

A Comparison of Agentic AI Systems and Human Economists

Serafin Grundl*

April 9, 2026

Abstract

This paper compares agentic AI systems and human economists performing the same causal inference tasks. AI systems and humans generally obtain similar median causal effect estimates. While there is substantial dispersion of estimates across model instances, the human distributions of estimates have wider tails. Using AI models as reviewers to compare and rank “submissions,” the following ranking emerges regardless of reviewer model: (1) Codex GPT-5.4, (2) Codex GPT-5.3-Codex, (3) Claude Code Opus 4.6, and (4) Human Researchers. These findings suggest that agentic AI systems will allow us to scale empirical research in economics.

1 Introduction

Agentic AI systems such as Codex and Claude Code can now carry out large parts of the workflow of empirical economics. Starting from a research question, they can translate the problem into a research design, write and revise code, run analyses, inspect results, debug errors, and produce a research report. This raises the possibility that empirical research in

*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, Codex and Gemini CLI ran on my personal computer. Claude Code and Codex were also used extensively for writing, data analysis, and visualization. I thank Nick Huntington-Klein for suggestions that helped to greatly improve the paper. I also 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. Parts of the text were taken verbatim from the previous version [Grundl \(2026\)](#). Replication reports, code files, and comparison reports are available at claude-code-economist.com.

economics could become much more scalable. So far we do not know however how the work of agentic AI systems compares to the work of human economists. This paper provides such a comparison by evaluating agentic AI systems and human economists performing the same causal-inference tasks under the same instructions.

The instructions and the findings from human research teams come from [Huntington-Klein et al. \(2025\)](#). All human research 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 ethnically Hispanic-Mexican Mexican-born people living in the United States, using American Community Survey data. The study’s central innovation is a progressively constrained design across three tasks: In Task 1, teams received the research question with minimal constraints, leaving them free to choose their own sample restrictions, research design, variable definitions, etc. In Task 2, the research design was prescribed: teams were told to compare a treated group of Mexican-born non-citizens aged 26–30 to an untreated group aged 31–35. In Task 3, teams additionally received a pre-cleaned dataset. On these three tasks I compare human economists to three agentic AI systems: Codex with GPT-5.4, Codex with GPT-5.3-Codex, and Claude Code with Opus 4.6.

The first part of this paper compares the distribution of causal effect estimates obtained by human research teams and the AI systems. The means and medians of the preferred treatment-effect estimates obtained by the AI systems are usually fairly close to the human ones. Human means tend to be somewhat higher than AI means, while AI medians tend to be somewhat higher than human medians. There is however one notable exception: Claude Code with Opus 4.6 has substantially smaller median and mean estimates for Task 1 than humans and the two Codex systems. Next, turn to the dispersion of estimates. As LLMs are stochastic there is substantial dispersion of estimates across different instances of the same AI model, including differences in the sign. If we compare the dispersion of human and AI estimates, the ranking depends on the dispersion measure. While the human estimates have larger standard deviations and wider ranges, the AI models sometimes have wider interquartile ranges. In summary, the middle of the human and AI estimate distributions often look similar, but the human distributions have wider tails.

The second part of this paper is an AI review tournament in which “submissions” (codes and write-ups) from humans and the AI models are compared and ranked against each other. The reviewers are the following AI models: Gemini 3.1 Pro Preview, Opus 4.6 and GPT-5.4. For each review the reviewer is asked to write a report comparing four submissions (human, Opus 4.6, GPT-5.3-Codex, GPT-5.4). Each reviewer model writes comparison reports for the same 300 comparison groups. The average rankings are strikingly similar across reviewer models: (1) Codex GPT-5.4, (2) Codex GPT-5.3-Codex, (3) Claude Code Opus 4.6, and

(4) Human Researchers. While I have tried to ensure that the reviews are unbiased and focus on substantive issues (e.g. identification) with the help of a detailed review prompt and a template for the comparison reports I cannot rule out that the AI models are biased against human submissions. It should however be noted that the AI models have almost no bias in favor of their “own” submissions. For instance, Opus 4.6 consistently ranks submissions by GPT-5.4 and GPT-5.3-Codex ahead of its own. Even a cautious interpretation of the review tournament supports the view that the AI submissions are on average not worse than the human submissions.

Taken together, these results suggest that agentic AI systems will allow us to scale research in empirical economics without having to sacrifice quality, but there are at least two important caveats. First, AI systems make mistakes. An earlier draft of this paper used Claude Code with Opus 4.5 and documented two instructive errors (Grundl, 2026). While I have not been able to document such errors in the submissions of the more recent models used in this paper I cannot rule them out. Second, LLM based AI systems are stochastic so different instances of the same model can reach very different conclusions such as treatment effect estimates with opposite signs as documented in this study. Both of these caveats must be weighed against the fact that they apply to human researchers as well. Moreover, by using multiple AI models and/or by using multiple instances of the same model it is easier to detect errors and to explore different research choices that result in different conclusions.

This paper contributes to a growing literature comparing AI and human performance on professional tasks. Recent studies benchmark 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 related literature studies how AI tools affect human productivity in professional settings (Dell’Acqua et al., 2023; Noy and Zhang, 2023; Brynjolfsson et al., 2025). The present paper differs from both strands in two ways. First, it studies an open-ended empirical research task that runs from data handling to estimation to interpretation rather than a narrow exam-style benchmark. Second, it examines AI in two distinct roles within the research process: as a producer of empirical analysis and as a reviewer that evaluates and ranks competing submissions.

To my knowledge the first draft of this paper using Claude Code with Opus 4.5 was the first comparison of human economists and agentic AI systems performing empirical tasks (Grundl, 2026). While this older draft has mostly been superseded by the current version the documentation of errors that AI systems can make remains instructive.¹ More recently, Huang et al. (2026) run GPT-5.2 on the Menkveld et al. (2024) dataset and compare the resulting AI outcome distributions to those of human research teams. They carefully study

¹I have not been able to find similar errors with the newer models used in this paper.

the decisions or forks on the analysis path that drive the dispersion of AI estimates and the differences compared to the human researchers. [Gao and Xiao \(2026\)](#) use Claude Code with Sonnet 4.6 and Opus 4.6 but because they use a different dataset than [Menkveld et al. \(2024\)](#) they cannot compare their findings directly to humans. They also focus on the forks in the analysis path, find that Sonnet 4.6 and Opus 4.6 exhibit stable “empirical styles,” and show that AI peer review leaves dispersion largely unchanged. The most important way in which the present paper differs is the review tournament that attempts to compare the quality of human and AI analysis.

The remainder of the paper proceeds as follows. Section 2 reviews the study by [Huntington-Klein et al. \(2025\)](#). Section 3 describes the AI replication design. Section 4 compares the distributions of AI and human point estimates. Section 5 presents the AI review tournament and the ranking results across reviewer models. The appendix reports additional tables and review materials.

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.” 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.²

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 Setup for Agentic AI Systems

I gave the same three tasks as in Huntington-Klein et al. (2025) to three agentic AI systems: Claude Code with Opus 4.6, Codex with GPT-5.4, and Codex with GPT-5.3-Codex. For each model, I ran 100 independent instances for each of the three tasks for a total of 900 runs.

Prompt. The prompts for Tasks 1 and 2 were as follows.³

You are doing an INDEPENDENT replication.

²Following 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.

³The prompt for Task 3 differed slightly because for Task 3 a cleaned data set was provided.

Treat this as a clean-room run.

You must work ONLY inside this directory.

Do not read or write outside it.

This directory contains:

- replication_instructions.docx
- data/ (input data)

Rules:

1) Follow replication_instructions.docx exactly.

2) data.parquet has 33M+ rows.

Select 30-50 columns and load via

PyArrow iter_batches(),

filtering inside the loop:

```
import pyarrow.parquet as pq
pf = pq.ParquetFile('data/data.parquet')
chunks = []
for batch in pf.iter_batches(columns=SELECTED_COLS, batch_size=500_000):
    chunk = batch.to_pandas()
    chunk = chunk[<your sample filters here>]
    chunks.append(chunk)
df = pd.concat(chunks, ignore_index=True)
```

Do NOT load the full file into pandas at once,

it will exceed memory.

Throughout your script,

keep peak memory low:

delete large intermediate

DataFrames and model objects once they are no longer needed

(del df_temp; import gc; gc.collect()).

The process will be killed if

it exceeds 16 GB.

3) Produce a ~20-page replication report in the style of an academic paper (LaTeX -> PDF).

4) Log all commands and key decisions in run_log_01.md.

5) Save cleaned subsample as data_01.parquet.

6) Main code must be analysis_01.py.

REQUIRED OUTPUT FILENAMES:

- replication_report_01.tex

- replication_report_01.pdf

- run_log_01.md

- data_01.parquet
- analysis_01.py

Begin now.

Discussion. The replication instructions given to the AI systems were identical to those given to the human research teams in [Huntington-Klein et al. \(2025\)](#). The full instructions are reproduced in the Appendix.

One way in which the workflow of AI agents departed from the human researchers' is that they did not download the data for each of the 900 runs. Instead I downloaded a single raw data file, `data.parquet`, with 369 harmonized IPUMS variables and instructed the models to select 30–50 variables. I also instructed the models not to use too much memory to avoid out-of-memory errors with this large data set.

For Tasks 1 and 2, each instance received a `data` folder containing three files: the main dataset, the IPUMS-generated data dictionary, and the supplemental state-by-year file covering demographics and immigration policy variables provided by the [Huntington-Klein et al. \(2025\)](#) study organizers. For Task 3, each instance received the pre-cleaned dataset prepared by the study organizers together with a data documentation file.

Each of the 900 replications was executed as an independent AI instance in a separate working directory. Automation scripts created a fresh directory for each run, copied in the data files, the replication instructions, and a text file with the prompt. Multiple replications were run in parallel with staggered startup times to avoid synchronized computation loads.

Outputs. Replication reports, log files, and code files can be browsed at [claude-code-economist.com](#).

Estimate Extraction. From each replication's outputs, I extract the preferred point estimate, standard error, and sample size with the help of Codex and Claude. Identifying the "preferred" estimate is sometimes genuinely ambiguous rather than a simple extraction problem. In some runs, one specification is marked as preferred in the code or tables, while the abstract or replication report highlights a different estimate. In such cases I prioritized the preferred estimate in the abstract of the replication report.

4 Estimate Comparison

This section compares the three AI models to the 146 human research teams in [Huntington-Klein et al. \(2025\)](#). For the human benchmark, I use the summary statistics reported in Ta-

ble 3 of their paper. As in that paper, “Human Weighted” uses inverse-standard-error weights truncated at the 95th percentile, while “Human Unweighted” weights all teams equally. Appendix Tables 3–5 report the underlying summary statistics, Appendix Table 6 reports sign and significance counts, and Appendix Tables 7 and 8 report the corresponding ratio tables.

Figures 1, 2, and 3 compare the distributions of point estimates. Figure 1 shows box plots, Figure 2 compares means and medians, and Figure 3 compares dispersion measures.

In addition to each of the three individual AI models, the figures also show statistics for the pooled distribution of estimates.

With very few exceptions all submissions by both humans and by the AI models estimate DiD or in some cases triple-difference models. The approaches differ however substantially in their sample construction and thereby in their control groups, whether and how they control for other observables, most importantly age, and in the included fixed effects.

Means and Medians. The means and medians of the AI and human point-estimate distributions are usually fairly close (Figure 2).⁴ Human means tend to be somewhat higher than AI means, while AI medians tend to be somewhat higher than human medians. The clear exception is Opus 4.6 in Task 1, where both the mean and median are only about 0.4 pp—far below humans and the two Codex models.

Dispersion Measures. Different runs of the same AI model can produce materially different estimates—including differences in sign. Opus 4.6 produces negative preferred estimates in 44 of 100 Task 1 runs, and GPT-5.4 does so in 32 of 100 Task 3 runs. GPT-5.3-Codex also generates occasional negative estimates (6 in Task 1, 13 in Task 3). Appendix Table 6 shows that almost all negative statistically significant estimates come from Opus 4.6 in Task 1, whereas GPT-5.4’s negative Task 3 estimates are not statistically significant.

Despite the considerable dispersion of the AI estimates, the human estimates have larger standard deviations and a wider range (Figure 3). The interquartile-range comparison is more mixed: in Task 1 Opus 4.6 has a wider IQR than the human distributions, and in Task 3 both Opus 4.6 and GPT-5.4 do as well. So the middle 50 percent of the AI estimates can sometimes be at least as dispersed as the middle 50 percent of the human estimates. Appendix Figure 5 shows the corresponding histograms by task and model, and Appendix Figures 7–15 show the run-level forest plots. These appendix figures make clear that several model-task cells exhibit visible breaks rather than smooth continua.

⁴The earlier Opus 4.5 comparison reported a Task 1 mean of 2.9 pp and median of 2.7 pp, compared with human means of 4.4 pp (weighted) and 5.3 pp (unweighted) and human medians of 2.6 pp (weighted) and 3.0 pp (unweighted). Across Tasks 1–3, the Opus 4.5 means were 2.9, 4.5, and 6.1 pp, and the medians were 2.7, 4.6, and 6.1 pp.

