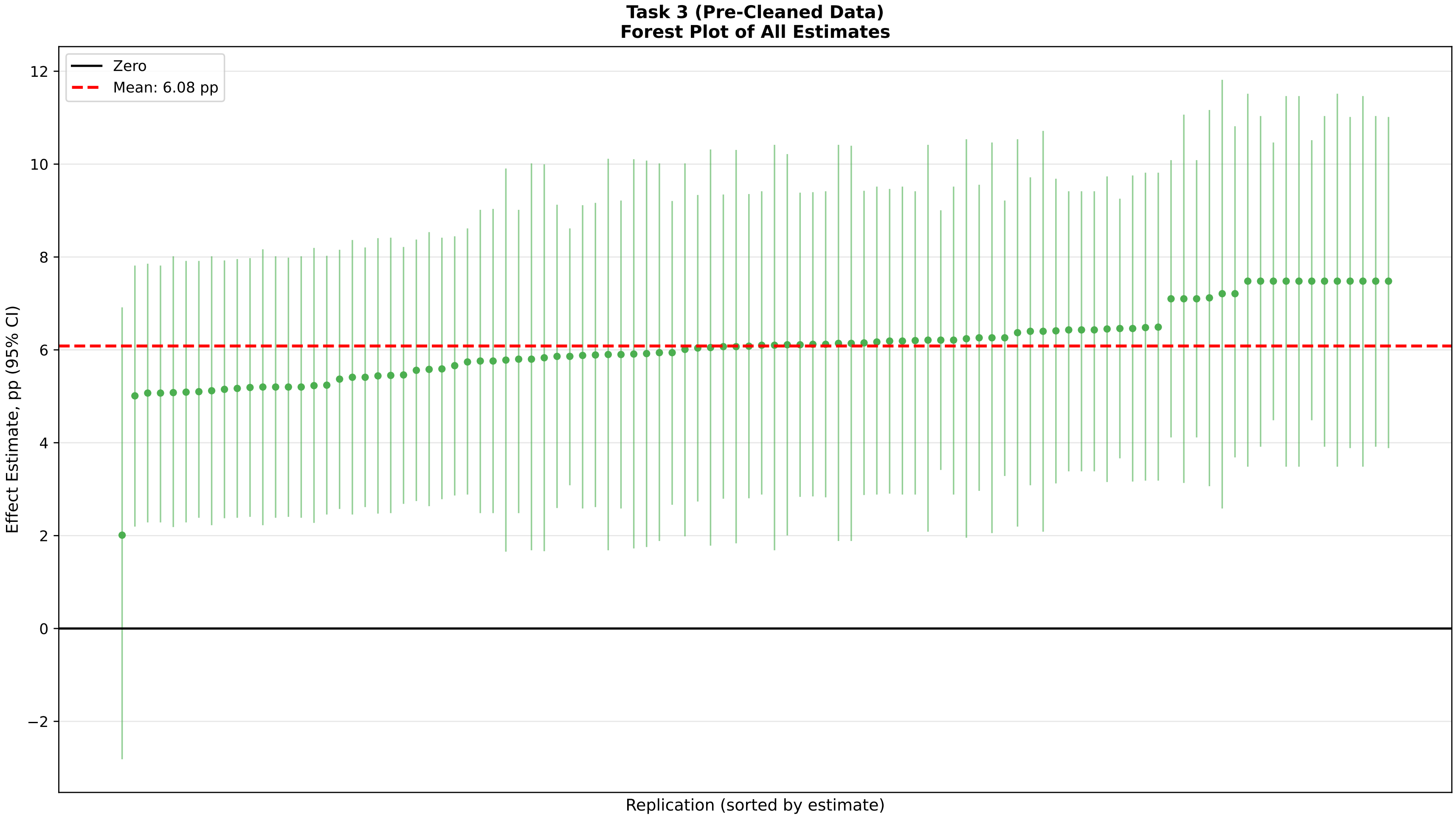

Figure 4: Forest Plot of Point Estimates and 95% Confidence Intervals: Task 3 (Pre-Cleaned Data)

6 Comparison with Human Researchers

This section compares the Claude Code results to those of the 146 human research teams in Huntington-Klein et al. (2025). We report both the weighted and unweighted human statistics for point estimates, where the weighted statistics use inverse-standard-error weights, reducing the influence of noisy estimates; weights are truncated at the 95th percentile to prevent any single researcher from dominating the weighted sample (Huntington-Klein et al., 2025).

Point Estimates. Figure 5 displays the distribution of point estimates for the human weighted, human unweighted, and Claude Code samples using quantile range plots. Table 3 reports the corresponding summary statistics.

Table 3 reports the summary statistics of point estimates for all three groups. In Task 1, Claude Code's median (2.7 pp) nearly matches the human weighted median (2.6 pp) and the human unweighted median (3.0 pp). In Tasks 2 and 3, the medians remain fairly closely aligned: Claude Code produces medians of 4.6 and 6.1 pp, compared to 3.4 and 5.1 pp for the human weighted estimates and 3.2 and 5.0 pp for the human unweighted estimates. Across all tasks and weighting schemes, the median estimates differ by at most 1.4 percentage points, and the weighted mean estimates are within 0.1 pp in Tasks 2 and 3, though the unweighted mean gap reaches 1.6 pp in Task 3. The means tell a broadly similar story. In Task 1, Claude Code's mean (2.9 pp) falls somewhat below both human means (4.4 pp weighted, 5.3 pp unweighted)—a difference driven by the right tail of the human distribution, which pulls the human means well above their medians. In Tasks 2 and 3, the means are closer: Claude Code produces means of 4.5 and 6.1 pp, compared to 4.6 and 6.2 pp for the human weighted, and 4.4 and 4.5 pp for human unweighted.

Table 4, Panel A reports the ratios of human to Claude Code central tendencies; a ratio of 1 indicates identical estimates. The median ratios are close to 1 across all tasks and weighting schemes, ranging from 0.70 to 1.11. In Task 1, the median ratios are closest to unity (0.96 weighted, 1.11 unweighted), indicating that Claude Code and humans arrive at nearly the same median estimate when given full analytical freedom. In Tasks 2 and 3, the ratios fall to 0.70–0.84, reflecting the fact that Claude Code's median is somewhat larger than the human median under the constrained designs. The median ratios also reveal an asymmetry in how Claude Code and humans respond to the prescribed research design. From Task 1 to Task 2, Claude Code's median shifts upward by 70% (from 2.7 to 4.6 pp) and its mean by 55% (from 2.9 to 4.5 pp). The human central estimates respond much less: the human weighted median rises 31% (from 2.6 to 3.4 pp) and the unweighted median rises only 7% (from 3.0 to 3.2 pp); the human unweighted mean actually falls 17% (from 5.3 to 4.4 pp). Huntington-Klein et al. (2025) attribute the limited human response to imperfect adherence: a substantial fraction of human researchers did not fully adopt the prescribed design, leading to a bimodal Task 2 distribution with one mode near the Task 1 estimates and another near the Task 3 estimates. In Task 1, where both groups had full freedom, their medians nearly coincided; the gap in Tasks 2 and 3 thus reflects differential adherence to the instructions rather than a systematic difference in how the two groups approach the problem.

The mean ratios show a different pattern. In Task 1, human means substantially exceed Claude Code's (ratios of 1.52 weighted, 1.83 unweighted), driven by a handful of large human estimates in the right tail that inflate the human mean. In Task 2, the mean ratios converge to near-unity (1.02 weighted, 0.98 unweighted). In Task 3, the weighted mean ratio remains close to 1 (1.02), but the unweighted ratio drops to 0.74, reflecting a few human outliers with large negative estimates that pull the unweighted human mean down to 4.5 pp relative to Claude Code's 6.1 pp. The divergence between weighted and unweighted human statistics across tasks underscores the sensitivity of the human distribution to outliers—a sensitivity largely absent from the Claude Code distribution, whose mean and median are always within 0.2 pp of each other.