Effect of Prescribed Research Design and Cleaned Dataset. One surprising finding in [Huntington-Klein et al. \(2025\)](#) is that prescribing the research design in Task 2 and providing a cleaned data set in Task 3 has a surprisingly small effect on the dispersion of estimates for human researchers (Figure 3). Prescribing the research design (Task 1 to Task 2) reduces dispersion for all three AI models: standard deviations fall from 1.0 to 0.8 pp (GPT-5.4), 2.3 to 1.3 pp (GPT-5.3-Codex), and 2.7 to 1.8 pp (Opus 4.6), while ranges contract from 5.7 to 4.8, 14.7 to 6.5, and 11.5 to 5.9 pp, respectively. Additionally providing a cleaned data set (Task 2 to Task 3) does however not lead to a further reduction in estimate dispersion. One reason why the shared data set does not help to reduce dispersion is that the AI models already construct similar data sets once the research design is prescribed. This can be seen in Figure 4, which shows a box plot for the sample sizes; Appendix Figure 6 shows the corresponding histograms by task and model.

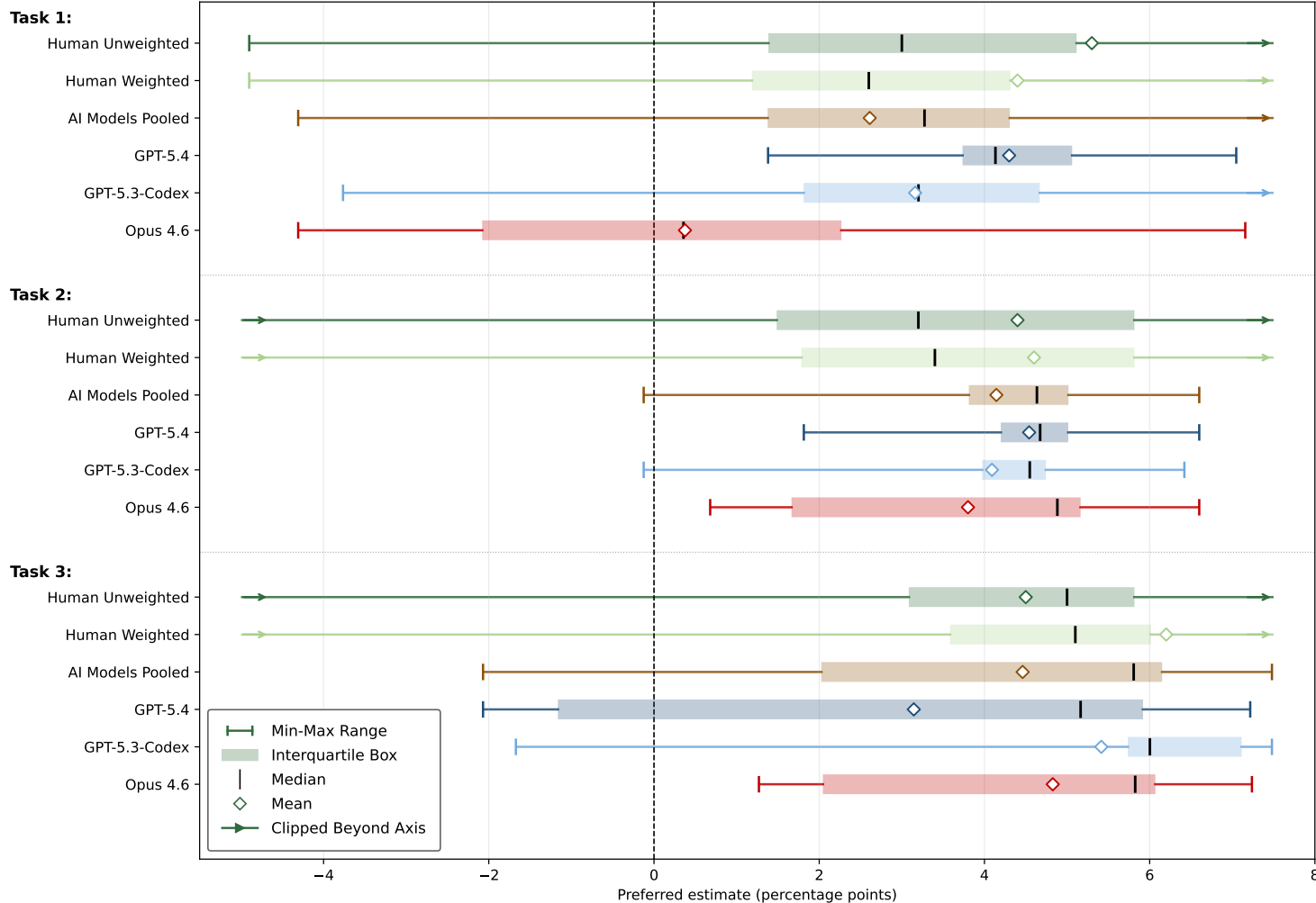


Figure 1: **Distribution of Point Estimates:** Human Researchers vs. AI Models. For each task, the plot displays a box-and-whisker summary with the interquartile range (box), median (vertical marker), mean (diamond), and full range (whiskers with end caps) for human weighted, human unweighted, AI Models Pooled, GPT-5.4, GPT-5.3-Codex, and Opus 4.6 estimates. To keep the central mass visible, the horizontal axis is clipped to $[-5, 7.5]$ percentage points; arrows mark ranges that extend beyond the plotting window.

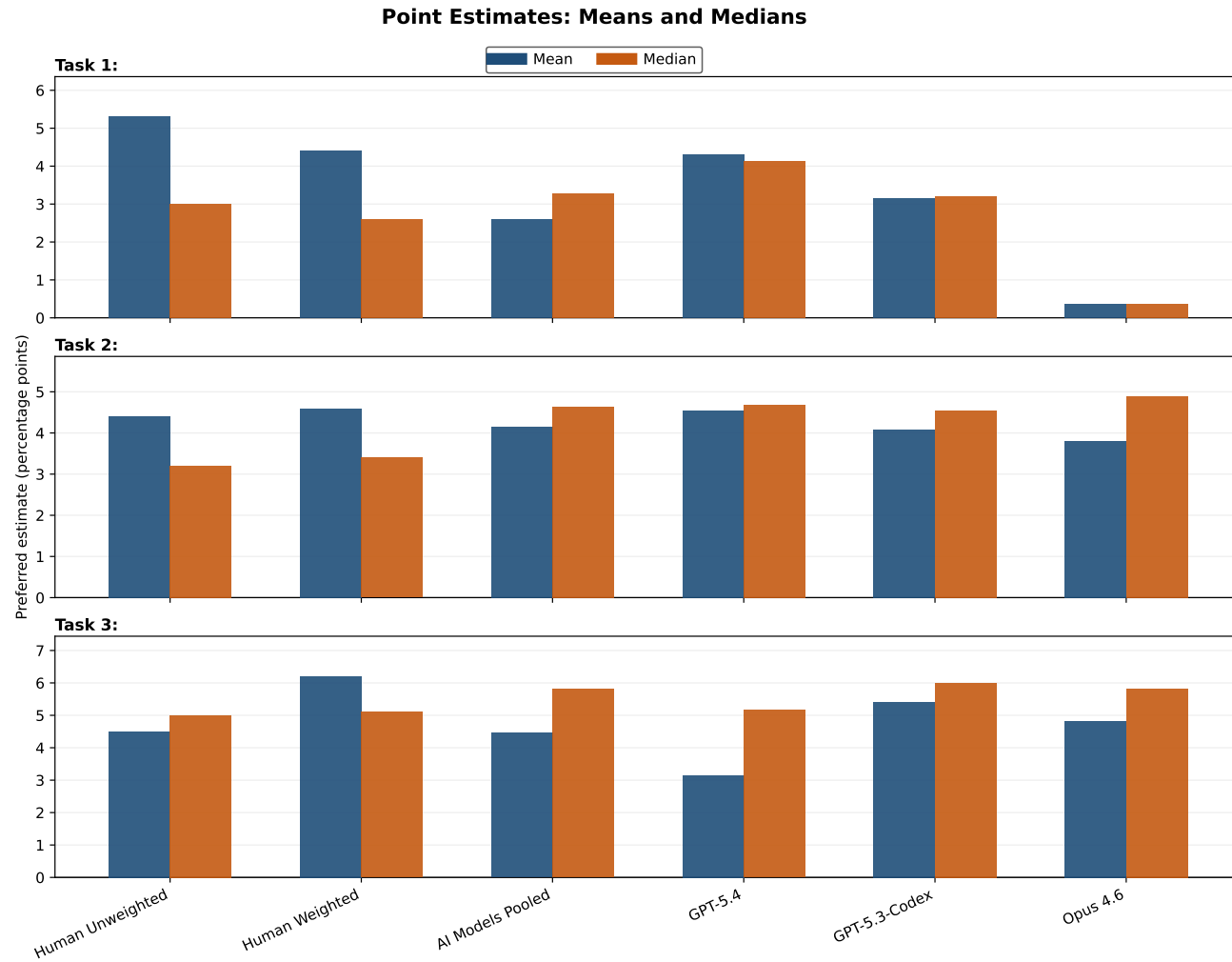


Figure 2: **Point-Estimate Means and Medians:** Human Researchers vs. AI Models. Each panel shows bar charts of the mean and median preferred estimate for each task-group cell on the original percentage-point scale.

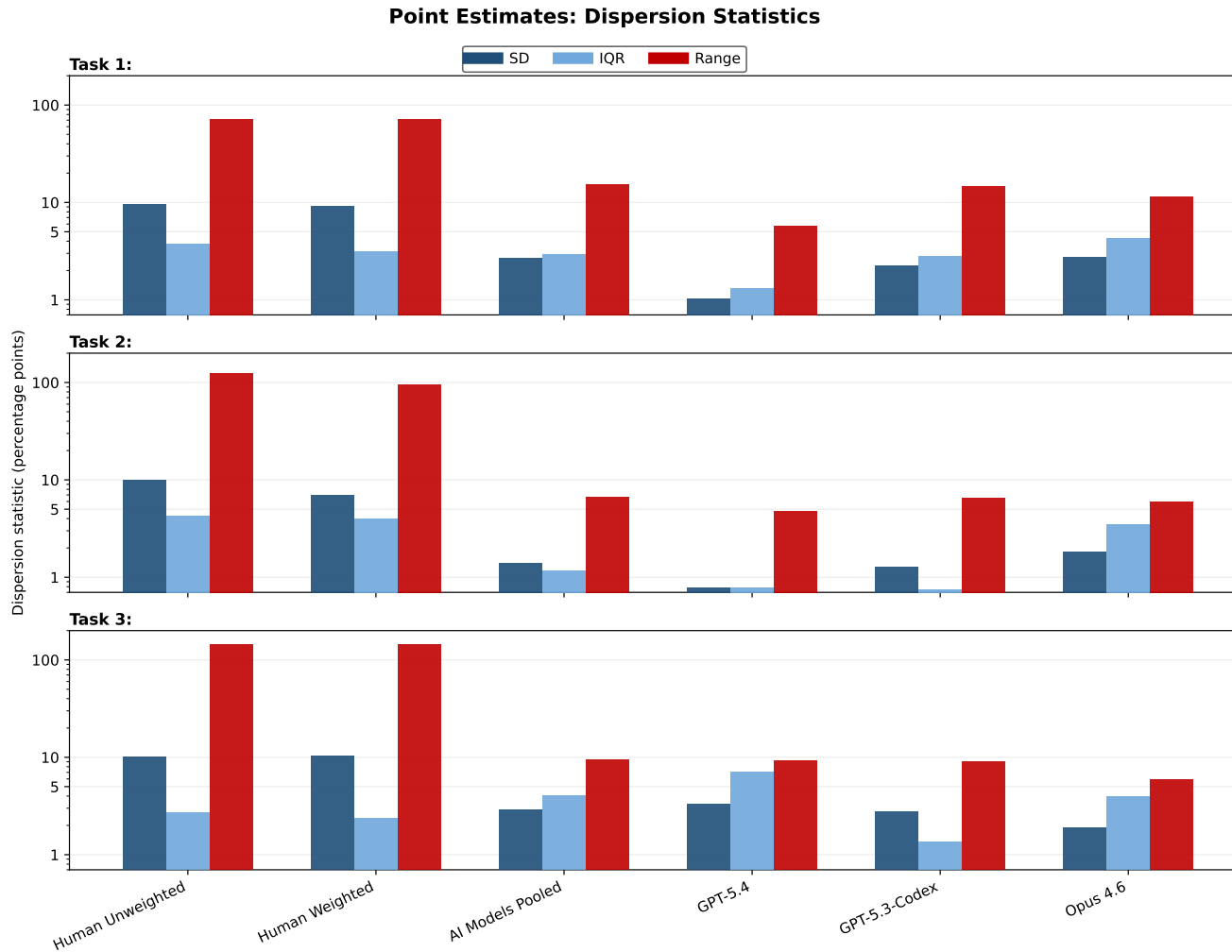


Figure 3: **Point-Estimate Dispersion Measures:** Human Researchers vs. AI Models. Each panel shows bar charts of the standard deviation, interquartile range, and full range of preferred estimates for each task-group cell. The vertical axis is on a log scale and uses the same tick labels in all three panels.

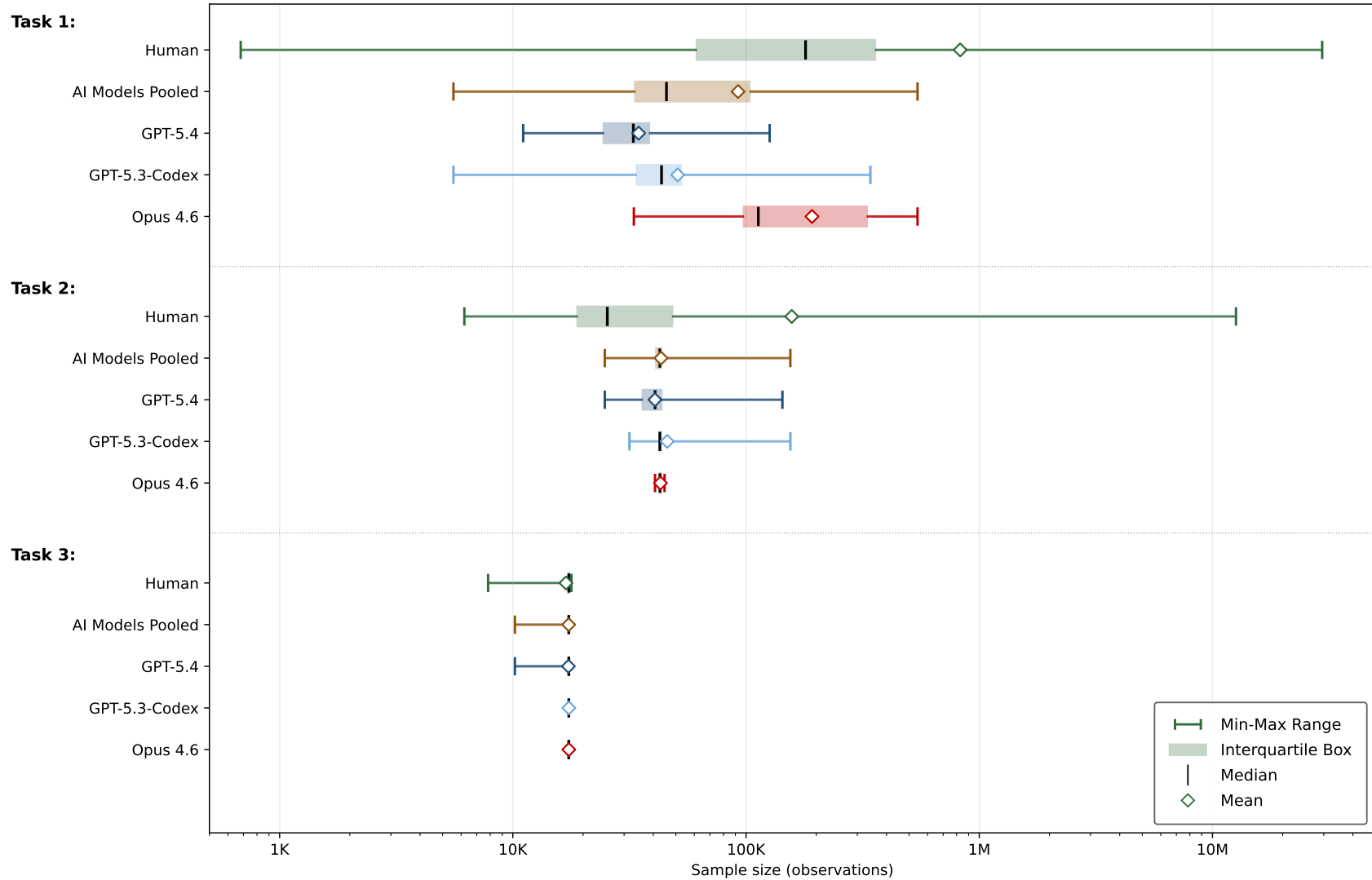


Figure 4: **Distribution of Sample Sizes:** Human Researchers vs. AI Models. The sample-size panels use a log scale on the horizontal axis. For each task, the plot displays a box-and-whisker summary with the interquartile range (box), median (vertical marker), mean (diamond), and full range (whiskers with end caps).

5 Comparison Reviews

The findings in the previous section suggest that the AI models produce estimates with similar central tendency but fewer extreme estimates than human researchers. What we would really like to know, however, is: “How does the quality of the AI analysis compare to the human analysis?” In this section I take a step in that direction using an AI review tournament, in which AI referees compare and rank submissions produced by humans and by the different AI models.

For each task, we want to compare AI submissions and human submissions. To do this I form fixed comparison groups. Each group contains four submissions—one human, one Opus 4.6, one GPT-5.3-Codex, and one GPT-5.4—reviewed side by side on their code and write-ups. This yields 100 comparison groups per task, or 300 in total. Multiple AI reviewer models then score each group: Gemini 3.1 Pro Preview was run twice on all 300 groups, Opus 4.6 and GPT-5.4 each once on all 300 groups, and GPT-5.3-Codex on a 30-group subset. Gemini did not generate any of the candidate submissions and therefore serves as the closest thing to an independent reviewer, but it is of course closely related to the other models.

Since only 100 human submissions are needed per task, some human submissions must be dropped. I first dropped submissions with no code and submissions with sparse write-ups. After this filter, 108 human submissions remained in Task 1, 114 in Task 2, and 112 in Task 3. I then included all 92 participant IDs that survived the filter in every task and filled the remaining eight slots per task with a random draw.

The comparison reports can be browsed at claude-code-economist.com.

5.1 Review Template and Prompt

One concern with using LLMs as reviewers is that they could be biased against human submissions. To focus the reviewers on substantive issues I provided a review template and a detailed review prompt. The review template asks the reviewers to discuss the following items for each of the four submissions in each group:

1. Sample construction
2. Definitions of treatment and control groups
3. Definition of the outcome variable
4. Estimated specification

5. Parallel Trends Assumption
6. No-anticipation and overlap/support requirements
7. Treatment effect vs age
8. Covariates and fixed effects
9. Robustness, assumptions, heterogeneity of effects
10. Standard errors and inference
11. Fatal flaws and other material issues

The template also asks the reviewers to list (1) strengths, (2) risks, (3) minimal fixes needed, and (4) final score for each candidate. The candidate-by-candidate discussion is then followed by a comparison section where the reviewer compares the four submissions.

The review prompt tells the reviewer to weight identification strategy most heavily, to use code evidence for each major claim, and not to reward polished presentation, verbosity, or the sign and significance of the preferred estimate by themselves. The full review prompt is reproduced in Appendix C.

Each review was executed inside a Docker-isolated match directory containing only the task instructions, the review prompt, the review template, and the candidate materials. Submissions were presented to the reviewer as anonymized labels such as Candidate A, Candidate B, Candidate C, and Candidate D. If the reviewer models are prompted to guess the identities of the four candidates, however, they can usually do so based on coding language and based on coding and writing styles.

5.2 Rankings and Scores

Table 1 shows average rankings and scores by reviewer model and submission source if all three tasks are pooled. Appendix Tables 9–11 show the same information by task. Across reviewer models, the rankings are strikingly similar: GPT-5.4 submissions rank first, GPT-5.3-Codex submissions rank second, Opus 4.6 submissions rank third, and human submissions rank fourth. The table also shows that the reviewer models have only a slight bias in favor of their own submissions. For each reviewer model, human scores have the highest standard deviation (Panel C).⁵

⁵I also ran a code-only review profile on the balanced shard-1 subset. In this profile, the reviewer saw only the candidate code and was explicitly instructed not to evaluate the preferred estimate, confidence interval, or sample size fields from non-code artifacts. On this matched shard-1 subset, the code-only results

Table 1: **Average Ranks and Scores:** Reviewer models in rows and submission sources in columns. Panel A reports average ranks, where lower numbers are better. Panel B reports average scores (0–100). Panel C reports the standard deviation of scores. Gemini 3.1 Pro Preview Pass 1, Gemini 3.1 Pro Preview Pass 2, Opus 4.6, and GPT-5.4 each cover all 300 groups; GPT-5.3-Codex only covers 30 groups.

Reviewer Models	Submission Sources			
	GPT-5.4	GPT-5.3-Codex	Opus 4.6	Human
Panel A: Average Rank				
GPT-5.4	1.15	1.95	3.08	3.82
GPT-5.3-Codex	1.27	1.87	3.07	3.80
Opus 4.6	1.14	2.23	2.86	3.76
Gemini 3.1 Pro Preview Pass 1	1.24	2.05	2.77	3.94
Gemini 3.1 Pro Preview Pass 2	1.27	2.02	2.83	3.88
Panel B: Average Score				
GPT-5.4	83.0	75.7	55.6	32.5
GPT-5.3-Codex	83.4	78.2	61.4	40.2
Opus 4.6	77.8	67.6	58.3	38.2
Gemini 3.1 Pro Preview Pass 1	92.3	84.4	72.9	32.4
Gemini 3.1 Pro Preview Pass 2	92.5	84.9	71.4	32.0
Panel C: SD of Score				
GPT-5.4	5.1	6.7	9.9	16.2
GPT-5.3-Codex	3.9	6.4	9.0	14.6
Opus 4.6	4.5	8.6	9.5	14.0
Gemini 3.1 Pro Preview Pass 1	8.1	9.6	11.7	13.4
Gemini 3.1 Pro Preview Pass 2	8.1	9.1	11.3	13.9

Table 2: **Review Outcomes and Point-Estimate Sign:** OLS regressions of review outcomes on an indicator for whether the preferred estimate is positive, with reviewer \times task \times submission-source fixed effects. The sample uses the following reviews: Gemini 3.1 Pro Preview Pass 1, Gemini 3.1 Pro Preview Pass 2, Opus 4.6, and GPT-5.4. Clustered standard errors at the review-group level are shown in parentheses. Lower ranks are better.

	Score	Rank
Positive-estimate indicator	3.11 (0.65)	-0.04 (0.03)

The review prompt explicitly instructs the models not to reward the sign of the preferred estimate by itself. Table 2 shows that, conditional on reviewer \times task \times submission-source fixed effects, positive estimates are nevertheless associated with somewhat higher scores (about 3 points). They are also associated with a lower, i.e. better, rank (Column 2), but this effect is small and not significant.

6 Conclusion

This paper compares agentic AI systems and human economists performing the same causal inference tasks. The central tendency of estimates is usually similar between humans and AI systems. While there is substantial dispersion of estimates across model instances, including in the sign of the estimates, the human distributions of estimates have wider tails. Using AI models as reviewers to compare and rank submissions, the following ranking emerges regardless of reviewer model: (1) Codex GPT-5.4, (2) Codex GPT-5.3-Codex, (3) Claude Code Opus 4.6, and (4) Human Researchers. These findings suggest that agentic AI systems will allow us to scale empirical research in economics.

A Figures and Tables

do not overturn the main ranking. Averaging across the 12 task-reviewer cells in the shard-1 comparison, the code-only average ranks are 1.18 for GPT-5.4, 2.00 for GPT-5.3-Codex, 2.68 for Opus 4.6, and 3.81 for humans. The corresponding code-and-writeup averages on the same matched subset are 1.21, 2.09, 2.84, and 3.86. In other words, the broad ordering is the same under both review profiles. I nevertheless treat the code-and-writeup review as the main specification, because the original submissions are full replication packages and because the write-up contains the preferred estimate, confidence interval, sample size, and stated robustness logic that a referee would ordinarily evaluate.