While central tendencies agree, the dispersion of point estimates across Claude Code runs is substantially smaller than the variation observed across human teams. Table 4, Panel B reports the ratios of human to Claude Code standard deviations and IQRs, which range from 2.1 to 12.9. The standard deviation ratio increases as constraints increase, from Task 1 to Task 3. Claude Code's standard deviation of point estimates declines monotonically from 2.0 pp in Task 1 to 1.3 pp in Task 2 to 0.8 pp in Task 3, as progressively tighter instructions eliminate degrees of freedom. The human unweighted standard deviation, by contrast, remains roughly constant across tasks (9.5, 10.0, 10.1 pp).[5]

The range of estimates (maximum minus minimum) tells a similar story. Claude Code's range drops from 11.3 pp in Task 1 to 5.4 pp in Task 2 and remains essentially unchanged at 5.5 pp in Task 3. The human unweighted range, by contrast, widens from 70.9 pp in Task 1 to 124.0 pp in Task 2 to 146.0 pp in Task 3, driven by increasingly extreme outliers—the Task 3 minimum is −81.0 pp and the maximum is 65.0 pp. The ratio of the human to Claude Code range rises accordingly, from 6.3 in Task 1 to 17.4–23.0 in Task 2 to 26.5 in Task 3.[6]

[5] Huntington-Klein et al. (2025) attribute the lack of decline from Task 1 to Task 2 to bimodality: some human researchers precisely implemented the newly specified treated-group definition while others did not, with only about 20–25% matching it exactly across all criteria.

[6] Because the human sample (145 teams) is slightly larger than the Claude Code sample (100 runs), one would expect a somewhat wider range for humans even if the underlying distributions were identical. However, this mechanical effect is small relative to the observed range ratios of 6–27.

Table 3: Point Estimates: Human Researchers vs. Claude Code (Percentage Points)

	Task 1	**Task 2**	**Task 3**
Human Weighted (N=138–142)			
Mean	4.4	4.6	6.2
Median	2.6	3.4	5.1
Pctl. 25	1.2	1.8	3.6
Pctl. 75	4.3	5.8	6.0
Min	−4.9	−9.0	−81.0
Max	66.0	85.0	65.0
Std. Dev.	9.2	6.9	10.3
IQR	3.1	4.0	2.4
Range	70.9	94.0	146.0
Human Unweighted (N=145)			
Mean	5.3	4.4	4.5
Median	3.0	3.2	5.0
Pctl. 25	1.4	1.5	3.1
Pctl. 75	5.1	5.8	5.8
Min	−4.9	−39.0	−81.0
Max	66.0	85.0	65.0
Std. Dev.	9.5	10.0	10.1
IQR	3.7	4.3	2.7
Range	70.9	124.0	146.0
Claude Code (N=100)			
Mean	2.9	4.5	6.1
Median	2.7	4.6	6.1
Pctl. 25	1.9	4.3	5.6
Pctl. 75	3.4	4.9	6.4
Min	−2.1	1.4	2.0
Max	9.2	6.8	7.5
Std. Dev.	2.0	1.3	0.8
IQR	1.5	0.5	0.8
Range	11.3	5.4	5.5

Table 4: Ratios: Human Researchers vs. Claude Code

	Task 1	**Task 2**	**Task 3**
Panel A: Mean and Median Ratios			
Mean Estimate Ratio			
Human Weighted / Claude Code	1.52	1.02	1.02
Human Unweighted / Claude Code	1.83	0.98	0.74
Median Estimate Ratio			
Human Weighted / Claude Code	0.96	0.74	0.84
Human Unweighted / Claude Code	1.11	0.70	0.82
Panel B: Dispersion Ratios			
SD Ratio			
Human Weighted / Claude Code	4.6	5.3	12.9
Human Unweighted / Claude Code	4.8	7.7	12.6
IQR Ratio			
Human Weighted / Claude Code	2.1	8.0	3.0
Human Unweighted / Claude Code	2.5	8.6	3.4
Range Ratio			
Human Weighted / Claude Code	6.3	17.4	26.5
Human Unweighted / Claude Code	6.3	23.0	26.5
Panel C: SD / SE Ratios			
SD / Mean SE			
Human Weighted	4.84	2.23	1.75
Human Unweighted	5.00	3.23	1.71
Claude Code	4.35	1.20	0.47
SD / Median SE			
Human Weighted	13.1	4.9	5.7
Human Unweighted	13.6	7.1	5.6
Claude Code	4.76	1.21	0.48

Notes: This table compares 100 independent Claude Code (Opus 4.5) runs to 146 human research teams from Huntington-Klein et al. (2025) across three progressively constrained tasks. Task 1 gives full analytical freedom; Task 2 specifies the research design; Task 3 additionally provides a pre-cleaned dataset. "Human Weighted" uses inverse-standard-error weights following meta-analytic convention; "Human Unweighted" weights all teams equally. Panel A reports the ratio of human to Claude Code mean and median point estimates; a ratio of 1 indicates identical central tendencies. Panel B reports the ratio of human to Claude Code standard deviations (SD), interquartile ranges (IQR), and ranges (maximum minus minimum) of point estimates; ratios above 1 indicate greater dispersion among humans. Panel C reports, for each group, the ratio of the cross-analyst standard deviation of point estimates to the typical reported standard error (SE); values above 1 indicate that analyst-choice variation exceeds sampling variation.