Table 3: **Point Estimates:** Human Researchers vs. AI Models (Percentage Points)

	Task 1	Task 2	Task 3
<i>Human Unweighted (N=145)</i>			
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
<i>Human Weighted (N=138-142)</i>			
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
<i>AI Models Pooled (N=300)</i>			
Mean	2.6	4.1	4.5
Median	3.3	4.6	5.8
Pctl. 25	1.4	3.8	2.0
Pctl. 75	4.3	5.0	6.1
Min	-4.3	-0.1	-2.1
Max	10.9	6.6	7.5
<i>GPT-5.4 (N=100)</i>			
Mean	4.3	4.5	3.1
Median	4.1	4.7	5.2
Pctl. 25	3.7	4.2	-1.1
Pctl. 75	5.0	5.0	5.9
Min	1.4	1.8	-2.1
Max	7.0	6.6	7.2
<i>GPT-5.3-Codex (N=100)</i>			
Mean	3.2	4.1	5.4
Median	3.2	4.5	6.0
Pctl. 25	1.8	4.0	5.8
Pctl. 75	4.7	4.7	7.1
Min	-3.8	-0.1	-1.7
Max	10.9	6.4	7.5
<i>Opus 4.6 (N=100)</i>			
Mean	0.4	3.8	4.8
Median	0.4	4.9	5.8
Pctl. 25	-2.1	1.7	2.1
Pctl. 75	2.3	5.1	6.1
Min	-4.3	0.7	1.3
Max	7.2	6.6	7.2

Table 4: **Sample Sizes:** Human Researchers vs. AI Models

	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
AI Models Pooled (N=300)			
Mean	92,437	43,181	17,358
Median	45,583	42,689	17,382
Pctl. 25	33,558	41,321	17,382
Pctl. 75	103,188	42,952	17,382
Min	5,563	24,792	10,205
Max	543,595	154,878	17,382
Std. Dev.	112,047	13,164	414
IQR	69,630	1,631	0
Range	538,032	130,086	7,177
GPT-5.4 (N=100)			
Mean	34,641	40,677	17,310
Median	32,864	40,734	17,382
Pctl. 25	24,587	36,145	17,382
Pctl. 75	38,287	43,238	17,382
Min	11,075	24,792	10,205
Max	126,253	143,185	17,382
Std. Dev.	17,037	11,558	718
IQR	13,700	7,093	0
Range	115,178	118,393	7,177
GPT-5.3-Codex (N=100)			
Mean	50,907	45,920	17,382
Median	43,415	42,689	17,382
Pctl. 25	34,052	42,689	17,382
Pctl. 75	52,427	43,219	17,382
Min	5,563	31,622	17,382
Max	341,332	154,878	17,382
Std. Dev.	37,028	19,365	0
IQR	18,375	530	0
Range	335,769	123,256	0
Opus 4.6 (N=100)			
Mean	191,762	42,946	17,382
Median	112,870	42,759	17,382
Pctl. 25	98,105	42,735	17,382
Pctl. 75	328,753	42,952	17,382
Min	33,034	40,707	17,379
Max	543,595	44,725	17,382
Std. Dev.	145,515	906	1
IQR	230,648	217	0
Range	510,561	4,018	3

Table 5: **Standard Errors: Human Researchers vs. AI Models (Percentage Points)**

	Task 1	Task 2	Task 3
<i>Human Researchers (N=139-144)</i>			
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
<i>AI Models Pooled (N=300)</i>			
Mean	1.2	1.0	2.3
Median	1.1	1.0	2.1
Pctl. 25	0.7	0.9	2.0
Pctl. 75	1.5	1.1	2.3
Min	0.3	0.4	1.5
Max	4.0	1.6	4.5
Std. Dev.	0.6	0.2	0.5
IQR	0.8	0.2	0.2
Range	3.7	1.2	3.0
<i>GPT-5.4 (N=100)</i>			
Mean	1.6	1.0	2.5
Median	1.5	0.9	2.2
Pctl. 25	1.3	0.8	2.1
Pctl. 75	1.7	1.1	2.8
Min	0.8	0.6	1.5
Max	4.0	1.5	4.5
Std. Dev.	0.6	0.2	0.6
IQR	0.4	0.3	0.7
Range	3.2	0.9	3.0
<i>GPT-5.3-Codex (N=100)</i>			
Mean	1.2	0.9	2.2
Median	1.1	0.9	2.1
Pctl. 25	1.0	0.8	2.0
Pctl. 75	1.4	1.0	2.1
Min	0.4	0.4	1.8
Max	2.7	1.6	3.5
Std. Dev.	0.4	0.1	0.4
IQR	0.4	0.2	0.1
Range	2.3	1.2	1.7
<i>Opus 4.6 (N=100)</i>			
Mean	0.6	1.1	2.1
Median	0.5	1.1	2.2
Pctl. 25	0.4	1.0	2.0
Pctl. 75	0.7	1.1	2.2
Min	0.3	0.9	1.5
Max	1.6	1.6	2.5
Std. Dev.	0.3	0.2	0.3
IQR	0.3	0.1	0.2
Range	1.3	0.7	1.0

Table 6: **Point Estimate Sign and Significance Counts:** Counts of preferred point estimates by sign and whether the 95% confidence interval excludes zero.

	Negative significant	Negative not significant	Positive not significant	Positive significant
<i>GPT-5.4</i>				
Task 1	0	0	15	85
Task 2	0	0	1	99
Task 3	0	32	15	53
<i>GPT-5.3-Codex</i>				
Task 1	2	4	34	60
Task 2	0	1	4	95
Task 3	0	13	4	83
<i>Opus 4.6</i>				
Task 1	34	10	12	44
Task 2	0	0	39	61
Task 3	0	0	30	70

Table 7: **Central-Tendency Ratios:** Human Researchers vs. AI Models

	Task 1	Task 2	Task 3
Panel: Mean Ratios			
Human Unweighted / AI Models Pooled	2.03	1.06	1.01
Human Unweighted / GPT-5.4	1.23	0.97	1.43
Human Unweighted / GPT-5.3-Codex	1.68	1.08	0.83
Human Unweighted / Opus 4.6	14.14	1.16	0.93
Human Weighted / AI Models Pooled	1.68	1.11	1.39
Human Weighted / GPT-5.4	1.02	1.01	1.97
Human Weighted / GPT-5.3-Codex	1.39	1.12	1.15
Human Weighted / Opus 4.6	11.74	1.21	1.28
Panel: Median Ratios			
Human Unweighted / AI Models Pooled	0.92	0.69	0.86
Human Unweighted / GPT-5.4	0.73	0.68	0.97
Human Unweighted / GPT-5.3-Codex	0.94	0.70	0.83
Human Unweighted / Opus 4.6	8.40	0.66	0.86
Human Weighted / AI Models Pooled	0.79	0.73	0.88
Human Weighted / GPT-5.4	0.63	0.73	0.99
Human Weighted / GPT-5.3-Codex	0.81	0.75	0.85
Human Weighted / Opus 4.6	7.28	0.70	0.88

Notes: Each entry is the human statistic divided by the corresponding AI-model statistic. Values above 1 indicate the human statistic is larger. Negative ratios occur when the AI model’s mean or median estimate is negative, which makes those specific cells harder to interpret than the level comparisons in the main point-estimate table and figures.

Table 8: **Dispersion Ratios:** Human Researchers vs. AI Models

	Task 1	Task 2	Task 3
Panel: SD Ratios			
Human Unweighted / AI Models Pooled	3.54	7.24	3.49
Human Unweighted / GPT-5.4	9.21	12.71	3.04
Human Unweighted / GPT-5.3-Codex	4.22	7.93	3.60
Human Unweighted / Opus 4.6	3.51	5.53	5.30
Human Weighted / AI Models Pooled	3.43	5.00	3.55
Human Weighted / GPT-5.4	8.92	8.77	3.10
Human Weighted / GPT-5.3-Codex	4.09	5.47	3.67
Human Weighted / Opus 4.6	3.40	3.82	5.40
Panel: IQR Ratios			
Human Unweighted / AI Models Pooled	1.28	3.66	0.66
Human Unweighted / GPT-5.4	2.85	5.47	0.38
Human Unweighted / GPT-5.3-Codex	1.31	5.81	2.00
Human Unweighted / Opus 4.6	0.86	1.24	0.68
Human Weighted / AI Models Pooled	1.07	3.40	0.59
Human Weighted / GPT-5.4	2.39	5.09	0.34
Human Weighted / GPT-5.3-Codex	1.10	5.40	1.78
Human Weighted / Opus 4.6	0.72	1.15	0.60
Panel: Range Ratios			
Human Unweighted / AI Models Pooled	4.65	18.44	15.29
Human Unweighted / GPT-5.4	12.51	25.91	15.72
Human Unweighted / GPT-5.3-Codex	4.82	18.95	15.95
Human Unweighted / Opus 4.6	6.18	20.95	24.46
Human Weighted / AI Models Pooled	4.65	13.98	15.29
Human Weighted / GPT-5.4	12.51	19.64	15.72
Human Weighted / GPT-5.3-Codex	4.82	14.36	15.95
Human Weighted / Opus 4.6	6.18	15.88	24.46

Notes: Each entry is the human statistic divided by the corresponding AI-model statistic. Values above 1 indicate the human statistic is larger. Negative ratios occur when the AI model’s mean or median estimate is negative, which makes those specific cells harder to interpret than the level comparisons in the main point-estimate table and figures.

Table 9: **Average Ranks and Scores by Task:** Task 1 only. This table mirrors Table 1 but restricts the aggregation to Task 1 groups. Gemini 3.1 Pro Preview Pass 1, Gemini 3.1 Pro Preview Pass 2, Opus 4.6, and GPT-5.4 each cover all 100 Task 1 groups; GPT-5.3-Codex covers the 10 Task 1 shard-1 groups. Panel A reports average ranks, where lower numbers are better. Panel B reports average scores. Panel C reports the standard deviation of scores.

Reviewer Models	Submission Sources			
	GPT-5.4	GPT-5.3-Codex	Opus 4.6	Human
Panel A: Average Rank				
GPT-5.4	1.12	1.98	3.11	3.79
GPT-5.3-Codex	1.30	1.70	3.40	3.60
Opus 4.6	1.11	2.11	3.13	3.65
Gemini 3.1 Pro Preview Pass 1	1.18	2.09	2.83	3.90
Gemini 3.1 Pro Preview Pass 2	1.17	2.06	2.87	3.90
Panel B: Average Score				
GPT-5.4	84.0	75.2	49.8	28.6
GPT-5.3-Codex	83.2	79.7	52.4	41.1
Opus 4.6	78.3	67.4	50.9	36.6
Gemini 3.1 Pro Preview Pass 1	93.0	83.3	67.2	30.1
Gemini 3.1 Pro Preview Pass 2	93.2	84.2	66.5	29.3
Panel C: SD of Score				
GPT-5.4	6.1	6.2	9.8	16.4
GPT-5.3-Codex	3.9	4.9	7.0	15.6
Opus 4.6	4.1	8.7	7.7	15.8
Gemini 3.1 Pro Preview Pass 1	6.9	9.9	12.5	14.2
Gemini 3.1 Pro Preview Pass 2	7.1	8.0	12.3	14.4

Table 10: **Average Ranks and Scores by Task:** Task 2 only. This table mirrors Table 1 but restricts the aggregation to Task 2 groups. Gemini 3.1 Pro Preview Pass 1, Gemini 3.1 Pro Preview Pass 2, Opus 4.6, and GPT-5.4 each cover all 100 Task 2 groups; GPT-5.3-Codex covers the 10 Task 2 shard-1 groups. Panel A reports average ranks, where lower numbers are better. Panel B reports average scores. Panel C reports the standard deviation of scores.

Reviewer Models	Submission Sources			
	GPT-5.4	GPT-5.3-Codex	Opus 4.6	Human
Panel A: Average Rank				
GPT-5.4	1.14	1.90	3.04	3.92
GPT-5.3-Codex	1.40	1.70	3.00	3.90
Opus 4.6	1.22	2.13	2.78	3.87
Gemini 3.1 Pro Preview Pass 1	1.27	2.00	2.73	4.00
Gemini 3.1 Pro Preview Pass 2	1.25	1.99	2.78	3.98
Panel B: Average Score				
GPT-5.4	84.6	78.5	56.1	27.7
GPT-5.3-Codex	83.2	81.3	64.7	37.3
Opus 4.6	78.6	71.0	61.8	36.9
Gemini 3.1 Pro Preview Pass 1	92.4	85.8	76.1	28.3
Gemini 3.1 Pro Preview Pass 2	92.3	85.6	75.1	28.1
Panel C: SD of Score				
GPT-5.4	3.0	6.6	9.1	12.6
GPT-5.3-Codex	4.7	5.9	4.9	12.0
Opus 4.6	5.0	7.5	9.0	12.5
Gemini 3.1 Pro Preview Pass 1	6.8	9.0	10.0	8.3
Gemini 3.1 Pro Preview Pass 2	6.5	10.7	8.8	9.3

Table 11: **Average Ranks and Scores by Task:** Task 3 only. This table mirrors Table 1 but restricts the aggregation to Task 3 groups. Gemini 3.1 Pro Preview Pass 1, Gemini 3.1 Pro Preview Pass 2, Opus 4.6, and GPT-5.4 each cover all 100 Task 3 groups; GPT-5.3-Codex covers the 10 Task 3 shard-1 groups. Panel A reports average ranks, where lower numbers are better. Panel B reports average scores. Panel C reports the standard deviation of scores.

Reviewer Models	Submission Sources			
	GPT-5.4	GPT-5.3-Codex	Opus 4.6	Human
Panel A: Average Rank				
GPT-5.4	1.18	1.98	3.09	3.75
GPT-5.3-Codex	1.10	2.20	2.80	3.90
Opus 4.6	1.09	2.46	2.68	3.77
Gemini 3.1 Pro Preview Pass 1	1.27	2.06	2.74	3.93
Gemini 3.1 Pro Preview Pass 2	1.21	2.02	2.84	3.93
Panel B: Average Score				
GPT-5.4	80.4	73.4	60.7	41.1
GPT-5.3-Codex	83.8	73.7	67.1	42.1
Opus 4.6	76.6	64.4	62.1	40.9
Gemini 3.1 Pro Preview Pass 1	91.4	84.1	75.3	38.9
Gemini 3.1 Pro Preview Pass 2	92.6	84.9	72.7	38.3
Panel C: SD of Score				
GPT-5.4	4.7	6.2	7.3	15.7
GPT-5.3-Codex	2.6	5.7	6.8	15.4
Opus 4.6	4.2	8.3	7.0	13.0
Gemini 3.1 Pro Preview Pass 1	10.2	9.7	10.3	14.4
Gemini 3.1 Pro Preview Pass 2	8.1	8.1	10.8	13.8

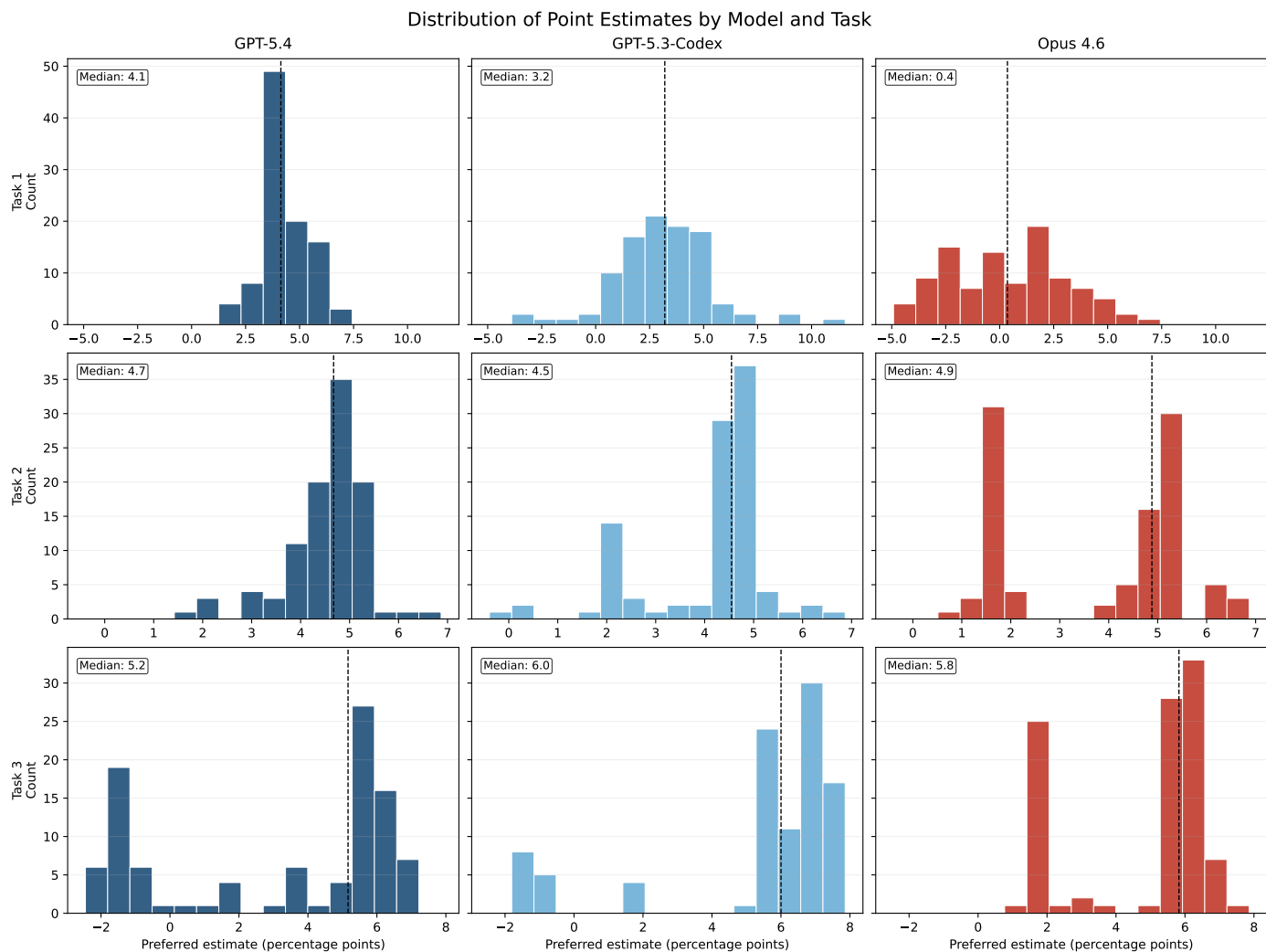


Figure 5: **Point-Estimate Histograms by Model and Task:** Each panel shows the histogram of preferred point estimates across the 100 independent AI replications for a given task-model pair. Columns correspond to GPT-5.4, GPT-5.3-Codex, and Opus 4.6; rows correspond to Tasks 1–3. The dashed vertical line marks the panel median.

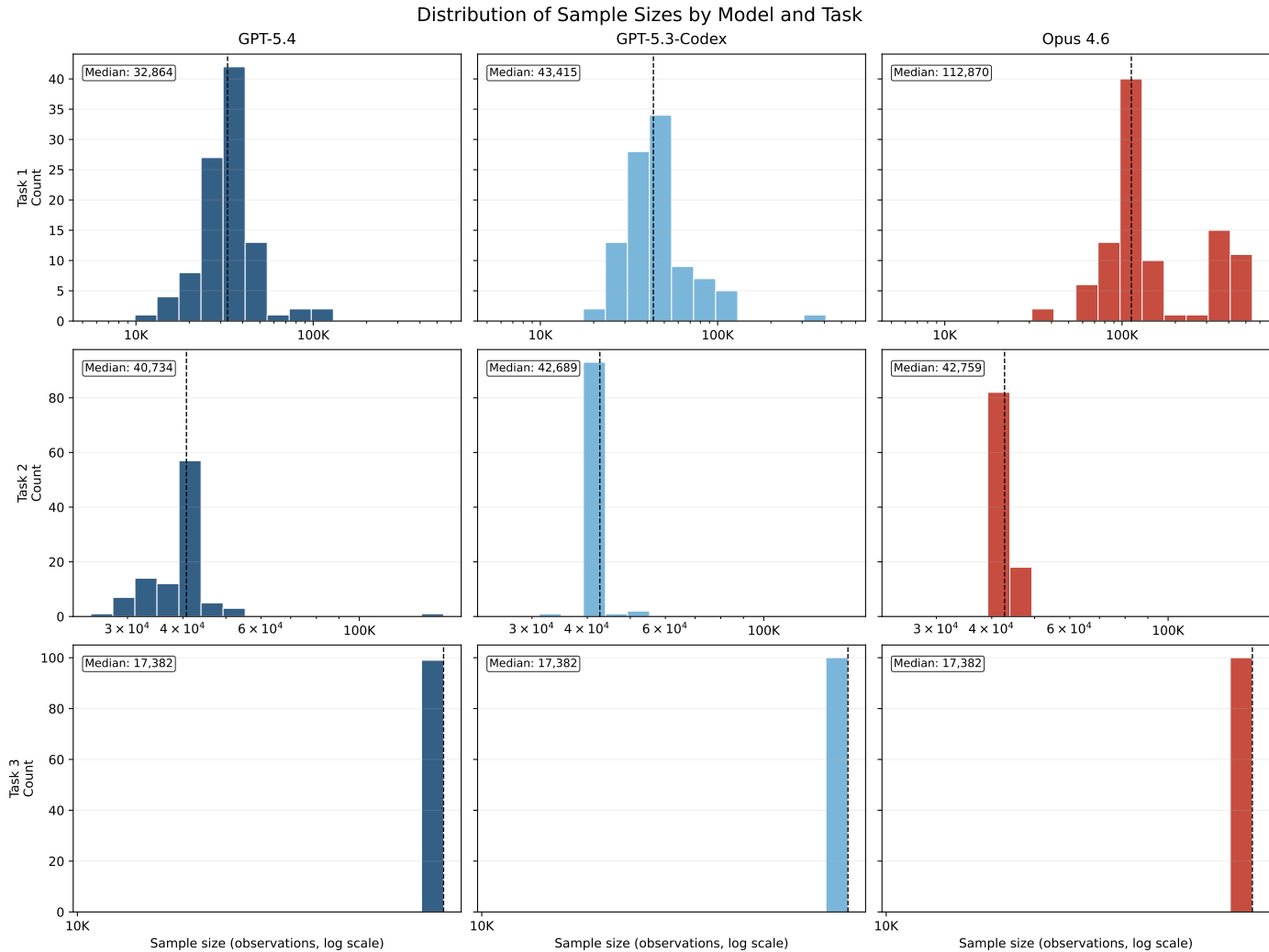


Figure 6: **Sample-Size Histograms by Model and Task:** Each panel shows the histogram of preferred-specification sample sizes across the 100 independent AI replications for a given task-model pair. Columns correspond to GPT-5.4, GPT-5.3-Codex, and Opus 4.6; rows correspond to Tasks 1–3. The horizontal axis is on a log scale, and the dashed vertical line marks the panel median.

Task 1: GPT-5.4

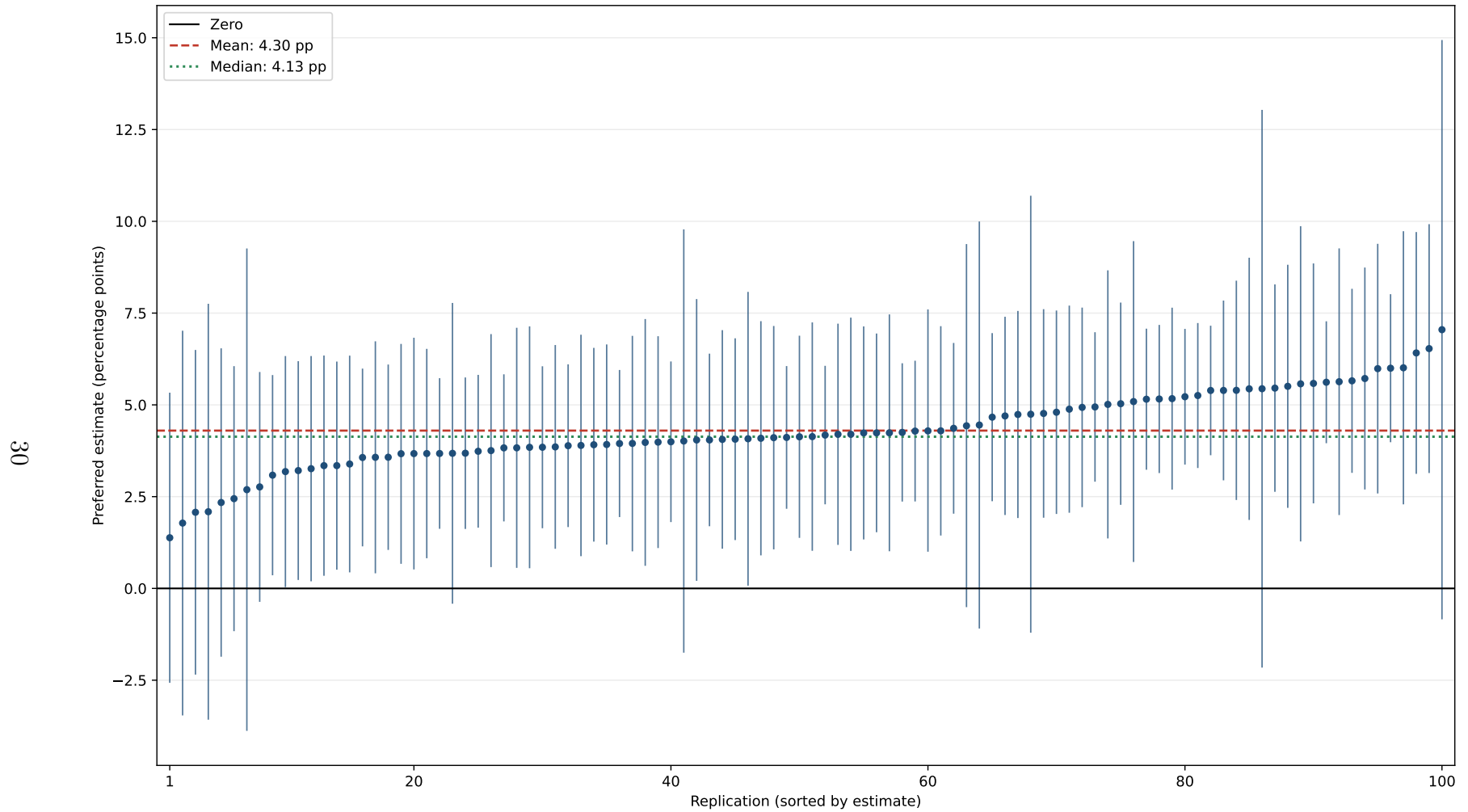


Figure 7: **Forest Plot of Preferred Estimates:** Task 1, GPT-5.4. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