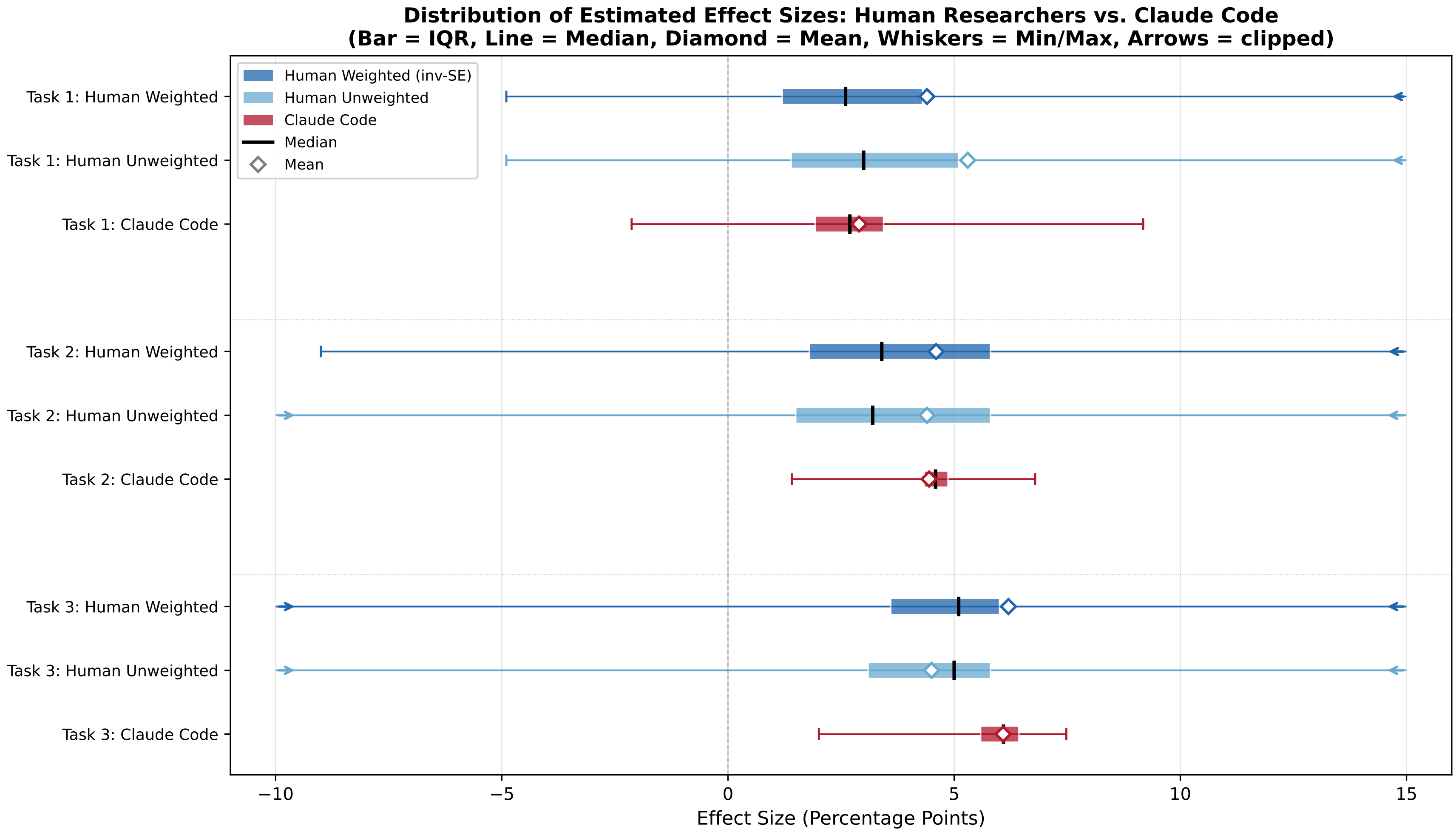

Figure 5: Distribution of Point Estimates: Human Researchers vs. Claude Code. For each task, the plot displays the interquartile range (thick bar), median (vertical line), mean (diamond), and full range (thin line) for human weighted, human unweighted, and Claude Code estimates.

A similar pattern is also visible in the ratio of the standard deviation of point estimates to the standard error, reported in Table 4, Panel C. The standard deviation of point estimates captures variation created by analyst choices whereas the standard error captures sampling variation. Using the mean SE as the denominator, the ratio for Claude Code falls sharply from 4.35 in Task 1 to 1.20 in Task 2 to 0.47 in Task 3; using the median SE, the pattern is similar (4.76, 1.21, 0.48). Thus by Task 3 variation of choices across runs plays a less important role than sampling variation. For human researchers the corresponding ratios decline as well but less sharply, and remain well above 1 even for Task 3. So for humans variation across researchers still plays a more important role than sampling variation in Task 3.

Sample Sizes. Figure 10 in the Appendix compares the distribution of sample sizes between human researchers and Claude Code. Table 5 reports the corresponding summary statistics.

Table 5: Sample Sizes: Human Researchers vs. Claude Code

	Task 1	**Task 2**	**Task 3**
Human Researchers (N=144–145)			
Mean	828,318	157,006	16,904
Median	179,960	25,414	17,382
Pctl. 25	61,600	18,981	17,379
Pctl. 75	356,787	48,125	17,382
Min	681	6,196	7,833
Max	29,536,580	12,609,847	17,832
Std. Dev.	3,056,037	1,065,593	1,756
IQR	295,187	29,144	3
Range	29,535,899	12,603,651	9,999
Claude Code (N=100)			
Mean	493,036	47,537	17,382
Median	550,898	43,238	17,382
Pctl. 25	427,762	43,238	17,382
Pctl. 75	561,470	44,725	17,382
Min	43,238	42,558	17,379
Max	771,888	164,874	17,382
Std. Dev.	173,884	20,471	1
IQR	133,708	1,487	0
Range	728,650	122,316	3

As with point estimates, the dispersion of sample sizes across Claude Code runs is substantially smaller than across human teams (Table 5). Claude Code's median sample size in

Task 1 is substantially higher than the human median, suggesting that Claude Code more consistently retained broad samples with wider age ranges. In Task 2, where the instructions specified a treated and comparison group, Claude Code converges sharply—nearly all replications land on approximately 43,000 to 45,000 observations. In Task 3, both groups converge to the pre-cleaned dataset of 17,382 observations.

The pattern of convergence across tasks again highlights Claude Code's more consistent adherence to the prescribed research design. The human sample size IQR falls from 295,187 in Task 1 to 29,144 in Task 2—a meaningful reduction, but one that still leaves substantial disagreement. Claude Code's IQR falls from 133,708 to 1,487, indicating near-complete convergence once the design was specified.

Standard Errors. Figure 11 in the Appendix compares the distribution of reported standard errors. Table 6 reports the corresponding summary statistics.

Table 6: Standard Errors: Human Researchers vs. Claude Code (Percentage Points)

	Task 1	**Task 2**	**Task 3**
Human Researchers (N=139–144)			
Mean	1.9	3.1	5.9
Median	0.7	1.4	1.8
Pctl. 25	0.5	1.0	1.5
Pctl. 75	1.3	2.0	2.6
Min	0.0	0.1	0.0
Max	46.0	74.4	274.7
Std. Dev.	5.5	7.8	26.8
IQR	0.8	1.0	1.1
Range	46.0	74.3	274.7
Claude Code (N=100)			
Mean	0.46	1.08	1.72
Median	0.42	1.07	1.67
Pctl. 25	0.39	0.97	1.50
Pctl. 75	0.47	1.11	2.03
Min	0.33	0.40	1.40
Max	1.19	1.60	2.47
Std. Dev.	0.13	0.21	0.28
IQR	0.08	0.14	0.53
Range	0.86	1.20	1.07

The dispersion of reported standard errors is again substantially smaller across Claude Code runs than across human teams (Table 6). The human distribution is heavily right-skewed: a small number of human teams report very large standard errors, which pulls the human mean well above the human median in every task. Claude Code's mean and median standard errors, by contrast, are very close to each other, reflecting a more symmetric distribution. Therefore, human median standard errors are much closer to Claude Code's than the means.

The main reason Claude Code's standard errors tend to be smaller is sample size: Claude Code's median sample sizes are larger in Tasks 1 and 2 (Table 5), which mechanically produces smaller standard errors. In Task 3, where both groups use the same pre-cleaned dataset of 17,382 observations, the median standard errors are nearly identical. A secondary factor may be that Claude Code uses clustered standard errors less often (Table 7). Human researchers clustered more aggressively: 55% of human team-task observations used some form of clustering, including 13% with two-way clustering (e.g., by state and year). Claude Code clustered in 23–36% of replications, almost exclusively one-way by state, and never used two-way clustering. More aggressive clustering inflates standard errors.

Table 7: Standard Error Type: Human Researchers vs. Claude Code

	Human (Pooled)	**Claude Task 1**	**Claude Task 2**	**Claude Task 3**
Cluster (State)	27%	35%	23%	35%
Cluster (State & Year)	13%	—	—	—
Cluster (Other)	15%	1%	—	—
Het-Robust (Not Clustered)	17%	57%	65%	55%
Other/Bootstrap	5%	—	—	—
None (Unadjusted)	22%	7%	12%	10%

7 Conclusion

This paper compares 100 independent Claude Code replications to 146 human research teams on the same causal inference task under identical instructions. The central estimates are broadly similar—median point estimates differ by at most 1.4 percentage points across tasks—but Claude Code is far less dispersed, with standard deviations 5 to 13 times smaller than across human teams. Claude Code also responds more completely to prescribed constraints: its dispersion falls monotonically from Task 1 to Task 3, while human dispersion remains roughly constant.

These findings demonstrate that AI systems can execute the full pipeline of empirical research—from data cleaning to estimation to written interpretation—at a level broadly comparable to human economists in central tendency and far more consistent in dispersion.

Claude Code does make coding errors: we identified bugs affecting the preferred point estimate in 3.3% of replications, though additional undetected errors likely exist. This rate must be weighed against the fact that human researchers also make mistakes, and could be reduced in practice through more explicit prompting or automated audit agents. The consistency of AI-generated estimates makes them a promising tool for large-scale replication, robustness analysis, and systematic exploration of how analytical choices affect empirical conclusions.

Several limitations deserve note. The near-absence of extreme estimates, while reducing dispersion, may also mean that AI systems are less likely to explore unconventional specifications that could reveal important features of the data—whether this reflects disciplined adherence to standard practice or a limitation in creative analytical reasoning remains an open question. Our comparison is limited to a single study in causal inference; generalizability to other empirical settings is unknown. Finally, the capabilities of frontier AI systems are improving rapidly, and the findings reported here represent a snapshot of one model at one point in time.