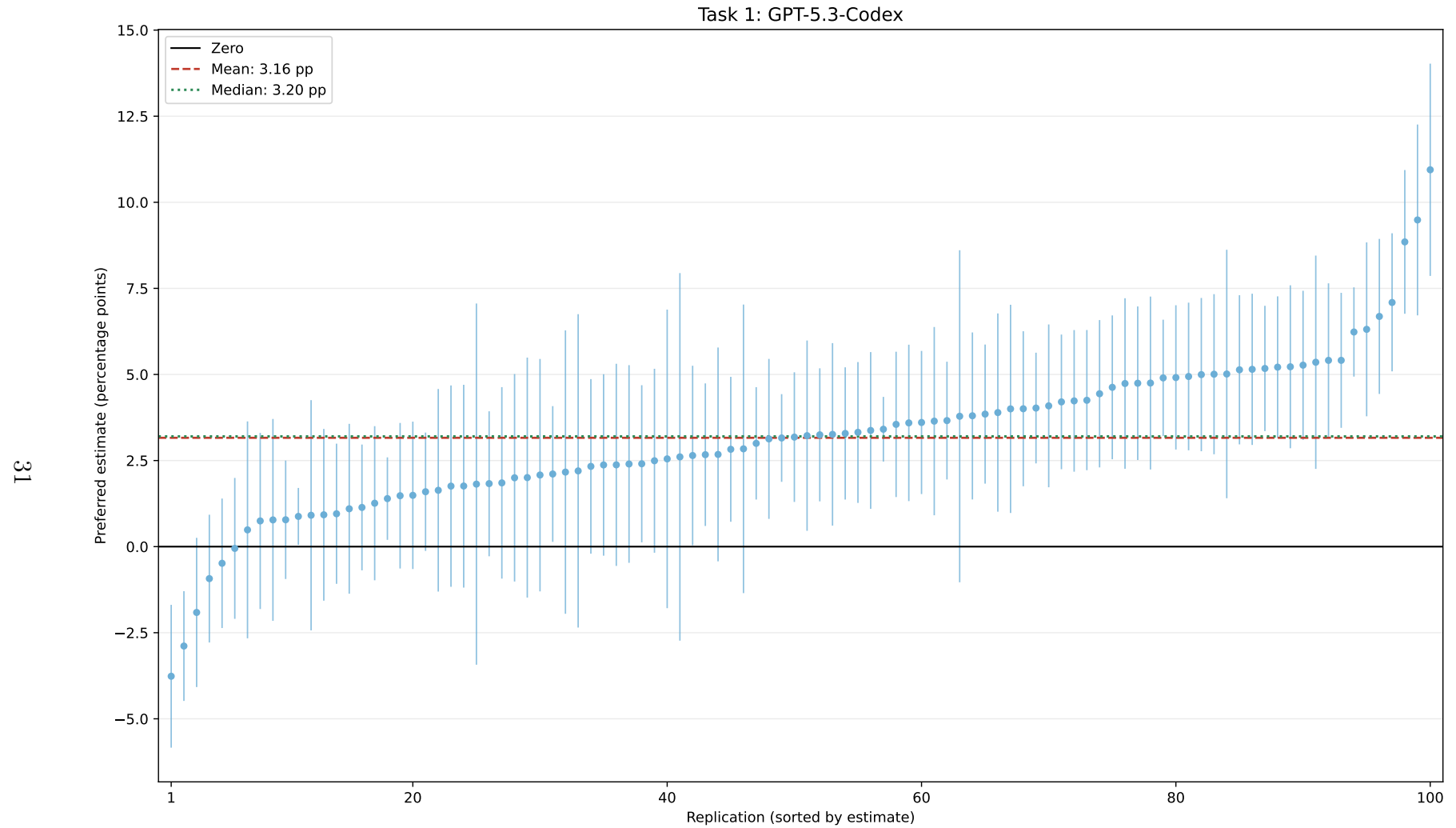


Figure 8: **Forest Plot of Preferred Estimates:** Task 1, GPT-5.3-Codex. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

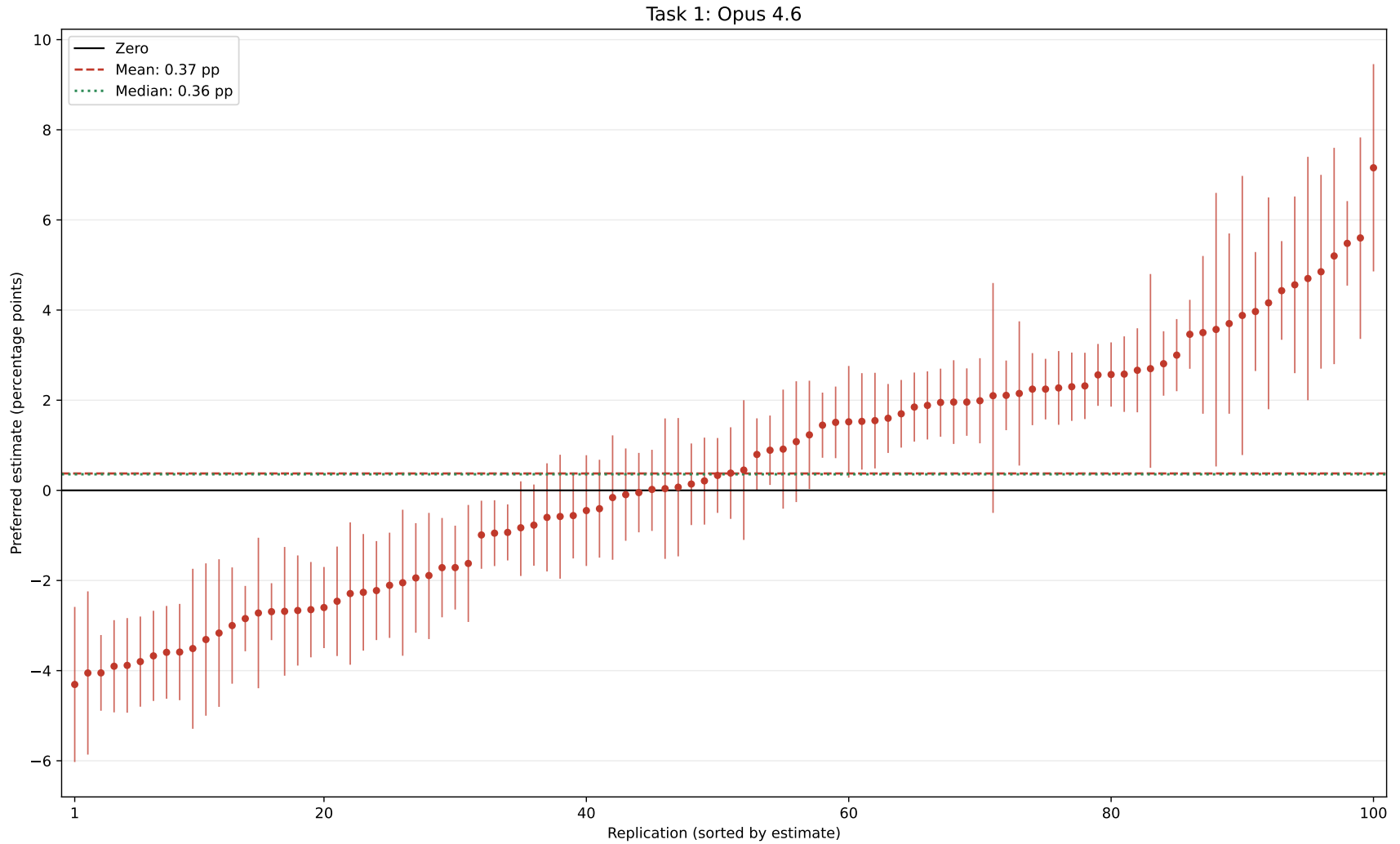


Figure 9: **Forest Plot of Preferred Estimates:** Task 1, Opus 4.6. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

Task 2: GPT-5.4

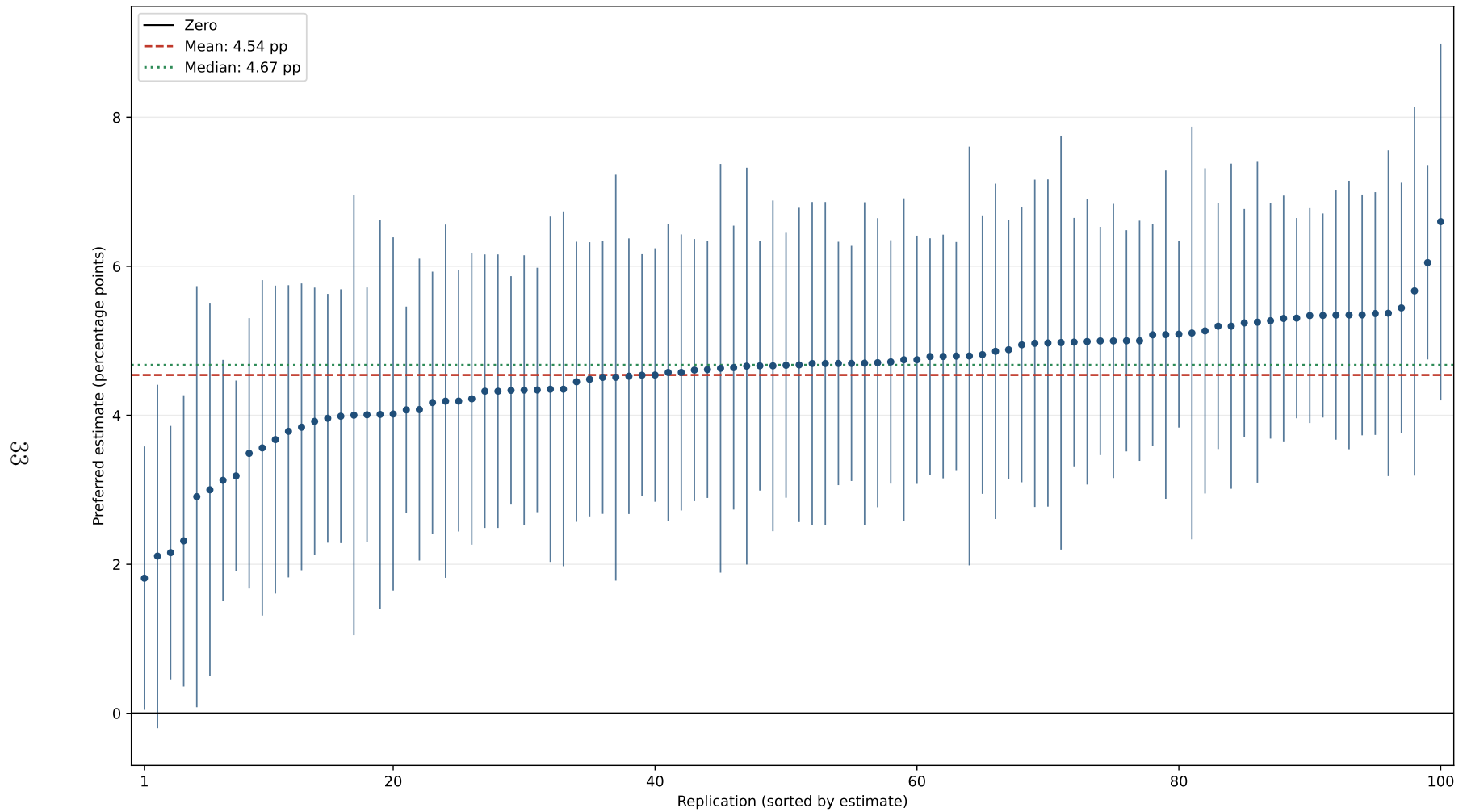


Figure 10: **Forest Plot of Preferred Estimates:** Task 2, GPT-5.4. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

Task 2: GPT-5.3-Codex

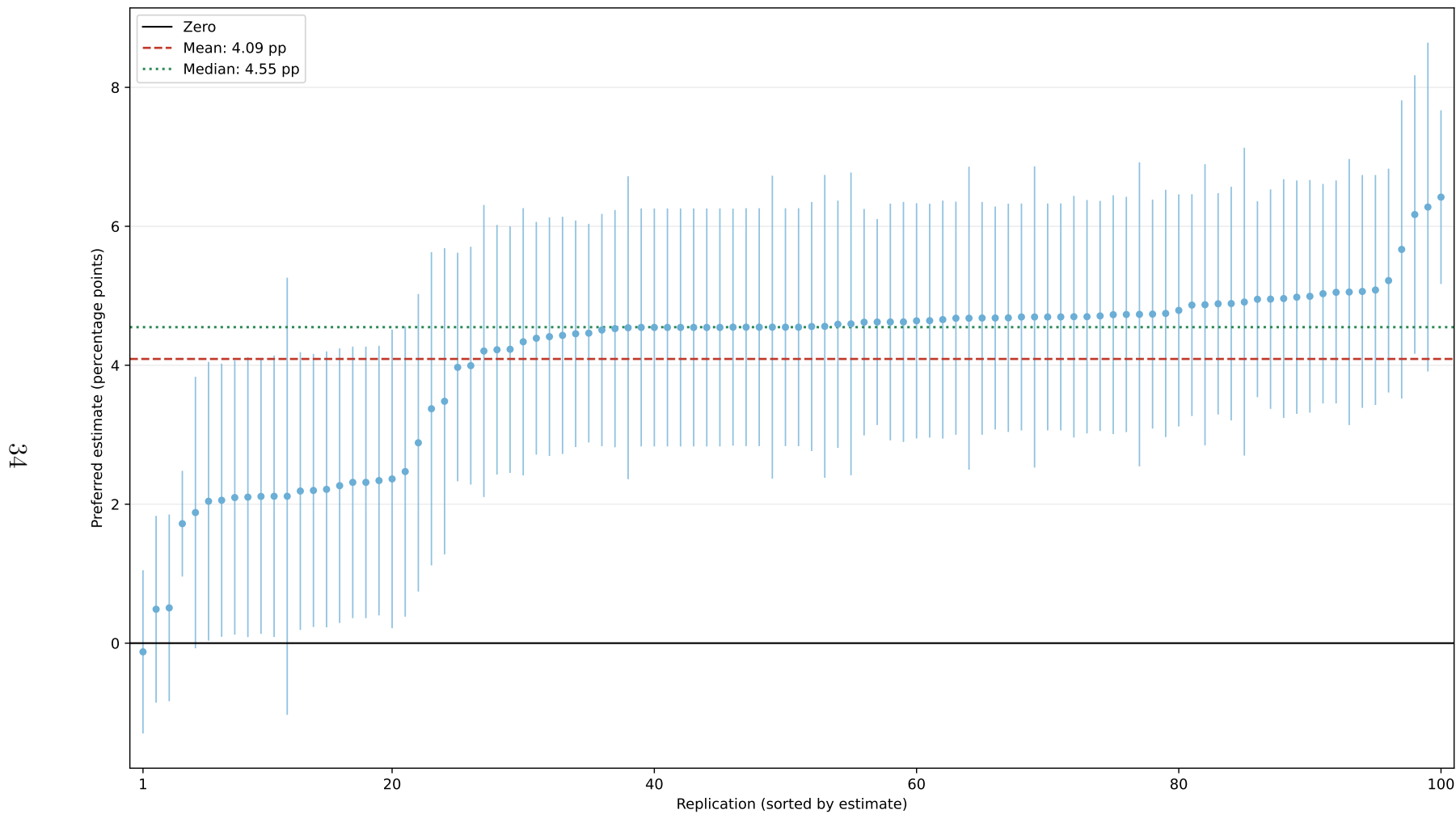


Figure 11: **Forest Plot of Preferred Estimates:** Task 2, GPT-5.3-Codex. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

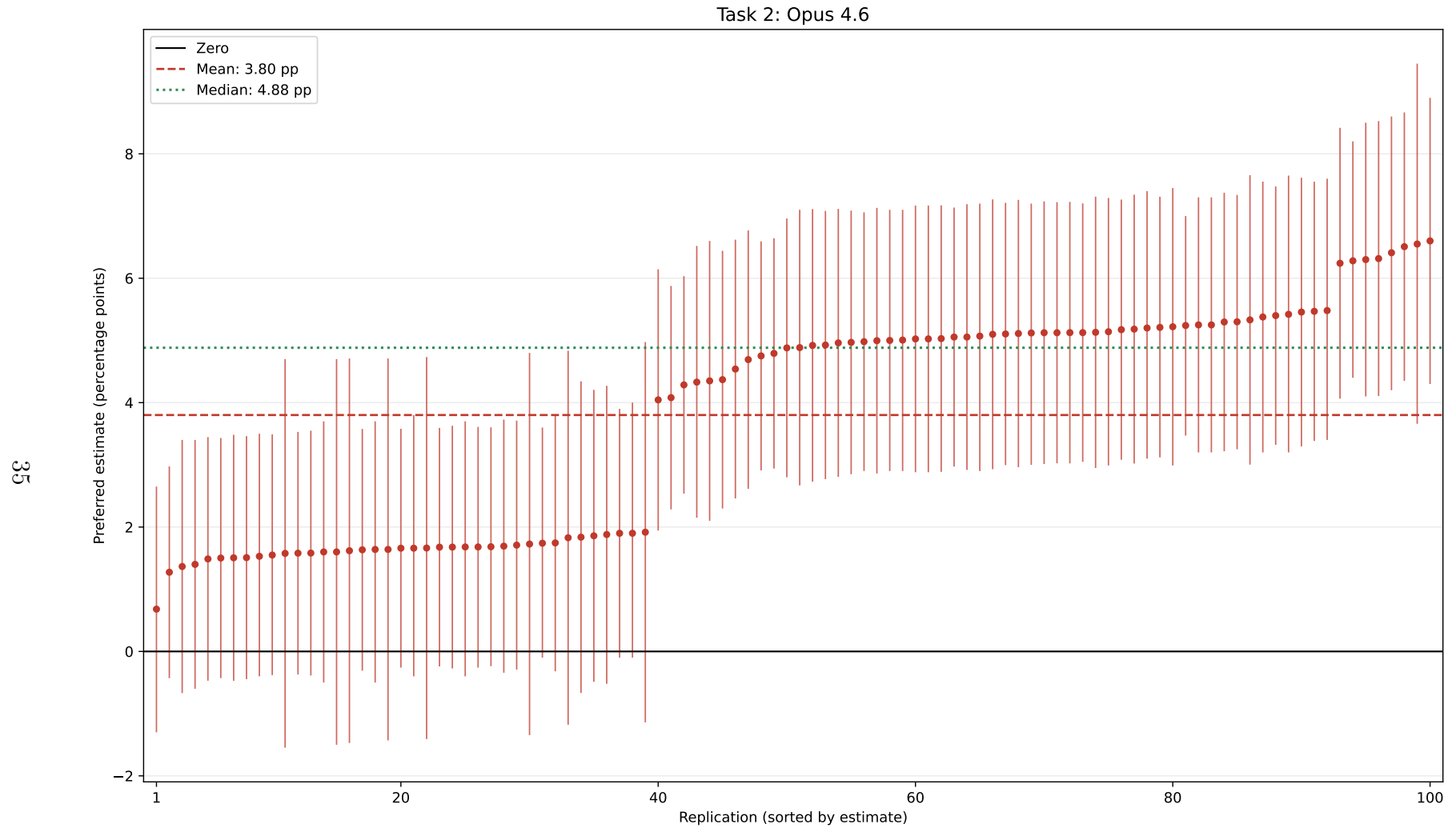


Figure 12: **Forest Plot of Preferred Estimates:** Task 2, Opus 4.6. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

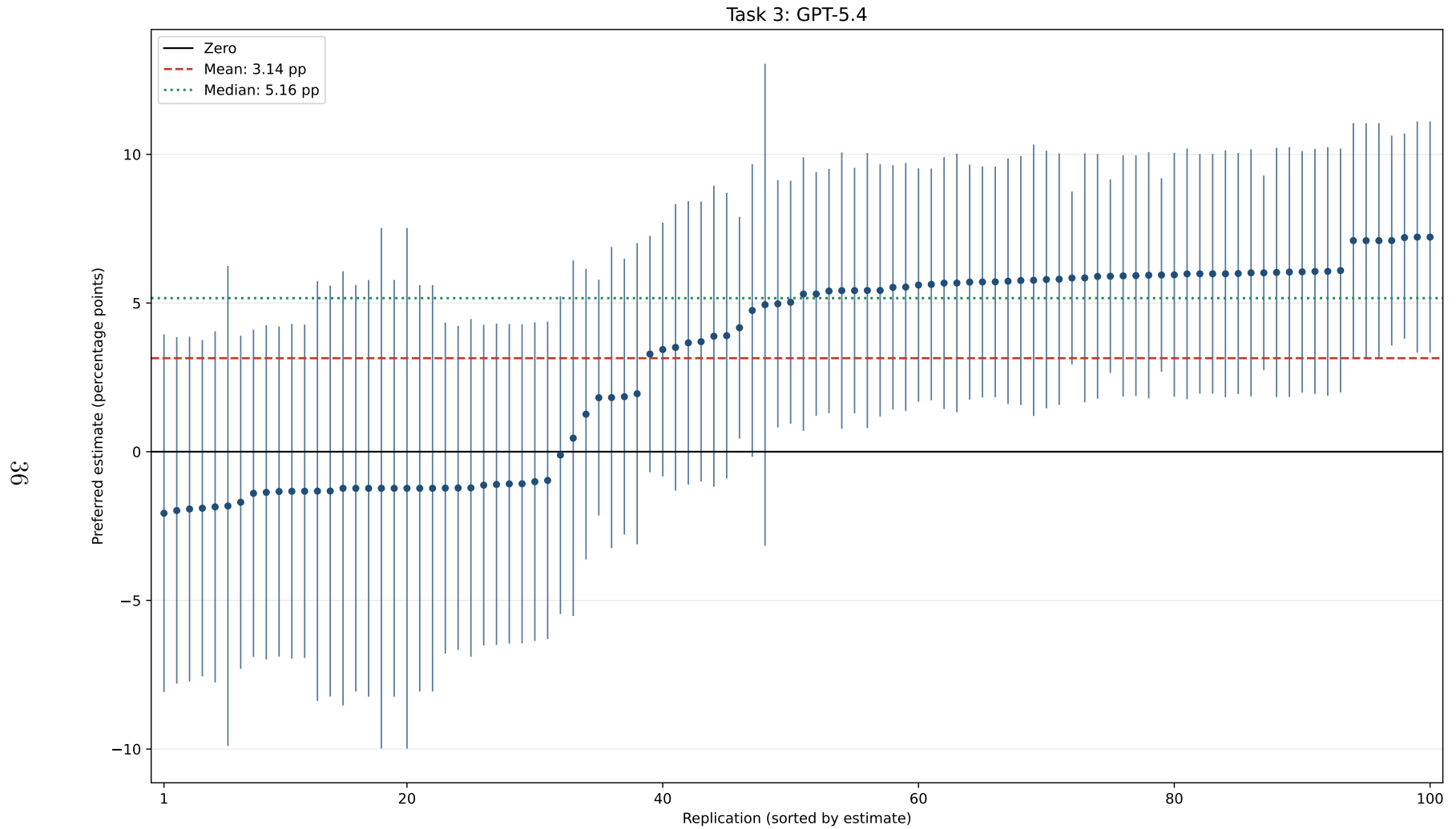


Figure 13: **Forest Plot of Preferred Estimates:** Task 3, GPT-5.4. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

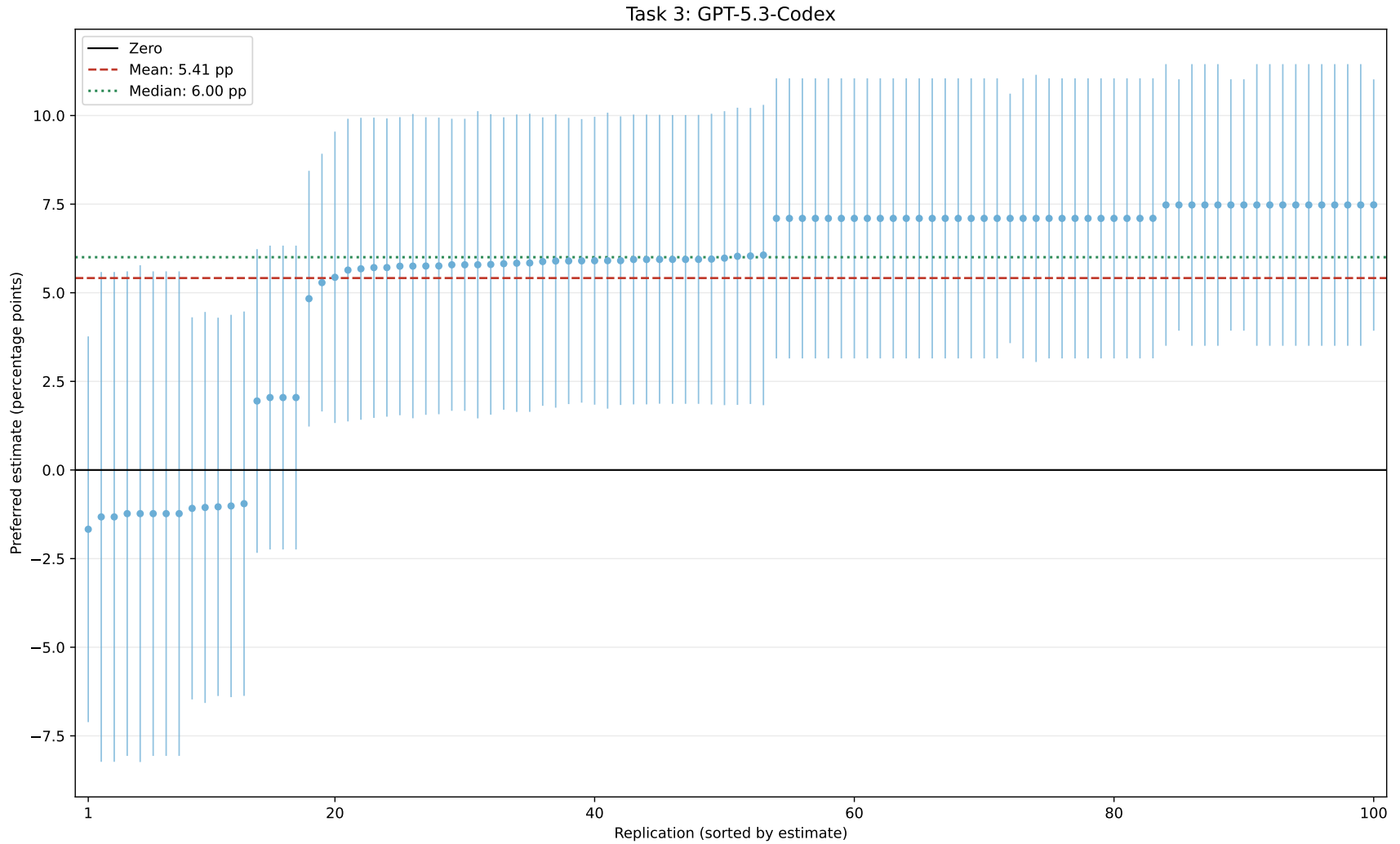


Figure 14: **Forest Plot of Preferred Estimates:** Task 3, GPT-5.3-Codex. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