References

Jojanneke A Bastiaansen, Yoram K Kunkels, Johan Ormel, Marieke Wichers, and Harriette Riese. Time to get personal? the impact of researchers' choices on the selection of treatment targets using the experience sampling methodology. *Journal of Psychosomatic Research*, 137:110211, 2020.

Rotem Botvinik-Nezer, Felix Holzmeister, Colin F Camerer, Anna Dreber, Juergen Huber, Magnus Johannesson, Michael Kirchler, Roni Iwanir, Jeanette A Mumford, R Alison Adcock, et al. Variability in the analysis of a single neuroimaging dataset by many teams. *Nature*, 582(7810):84–88, 2020.

Nate Breznau, Eike Mark Rinke, Alexander Wuttke, Hung H T Nguyen, Muna Adem, Jule Adriaans, Amalia Alvarez-Benjumea, Henrik K Andersen, Daniel Auer, Flavio Azevedo, et al. 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, 2022.

Erik Brynjolfsson, Danielle Li, and Lindsey R Raymond. Generative AI at work. *The Quarterly Journal of Economics*, 140(2):889–942, 2025.

Colin F Camerer, Anna Dreber, Eskil Forsell, Teck-Hua Ho, Jürgen Huber, Magnus Johannesson, Michael Kirchler, Johan Almenberg, Adam Altmejd, Taizan Chan, et al. Evaluating replicability of laboratory experiments in economics. *Science*, 351(6280):1433–1436, 2016.

Fabrizio Dell'Acqua, Edward McFowland, Ethan R Mollick, Hila Lifshitz-Assaf, Katherine Kellogg, Saran Rajendran, Lisa Krayer, Francois Candelon, and Karim R Lakhani. Navigating the jagged technological frontier: Field experimental evidence of the effects of AI on knowledge worker productivity and quality. Working Paper 24-013, Harvard Business School, 2023.

Miguel Faria-e Castro and Fernando Leibovici. Artificial intelligence and inflation forecasts. *Federal Reserve Bank of St. Louis Review*, 106(4):1–14, 2024.

Ethan Goh, Robert Gallo, Jason Hom, Eric Strong, Yingjie Weng, Hannah Kerman, Josephine A Cool, Zahir Kanjee, Andrew S Parsons, Neera Ahuja, Eric Horvitz, Daniel Yang, Arnold Milstein, Andrew P J Olson, Adam Rodman, and Jonathan H Chen. Large language model influence on diagnostic reasoning: A randomized clinical trial. *JAMA Network Open*, 7(10):e2440969, 2024.

Elliot Gould, Hannah S Fraser, Timothy H Parker, Shinichi Nakagawa, Simon C Griffith, Peter A Vesk, Fiona Fidler, David G Hamilton, Robin N Abbey-Lee, Jessica K Abbott, et al. Same data, different analysts: Variation in effect sizes due to analytical decisions in ecology and evolutionary biology. *EcoEvoRxiv*, 2023.

Danny Halawi, Fred Zhang, Chen Yueh-Han, and Jacob Steinhardt. Approaching human-level forecasting with language models. In *Advances in Neural Information Processing Systems*, 2024.

Felix Holzmeister, Magnus Johannesson, Robert Bohm, Anna Dreber, Juergen Huber, and Michael Kirchler. Heterogeneity in effect size estimates. *Proceedings of the National Academy of Sciences*, 121(32):e2403490121, 2024.

Suzanne Hoogeveen, Alexandra Sarafoglou, Balazs Aczel, Yonathan Aday, Ahmad Adber, Alaa Almolla, et al. A many-analysts approach to the relation between religiosity and well-being. *Religion, Brain & Behavior*, 13(3):237–283, 2023.

Nick Huntington-Klein, Claus C Pörtner, Ian McCarthy, and The Many Economists Collaborative on Researcher Variation. The sources of researcher variation in economics. Working Paper 33729, National Bureau of Economic Research, May 2025.

Daniel Martin Katz, Michael James Bommarito, Shang Gao, and Pablo Arredondo. GPT-4 passes the bar exam. *Philosophical Transactions of the Royal Society A*, 382(2270): 20230254, 2024.

Tiffany H Kung, Morgan Cheatham, Arielle Medenilla, Czarina Sillos, Lorie De Leon, Camille Elepano, Maria Madriaga, Rimel Aggabao, Giezel Diaz-Candido, James Maningo, and Victor Tseng. Performance of ChatGPT on USMLE: Potential for AI-assisted medical education using large language models. *PLOS Digital Health*, 2(2):e0000198, 2023.

Edward E Leamer. Let's take the con out of econometrics. *American Economic Review*, 73 (1):31–43, 1983.

Albert J Menkveld, Anna Dreber, Felix Duchene, Juergen Ber, Richard D F Harris, Erik Hjalmarsson, Gur Huberman, Gbenga Ibikunle, Georg von Krogh, et al. Nonstandard errors. *Journal of Finance*, 79(3):2339–2390, 2024.

Shakked Noy and Whitney Zhang. Experimental evidence on the productivity effects of generative artificial intelligence. *Science*, 381(6654):187–192, 2023.

Open Science Collaboration. Estimating the reproducibility of psychological science. *Science*, 349(6251):aac4716, 2015.

Anna Ostropolets, Liantao Zhang, Rupa Makadia, George Hripcsak, and Patrick B Ryan. Reproducible variability: Assessing investigator discordance across 9 research teams attempting to reproduce the same observational study. *Journal of the American Medical Informatics Association*, 30(5):859–868, 2023.

Steven Ruggles, Sarah Flood, Matthew Sobek, Daniel Backman, Annie Chen, Grace Cooper, Stephanie Richards, Renae Rodgers, and Megan Schouweiler. IPUMS USA: Version 15.0 [dataset]. Minneapolis, MN: IPUMS, 2024. https://doi.org/10.18128/D010.V15.0.

Martin Schweinsberg, Michael Feldman, Nicola Staber, Olmo R van den Akker, Robbie C M van Aert, Marcel A L M van Assen, Yang Liu, et al. Same data, different conclusions: Radical dispersion in empirical results when independent analysts operationalize and test the same hypothesis. *Organizational Behavior and Human Decision Processes*, 165:228–249, 2021.

Raphael Silberzahn, Eric L Uhlmann, Daniel P Martin, Pasquale Anselmi, Frederik Aust, Eli Awtrey, Štěpán Bahník, Feng Bai, Colin Bannard, Evelina Bonnier, et al. 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, 2018.

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

Urban Institute. State immigration policy resource. https://www.urban.org/data-tools/state-immigration-policy-resource, 2022.

A Illustrative Examples of Claude Code Output

Each Claude Code instance autonomously produces a complete research report—typically 15–25 pages—along with the underlying Python code and a run log documenting its reasoning process. All 300 reports, code files, and run logs are available at claude-code-economist.com. This appendix provides representative excerpts to give the reader an idea of what Claude Code produces.

Report structure. Table 8 displays the table of contents from one Task 2 replication (Replication 5). The report follows the standard structure of an applied economics paper: introduction, background, data, empirical methodology, results, and discussion. It includes an event study specification, heterogeneity analysis, robustness checks, and appendices with variable definitions, sample construction details, and computational information. This structure is broadly representative—while section titles and the level of detail vary across replications, all 300 reports follow a recognizable empirical economics format.

Abstracts. Figures 6 and 7 reprint the abstracts from two replications to illustrate the style and content of Claude Code's written output. Both abstracts identify the research question, describe the data and research design, report a point estimate with a confidence interval or *p*-value, and note robustness checks. The Task 1 abstract (Replication 5) reports an estimate of 3.1 percentage points and honestly flags that "pre-period parallel trends are partially violated." The Task 3 abstract (Replication 5) reports an estimate of 5.2–6.4 percentage points—closer to the consensus across Task 3 replications—and notes support from placebo tests.

Abstract (Task 1, Replication 5). This study examines the causal effect of eligibility for the Deferred Action for Childhood Arrivals (DACA) program on full-time employment among Hispanic-Mexican, Mexican-born non-citizens in the United States. Using American Community Survey data from 2006–2016 and a difference-in-differences research design, I estimate that DACA eligibility increased the probability of full-time employment by approximately 3.1 percentage points (95% CI: 2.3–3.9 pp, $p < 0.001$). This effect represents about a 7% increase relative to the pre-DACA baseline employment rate among eligible individuals. The results are robust to alternative sample restrictions, eligibility definitions, and model specifications. Event study analysis provides suggestive evidence that the employment gains materialized gradually following DACA implementation, though pre-period parallel trends are partially violated.

Figure 6: Abstract from Task 1, Replication 5

Table 8: Table of Contents: Task 2, Replication 5

1	Introduction
2	Background
2.1	The DACA Program
2.2	Theoretical Mechanisms
2.3	Related Literature
3	Data
3.1	Data Source
3.2	Sample Construction
3.3	Variable Definitions
3.4	Descriptive Statistics
4	Empirical Methodology
4.1	Difference-in-Differences Design
4.2	Extended Specifications
4.3	Event Study Specification
4.4	Standard Errors
4.5	Weighting
5	Results
5.1	Main Results
5.2	Event Study Results
5.3	Trends by Group
5.4	DiD Visualization
5.5	Heterogeneity Analysis
5.6	Robustness Checks
6	Discussion and Conclusion
	Appendix A: Data Dictionary
	Appendix B: Sample Selection Criteria
	Appendix C: Additional Regression Output
	Appendix D: Computational Details

Abstract (Task 3, Replication 5). This replication study investigates the causal effect of eligibility for the Deferred Action for Childhood Arrivals (DACA) program on full-time employment among ethnically Hispanic, Mexican-born individuals residing in the United States. Using data from the American Community Survey (2008–2016, excluding 2012) and a difference-in-differences research design, we compare individuals aged 26–30 (DACA-eligible) to those aged 31–35 (DACA-ineligible due to age cutoff) at the time of the policy's implementation in June 2012. Our preferred specification estimates that DACA eligibility increased full-time employment by approximately 5.2–6.4 percentage points ($p < 0.001$), representing a meaningful effect on labor market outcomes. Robustness checks including event studies, placebo tests, and alternative specifications support the validity of these findings.

Figure 7: Abstract from Task 3, Replication 5

Code. Figure 8 shows an excerpt from the Python analysis script of Task 2, Replication 5. The code illustrates a typical progression: a basic difference-in-differences OLS specification, followed by weighted least squares with survey weights, then the addition of covariates, year fixed effects, state fixed effects, and finally state-clustered standard errors. All 300 replications use Python with `statsmodels`; the specific covariates, fixed effects, and clustering choices vary across runs.

```python
# Model 1: Basic DiD (no controls, no weights)
model1 = smf.ols('fulltime ~ treated + post + treated_post',
                  data=df).fit()

# Model 2: DiD with survey weights (WLS)
model2 = smf.wls('fulltime ~ treated + post + treated_post',
                  data=df, weights=df['PERWT']).fit()

# Model 3: DiD with covariates and survey weights
model3 = smf.wls('fulltime ~ treated + post + treated_post'
                  ' + female + married + educ_hs + age_current',
                  data=df, weights=df['PERWT']).fit()

# Model with state-clustered standard errors
model_robust = smf.wls('fulltime ~ treated + C(year_str)'
    ' + C(state_str) + treated_post + female + married'
    ' + educ_hs + age_current',
    data=df, weights=df['PERWT']).fit(
    cov_type='cluster', cov_kwds={'groups': df['STATEFIP']})
```

Figure 8: Regression code excerpt from Task 2, Replication 5 (abridged)

B Additional Figures

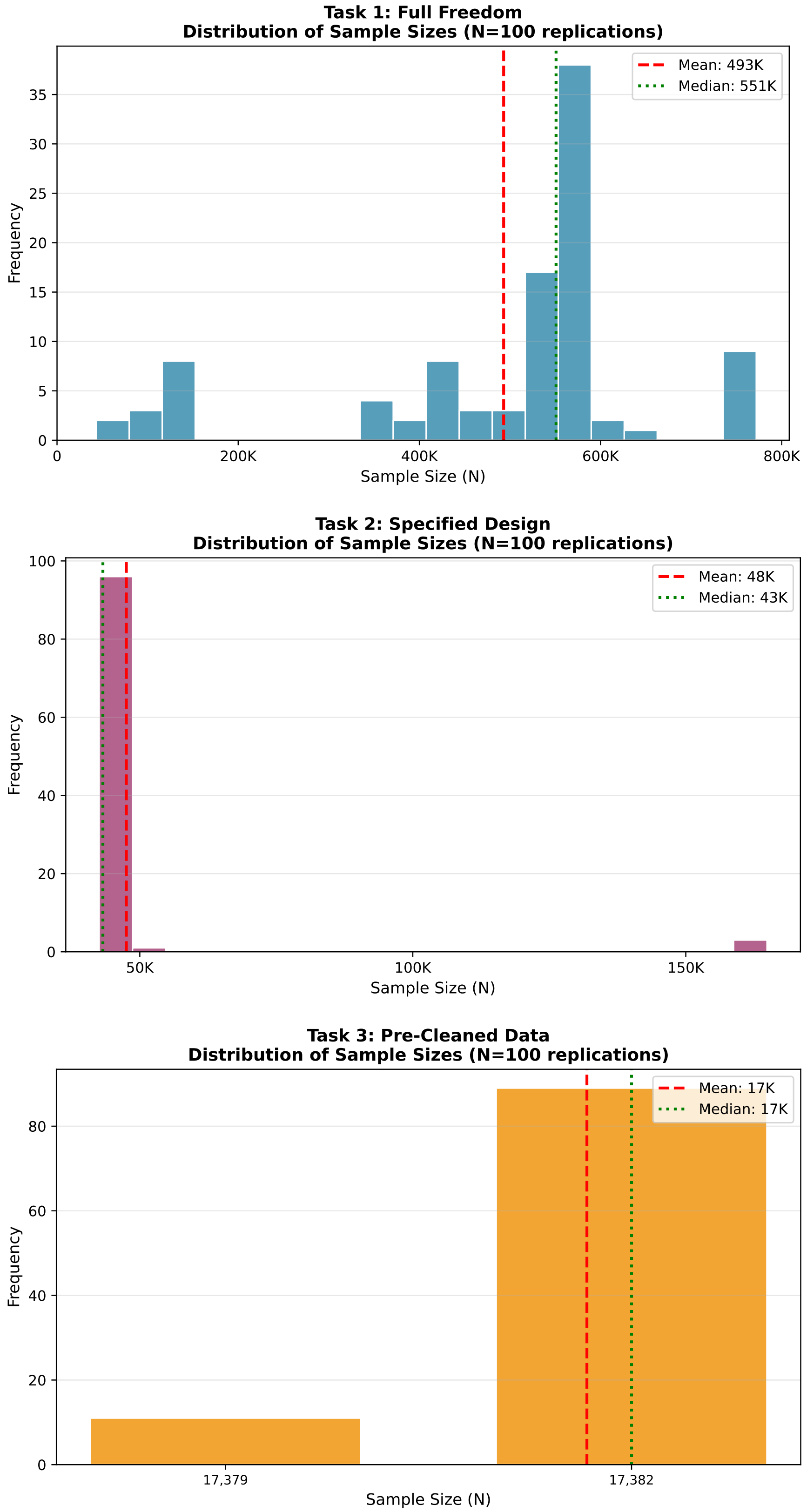

Figure 9: Distribution of Sample Sizes by Task. Top: Task 1 (Full Freedom). Middle: Task 2 (Specified Research Design). Bottom: Task 3 (Pre-Cleaned Data).

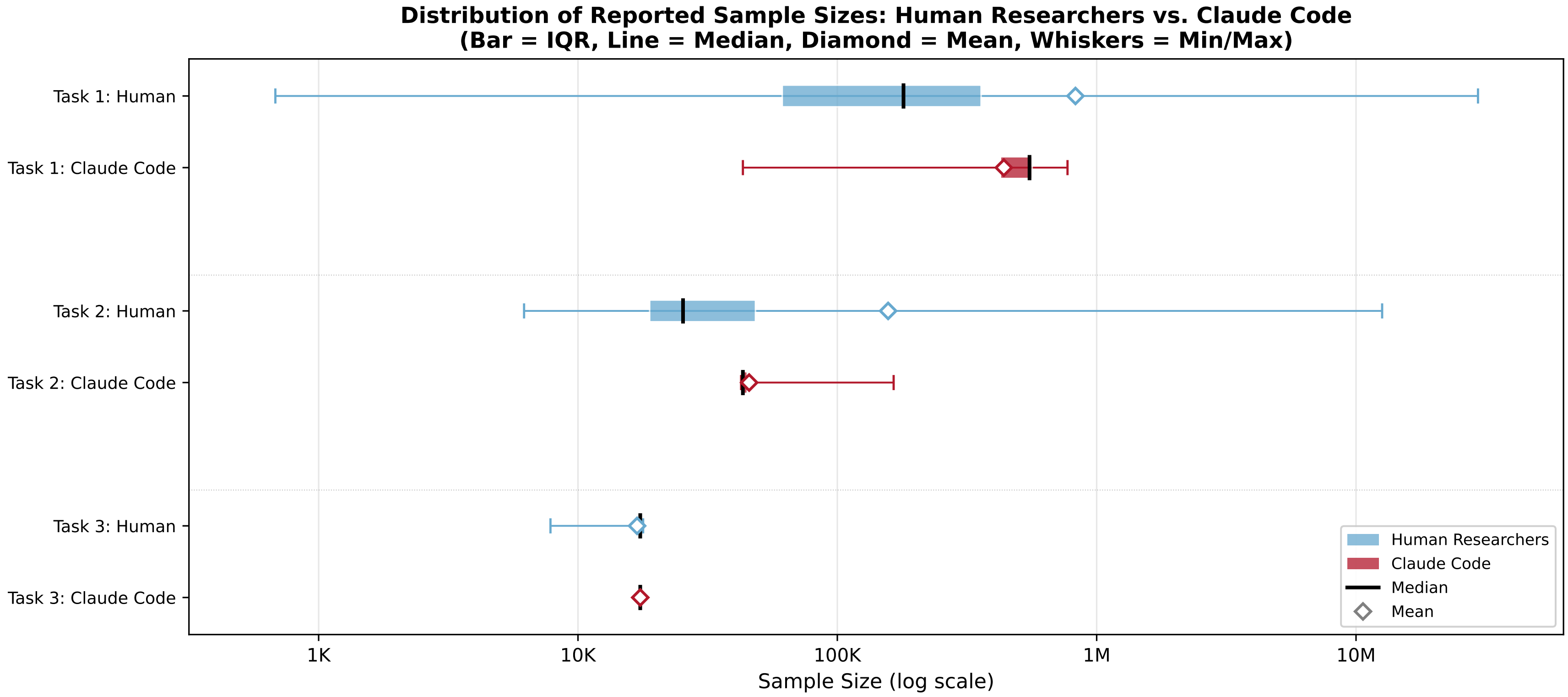

Figure 10: Distribution of Sample Sizes: Human Researchers vs. Claude Code. For each task, the plot displays the interquartile range (thick bar), median (vertical line), mean (diamond), and full range (thin line).

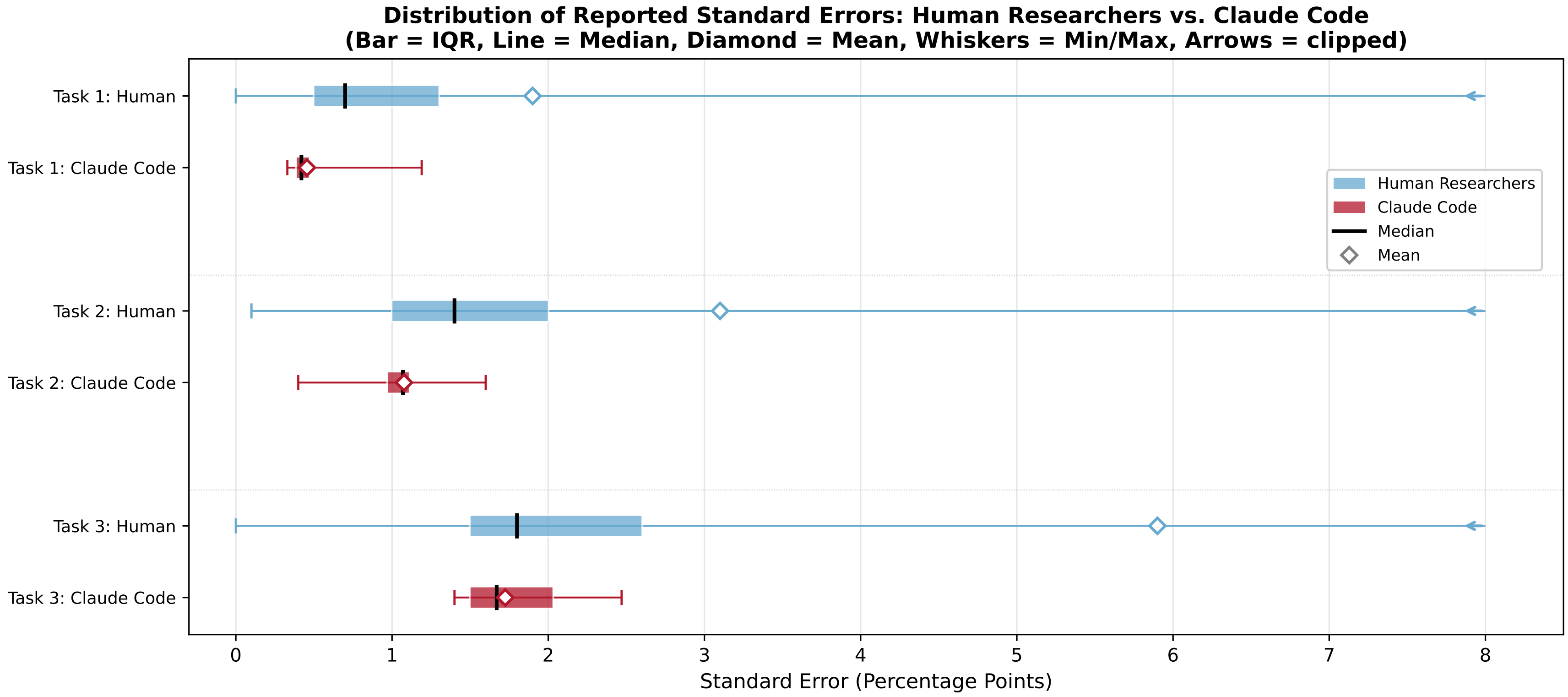

Figure 11: Distribution of Standard Errors: Human Researchers vs. Claude Code. For each task, the plot displays the interquartile range (thick bar), median (vertical line), mean (diamond), and full range (thin line).

C The Many-Analysts Literature

A growing body of evidence demonstrates that empirical research findings depend not only on data and theory but also on the analyst who conducts the work. Even when given identical data and an identical research question, independent researchers routinely arrive at different conclusions. This phenomenon—variously described as "researcher degrees of freedom" (Simmons et al., 2011), "non-standard errors" (Menkveld et al., 2024), or the "garden of forking paths"—arises because the research process requires a cascade of discretionary decisions: how to clean and restrict data, which variables to construct, what model to specify, how to compute standard errors, and how to handle outliers and missing values. Each decision opens a fork in the analytical path, and different researchers navigate these forks differently.

The "many-analysts" paradigm—in which multiple independent teams analyze the same dataset to answer the same question—provides a direct window into this variation. The foundational study in this tradition is Silberzahn et al. (2018), in which 29 research teams analyzed the same soccer dataset to determine whether dark-skinned players receive more red cards. The results ranged from essentially zero to large positive effects, with 69% of teams finding a statistically significant relationship. Subsequent many-analysts studies have extended this approach across disciplines. In political science, Breznau et al. (2022) coordinated 161 researchers organized into 73 teams to examine the relationship between immigration and public support for social policy, finding wide dispersion in both methods and conclusions despite identical data. In finance, Menkveld et al. (2024) had 164 teams test the same hypotheses on the same data, documenting substantial "non-standard errors" attributable to methodological choices—many of which are never reported in published work. Many-analysts designs have also been applied in neuroimaging (Botvinik-Nezer et al., 2020), religion (Hoogeveen et al., 2023), psychology (Bastiaansen et al., 2020; Schweinsberg et al., 2021), ecology and evolutionary biology (Gould et al., 2023), and medical informatics (Ostropolets et al., 2023).

This inter-analyst variation matters because it implies that the uncertainty surrounding any single empirical estimate is substantially larger than the standard error reported in that study. If the standard deviation of estimates across competent analysts exceeds the typical standard error, then analytical choices introduce more variation than sampling noise alone. Holzmeister et al. (2024) formalize this observation, showing that in empirical social science, the variation introduced by researcher choices can outweigh the population variation captured by standard errors. These findings are distinct from—though related to—the broader "replication crisis" in the social sciences (Open Science Collaboration, 2015; Camerer et al., 2016). Traditional replication failures involve new data or new populations; the many-analysts problem concerns variation with *the same* data. Even when the data are fixed and the research question is unambiguous, competent researchers diverge in their answers. As Leamer (1983) warned decades ago, the flexibility inherent in empirical analysis means that “hardly anyone takes data analysis seriously.” The many-analysts literature has made this warning empirically precise.

D Replication Instructions

The instructions below are reproduced from Huntington-Klein et al. (2025). All three task documents were given to both human researchers and Claude Code instances. For the Claude Code experiment, each document was renamed to `replication_instructions.docx` but otherwise left unchanged. The full Task 1 instructions are presented first; for Tasks 2 and 3, only the sections that differ from Task 1 are shown.

Task 1 Instructions (Full Freedom)

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 (treatment) on the probability that the eligible person is employed full-time (outcome), defined as usually working 35 hours per week or more?

DACA was implemented in 2012. Examine the effects on full-time employment in the years 2013–2016.

Background

DACA is a program enacted in the United States on June 15, 2012. The program, enacted by the US federal government, allowed a selected set of undocumented immigrants, who had arrived unlawfully in the US, to apply for and obtain authorization to work legally for two years without fear of deportation. Because the program offers legal work authorization, and also allows recipients to apply for drivers’ licenses or other identification in some states, we might expect that the program would increase employment rates among those eligible.

People were eligible for the program if they:

- Arrived unlawfully in the US before their 16th birthday
- Had not yet had their 31st birthday as of June 15, 2012

- Lived continuously in the US since June 15, 2007
- Were present in the US on June 15, 2012 and did not have lawful status (citizenship or legal residency) at that time

Additional background notes: Applications for the program started to be received on August 15, 2012, and in the first four years nearly 900,000 initial applications were received, about 90% of which were approved. After the initial two years of work authorization and deportation relief, people could reapply for an additional two years, which many did. While the program was not specific to immigrants from any origin country, because of the structure of undocumented immigration to the United States, the great majority of eligible people were from Mexico.

Data

Data for analysis will come from the American Community Survey (ACS) as provided by IPUMS USA, in addition to a provided supplemental file of state demographic and policy information. Please do not retrieve any other data for analysis, or retrieve ACS data from any source other than IPUMS. You are not required to use any of the supplemental state-level information.

In the Select Samples page: Use "USA Samples." Use only the one-year ACS files (these just say "ACS" instead of "ACS 3yr" or "ACS 5yr" or the older census files that say "5% state" and so on). Do not use any files newer than 2016. Do not use any files older than 2006. This is to avoid data definition inconsistencies, and to ensure that the variables necessary for identifying DACA eligibility are all present. You are not required to use all files back to 2006, but do not use any older than that.

On the "Select Variables" page, select Harmonized Variables. DACA eligibility and whether someone was Hispanic and born in Mexico can be determined using: Census year (included in data extract by default); Birth year and quarter (Person → Demographic); Hispanic-Mexican ethnicity, birthplace, citizenship, and year of immigration (Person → Race, Ethnicity, and Nativity).

We cannot distinguish in the data between documented and undocumented non-citizens. Assume that anyone who is not a citizen and who has not received immigration papers is undocumented for DACA purposes. Keep in mind that the ACS does not list the month the data was collected, so observations in 2012 from before and after DACA implementation cannot be distinguished. ACS is a repeated cross-section, not a year-to-year panel data set.

After you click "Create Data Extract" you can click "Select Cases" to limit your sample before you download, so as to reduce the size of the file. If you do this, please keep a record of the selections you make (both what variables you use to limit the sample, and what values you kept) as you will be asked about it after you finish the research task, and it is not easy to come back later and check what you chose. You will be asked later to describe your analytic choices using original IPUMS variable names. This may be easier to do if you refrain from renaming your variables in your code.

Reminders

You may use any statistics package you like. Coding languages are preferred (like Stata, R, Python, Matlab, etc.). Point-and-click statistics packages can be acceptable if they allow your analysis to be automatically replicated from start to finish, with all decisions you've made being fully visible (i.e. your results cannot just be a set of results tables, an Excel sheet with all the analysis already pre-performed so the analysis choices can't be seen, or a set of written instructions for point-and-click software of the form "1. Load the data, 2. In the Analysis menu select Regression…").

If you would typically use graduate students for a given task (data cleaning, coding, etc.) we encourage you to use them for that task in this project as well.

Unless necessary, we ask that you try not to ask for clarification on how the analysis should be done, as the analysis should be independent. Similarly, do not try to guess how other researchers will approach this task in order to match (or avoid matching) their approach. The idea is that we want to see how you would estimate this effect, if you'd had this question, this idea for identification, and had chosen this particular sample.

You may want to review the post-analysis survey form before starting. These are questions you will be asked after you are done, and it may be easier if you prepare to answer them as you work.

There are already published studies that use various methods to look at the effect of DACA or other immigration reforms on different outcomes, including employment. Some of these studies use ACS data as well. You may, if you like, seek out existing literature for background. However, do not assume that these published studies are "the right answer" and attempt to directly copy them just because they are published. This research task is not designed as a replication of any particular study, so there is no "right answer" study to emulate. The idea is that we want to see how you would estimate this effect, if you'd had this question, this idea for identification, and had chosen this particular sample. At most this would be informed by prior research, but not directed by it, as you might be informed by a literature review when writing a paper.

Turn in When Done

For each round of analysis, when you are finished: Make sure that your code and files contain no mention of your own name or the names of any assistants. Move all your code and data to the same folder, and double-check that your code runs properly from a clean session in that folder. Make sure you've held on to the IPUMS `.dat.gz` file provided by the IPUMS website.

Have your participant ID handy. From your preferred estimate, have your effect size, sample size, and standard error/confidence interval handy. Space will be available to submit nonstandard effect estimates as well. Please select a single "preferred estimate" rather than several estimates produced under differing assumptions (robustness tests). What's the estimate you'd mention in the abstract or intro of this paper if you were publishing it? Use that one.

Then, upload: A Word, PDF, or HTML document that contains both a short (1–2 paragraph) description and interpretation of your results, as you might find in the "Results" section of a paper, as well as a demonstration of your results (for example, a table of regression coefficients). Your IPUMS `.dat.gz` extract file. Any code files used. Code files should begin from a clean slate by loading the IPUMS `.dat.gz` or `.dat` file. If there is more than one code file that needs to be run, then there should be a clear indication (such as numbered scripts) as to the order they should be run in.

Finally, you will be asked to fill in: Your participant ID. Your preferred estimate's effect size, sample size, and standard error/confidence interval, among other questions. An explanation of what decisions you made in your analysis, and why you made those decisions.

After everyone is finished, you will hear back with further details on peer review and additional rounds of revision.

Task 2 Instructions (Specified Research Design): Differences from Task 1

The Task 2 instructions are identical to Task 1 except that the following paragraph is inserted into the Research Question section, immediately after the opening question:

> Proceed on this question by assuming that eligible people who were ages 26–30 at the time when the policy went into place comprise the treated group. Estimate the effect of the policy by comparing these individuals to an untreated group made up of people who were ages 31–35 at the time the policy went into place, but otherwise would have been eligible if not for their age. Estimate the effect of

> treatment by seeing how the 26–30 group changed from before treatment to after relative to how the 31–35 group changed (keeping in mind this is not panel data, so it's not actually the same people before and after). You may also, but are not required to, use data to account for differing trends, or use covariates to improve comparability or account for other predictors of full-time employment. Attempt to estimate the effect for all eligible individuals aged 26–30 at the time, and do not limit your estimate, for example, only to one subgroup, like only men or only women. (Note that this further specification of the research question does not imply that other approaches are or were incorrect or inferior. However, at this stage we must all work from the same research design, and so a single design must be chosen).

All other sections (Background, Data, Reminders, Turn in When Done) are unchanged.

Task 3 Instructions (Pre-Cleaned Data + Specified Design): Differences from Task 2

The Task 3 instructions retain the Research Question from Task 2 (including the specified research design) and all other sections unchanged, except that the Data section is replaced with the following:

> Data for analysis will come from the American Community Survey (ACS) as provided by IPUMS USA, in addition to a provided supplemental file of state demographic and policy information. Use the provided data file to perform your analysis. This file includes ACS data from 2008 through 2016, omitting all data from 2012, since it cannot be determined whether someone in 2012 is observed before or after treatment. This entire file constitutes the intended analytic sample for your analysis; do not further limit the sample by dropping individuals on the basis of their characteristics.
>
> The provided data contains a new variable `ELIGIBLE` that is equal to 1 for all observations considered eligible for DACA, and 0 for the comparison group (note that observations from before DACA went into place can also be considered `ELIGIBLE`, although they are not actually treated at the time). Use this variable to identify individuals in the treated and comparison groups, and do not create your own eligibility variable. Individuals who are neither treated nor in the comparison group have been omitted from the data.

There is a variable FT that is equal to 1 for anyone in full-time work, and 0 for anyone not in full-time work. Those not in the labor force are included, usually as 0 values; keep these individuals in your analysis. There is also a variable AFTER that is equal to 1 in the years 2013–2016, and 0 in the years 2008–2011, to indicate years in which DACA was in effect.

The data includes a long list of other variables from ACS and the state-level policy file you previously had access to. Descriptions of each variable can be found on the same IPUMS data selection portal you originally used. Please do not attempt to add in any other information aside from what is in the data, or return to the original ACS files. Also note that the inclusion of a long list of other variables does not imply that you must use these other variables. However, they are available in case anyone does want to use them. Some of these variables have been simplified into variables marked _RECODE, for example EDUC_RECODE takes the original EDUC variable from ACS and simplifies its categories into just "Less than High School," "High School Degree," "Some College," "Two-Year Degree," and "BA+."

Be aware that binary variables that come directly from IPUMS tend to be coded with 1 = No and 2 = Yes. This has been left in place to be consistent with IPUMS documentation. Binary variables added afterwards, including FT, AFTER, ELIGIBLE, and all of the state policy variables, are instead coded with 0 = No and 1 = Yes. Refer to the data documentation for details on coding of each variable. A data dictionary is provided. The code used to generate this file is also provided, and written in the R coding language (with comments describing the code for anyone not familiar with R).

Keep in mind: ACS is a repeated cross-section, not a year-to-year panel data set. You will be asked later to describe your analytic choices using original IPUMS variable names. This may be easier to do if you refrain from renaming your variables in your code.

E Automation Script

The following PowerShell script automated the 100 Claude Code replications for Task 1. The scripts for Tasks 2 and 3 are substantively identical, differing only in directory paths, data files, and the wording of the run prompt.

```
# -------------------------
# User settings
# -------------------------
$N = 100
$MaxParallel = 4   # <-- set to 2, 3, 4, ... to control how many run at once

$SourceDir  = "C:\Users\seraf\DACA Source"   # contains data\ and replication_instructions.pdf
$RootOutDir = "C:\Users\seraf\DACA Results Task 1"

New-Item -ItemType Directory -Force -Path $RootOutDir | Out-Null

# Optional: clean up any old jobs from previous runs
Get-Job | Remove-Job -Force -ErrorAction SilentlyContinue

$jobs = @()

for ($i = 1; $i -le $N; $i++) {

    # Throttle parallel jobs
    while (@(Get-Job -State Running).Count -ge $MaxParallel) {
        Start-Sleep -Seconds 2
    }

    $runId  = "{0:D2}" -f $i
    $repDir = Join-Path $RootOutDir "replication_$runId"
    New-Item -ItemType Directory -Force -Path $repDir | Out-Null

    # Copy required inputs
    Copy-Item (Join-Path $SourceDir "data") -Destination $repDir -Recurse -Force
    Copy-Item (Join-Path $SourceDir "replication_instructions.docx") -Destination $repDir -Force

    # Prompt file
    $promptPath = Join-Path $repDir "RUN_PROMPT.txt"
    $prompt = @"
You are doing an INDEPENDENT replication. Treat this as a clean-room
   run.

1) Read and follow the replication instructions in
   replication_instructions.docx. The data does not need to be
   downloaded. It is in the data folder. The main file is data.csv with
    a data dictionary in acs_data_dict. The file state_demo_policy is
   optional.
2) Produce a ~20-page replication report in LaTeX and turn it into a
   pdf.
3) Log all commands and key decisions in run_log_$runId.md

REQUIRED OUTPUT FILENAMES (do NOT change these):
- replication_report_$runId.tex
- replication_report_$runId.pdf
- run_log_$runId.md

DELIVERABLES MUST EXIST IN THIS FOLDER WHEN FINISHED.

Now begin.
"@

    Set-Content -Path $promptPath -Value $prompt -Encoding UTF8

    # Start replication as a background job
    $jobs += Start-Job -ArgumentList $repDir, $promptPath, $runId -
       ScriptBlock {
        param($repDir, $promptPath, $runId)

        $now = Get-Date

        Get-Process claude -ErrorAction SilentlyContinue |
                Where-Object {
                        $_.StartTime -lt $now.AddHours(-0.75) -and
                        $_.StartTime -gt $now.AddHours(-5)
        } |
        Stop-Process -Force


        $initialDelay = Get-Random -Minimum 0 -Maximum 120
        Start-Sleep -Seconds $initialDelay
        $logPath      = Join-Path $repDir "claude_console_$runId.log"
        $expectedPdf  = Join-Path $repDir "replication_report_$runId.pdf"
        $dataDir      = Join-Path $repDir "data"

        try {
            Set-Location $repDir

            # Run Claude (STDIN prompt), capture stdout+stderr to log file
            Get-Content $promptPath -Raw |
                claude -p --model claude-opus-4-5-20251101 --dangerously-skip-permissions --max-turns 200 2>&1 |
                Out-File -FilePath $logPath -Encoding utf8

            # Only delete data after the replication "completed" (PDF exists)
            if (Test-Path $expectedPdf) {
                Remove-Item $dataDir -Recurse -Force -ErrorAction SilentlyContinue
            }
            else {
                # Keep data for debugging if PDF wasn't produced
                "[$runId] PDF not found at end of run; kept data for troubleshooting." |
                    Out-File -FilePath (Join-Path $repDir "post_run_note_$runId.txt") -Encoding utf8
            }
        }
        catch {
            # Keep data if something fails; log the exception
            "[$runId] ERROR: $($_.Exception.Message)" |
                Out-File -FilePath (Join-Path $repDir "post_run_error_$runId.txt") -Encoding utf8
        }
    }
}

# Wait for all jobs to finish
$jobs | Wait-Job | Out-Null

# Pull job outputs (if any) and clean up
$jobs | Receive-Job | Out-Null
```

```
$jobs | Remove-Job -Force | Out-Null

Write-Host "All replications finished. Check each replication_XX folder for outputs/logs."
```