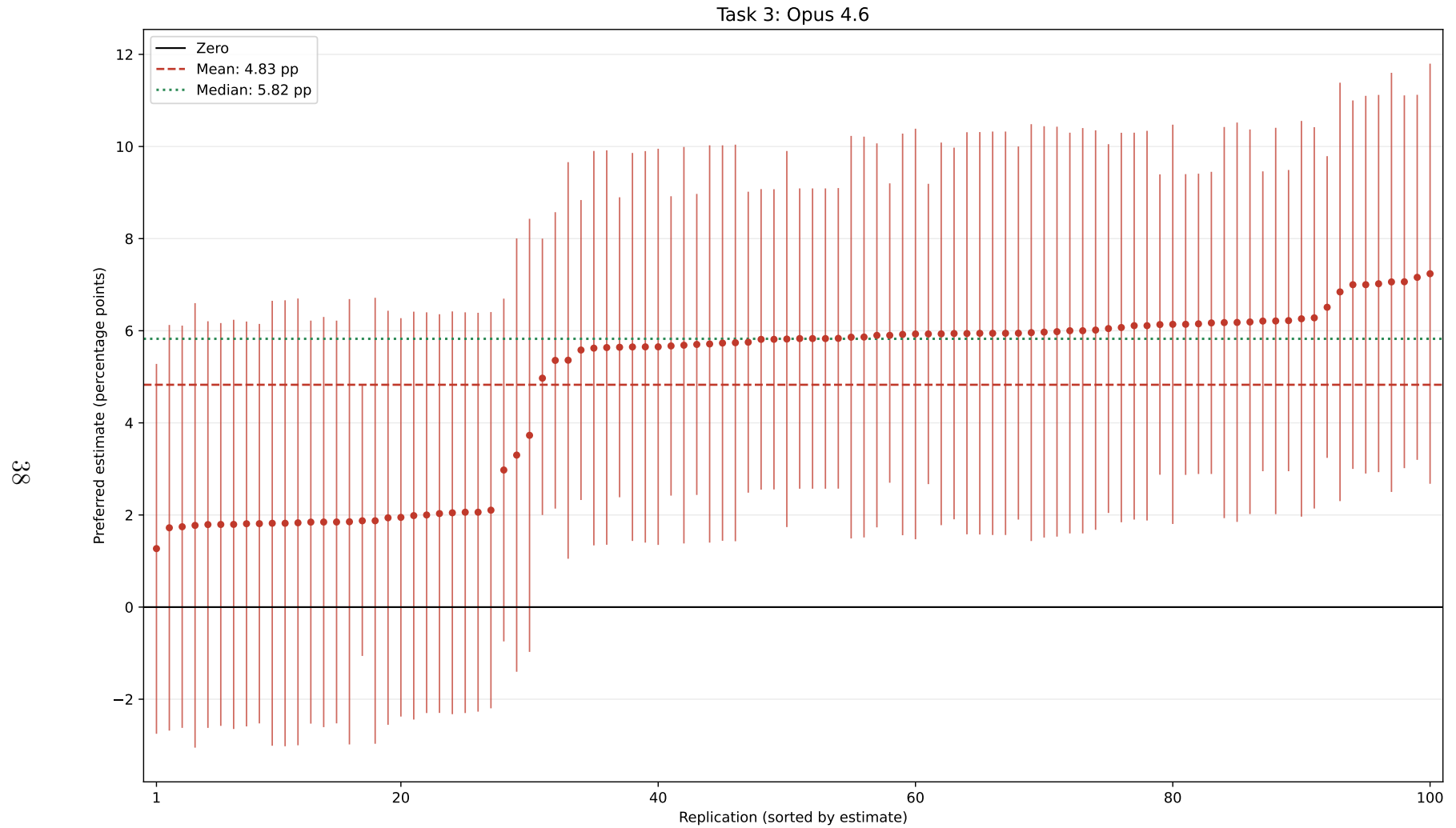


Figure 15: **Forest Plot of Preferred Estimates:** Task 3, Opus 4.6. The point estimates are sorted from left to right from lowest to highest, with vertical 95% confidence intervals for each replication. The solid horizontal line marks zero, the dashed horizontal line marks the mean estimate, and the dotted horizontal line marks the median estimate.

B Replication Instructions

The instructions below are reproduced from [Huntington-Klein et al. \(2025\)](#). All three task documents were given to both human researchers and the AI systems. For the AI 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.

C Review Prompt

You are an economist. Multiple candidates wrote code to answer the same research question by estimating a treatment effect. Your job is to read each candidate’s code and rank candidates.

The research question candidates were asked to answer: "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."

Files provided:

- task_instructions/replication_instructions.txt
- task_instructions/comparison_report.template.tex
- task_instructions/decision_template.json
- candidate_*/ (one folder per candidate label present in this match; each contains: code/ and writeup/)
- Start by reading task_instructions/replication_instructions.txt.
- Then read the two template files in task_instructions/ before drafting outputs.

Template workflow:

- Start from 'task_instructions/comparison_report.template.tex' and 'task_instructions/decision_template.json'.
- Copy them to 'comparison_report.tex' and 'decision.json'.
- Replace every '@@...@@" placeholder in those copied files.
- Keep the template markers intact:
 - '% REPORT_TEMPLATE_V1' in 'comparison_report.tex'
 - "template_version": "decision_template_v1" in 'decision.json'
- Do not edit the template files in place; leave them unchanged in 'task_instructions/'.
- Do not change section order, headings, bold labels, or the base LaTeX preamble unless you must duplicate an evidence block to add more evidence items.
- If you need more than 9 evidence items, duplicate the last evidence block, renumber it, and keep the same format.
- Leave no '@@...@@" tokens unreplaced in the final output files.

Rules:

- Work only in the provided directory.
- Do not infer candidate identities from style, formatting, verbosity, or artifacts.
- Treat programming language as neutral (Stata/R/Python/etc.).
- Do NOT reward coding style, formatting, verbosity, or polished presentation.
- In this review profile, the write-up is secondary evidence. Use it mainly to identify the preferred estimate, the intended estimand/specification, and claimed robustness checks. Code remains the primary evidence for ranking and scoring.
- Do NOT penalize missing data files.
- Do NOT judge by sign, magnitude, statistical significance, CI width, or sample size alone.
- Judge the methodology, not whether the findings confirm your prior.
- Use concrete code evidence for every important claim.
- For every major critique, explain mechanism and likely bias direction (upward, downward, or ambiguous). If ambiguous, explain the competing channels.
- Weight identification strategy most heavily in scoring and ranking.
- Do not repeat the same critique in full across multiple sections. Explain it once in the most relevant section, then refer back briefly if needed.

What you must discuss for each candidate (11 required sections):

- 1) Sample construction.
- 2) Definitions of treatment and control groups.
- 3) Definition of the outcome variable (full-time employment).
- 4) Estimated specification:
 - Write the core estimating equation in notation.
 - Discuss it.
 - Discuss whether it answers the research question.
 - Show a code anchor for the core implemented estimator.
 - State whether the notation accurately matches the implemented code.
- 5) Parallel Trends Assumption (PTA):
 - PTA is the main assumption for DiD and DDD.
 - PTA says: "In the absence of treatment, the treated and control groups would have experienced the same change in outcomes over time."
 - PTA cannot be directly tested.
 - Discuss PTA plausibility using code-visible design choices, pre-trends, event studies, placebos, and counterfactual logic.
- 6) No-anticipation and overlap/support requirements.
- 7) Treatment effect vs age:
 - Explain how age or cohort dynamics may confound the treatment effect estimates.
- 8) Covariates and fixed effects:
 - Discuss omitted confounders, omitted fixed effects, and treatment-affected bad controls.
- 9) Robustness, assumptions, heterogeneity of effects:
 - Pre-trends, placebos, sensitivity checks, robustness checks, heterogeneity.
- 10) Standard errors and inference:
 - How uncertainty is estimated and whether that is appropriate.
- 11) Fatal flaws and other material issues (if any):
 - Label "fatal flaw" only when supported by direct code evidence.
 - If evidence is indirect, label it as a major methodological risk instead.

Important identification guidance:

- Nearly all submissions in this task use DiD or DDD. Assume DiD or DDD reasoning is relevant unless the code clearly uses a different design.
- If the design is DiD or DDD, you must explicitly discuss PTA, no-anticipation, and overlap/support.
- If the design is not DiD or DDD, state the actual identifying assumptions used in code and write exactly: "PTA and no-anticipation are not the primary assumptions for this design."

Estimated specification requirement:

- In the "Estimated specification" subsection, you must include:

- 1) one displayed equation in notation;
 - 2) one short paragraph explaining whether that equation answers the research question;
 - 3) one code anchor showing the exact code implementation of the core estimator.
- If the displayed equation would be too long for one line, write it as a multi-line aligned display equation.
 - Break lines only after the main equals sign or at top-level '+' / '-' operators.
 - Never break inside indicator functions, sums, products, braces, parentheses, or interaction terms.
 - Prefer compact notation for controls and fixed effects rather than letting the equation run off the page.

- Use this pattern when needed:

```
\[
\begin{aligned}
Y_{\{ist\}} = {} & \{ \} & \langle \text{first part} \rangle \\
& & \&+ \langle \text{next part} \rangle \\
& & \&+ \langle \text{next part} \rangle
\end{aligned}
\]
```

- Use this exact code-anchor format:

Code anchor:

File: `\path{candidate_X/code/filename.ext}`

Line(s): 123-130

Snippet:

```
\begin{Verbatim}[breaklines=true,breakanywhere=true,fontsize=\small]
<short verbatim snippet>
\end{Verbatim}
```

Equation-to-code check: <one concise sentence saying whether the notation accurately matches the implemented code>

Score guidance:

- Give each candidate a total score from 1 to 100 inclusive.
- Use whole integers only.
- Scores should reflect absolute credibility, not only ordinal rank.
- Use these anchors:
 - 91-100: highly credible design with limited remaining concerns
 - 76-90: strong design with important but non-fatal limitations
 - 61-75: mixed credibility with meaningful identification or implementation threats
 - 41-60: weak credibility with serious threats to causal interpretation
 - 1-40: fatal flaw, estimand mismatch, or very weak causal credibility

Required output files:

- 1) comparison_report.tex
- 2) decision.json
- 3) review_log.md

Output-file workflow requirements:

- ‘comparison_report.tex’ must be based on ‘task_instructions/comparison_report.template.tex’.
- ‘decision.json’ must be based on ‘task_instructions/decision_template.json’.
- Keep the structure encoded in those templates as the source of truth for formatting.
- Fill the placeholders rather than rewriting the structure.

Placeholder content requirements for comparison_report.tex:

Summary section:

- For each ‘@@RANK_<N>_SUMMARY_SENTENCE@@’, write exactly one sentence of about 18-28 words summarizing the main reason for that candidate’s rank.
- For ‘@@SUMMARY_RANKING_JUSTIFICATION@@’, write exactly 1 short paragraph with 4-6 sentences justifying the ranking and the size of the margins between adjacent candidates.
- For 3-candidate matches, state clearly why first beats second and why second beats third.
- For 2-candidate matches, state clearly why first beats second.

Candidate-by-candidate section:

- The template already fixes subsection order, numbered headings, and the order of ‘Strengths’, ‘Risks’, ‘Minimal fixes needed’, and ‘Final score’. Do not rewrite that structure.
- Fill candidates in final-rank order, best to worst.
- For ‘@@RANK_<N>_PREFERRED_ESTIMATE@@’, print only the coefficient value itself, for example ‘0.0490’, or ‘NOT FOUND’.
- Do not add percentage-point translations, coefficient names, parenthetical explanations, or other prose in ‘@@RANK_<N>_PREFERRED_ESTIMATE@@’.
- For ‘@@RANK_<N>_PREFERRED_ESTIMATE@@’, ‘@@RANK_<N>_CI@@’, and ‘@@RANK_<N>_SAMPLE_SIZE@@’, use ‘writeup/report.txt’ when it contains an explicit value, or another explicit value from the write-up folder when clearly tied to the preferred estimate.
- If the exact value is not explicit in the write-up folder, write ‘NOT FOUND’.
- If the write-up and code disagree about the preferred estimate, specification, sample, or robustness claims, trust the code for methodological assessment and explicitly note the mismatch.
- For each numbered discussion placeholder from sample construction through fatal flaws, write exactly 1 paragraph.
- Each numbered discussion paragraph should usually be 3-6 sentences.
- ‘Definition of the outcome variable’ and ‘Standard errors and inference’ may be 2-4 sentences.
- Aim for about 2.5-3.5 pages per candidate, but prioritize substance over padding.
- For ‘@@RANK_<N>_STRENGTH_1@@’, ‘@@RANK_<N>_STRENGTH_2@@’, ‘@@RANK_<N>_RISK_1@@’, ‘@@RANK_<N>_RISK_2@@’,

‘@@RANK_<N>_FIX_1@@’, and ‘@@RANK_<N>_FIX_2@@’, write one concise bullet each.

- If a discussion subsection is genuinely not applicable, write exactly: ‘Not applicable for this design.’
- In the fatal-flaws paragraph, label ‘fatal flaw’ only when supported by direct code evidence.
- For ‘@@RANK_<N>_SCORE@@’, use an integer from 1 to 100. The template already prints ‘/100’, so do not add any explanation or extra text.

Comparison section:

- ‘@@COMPARISON_TOP_CANDIDATES@@’: 3-5 sentences comparing the top candidates and explaining decisive margins.
- ‘@@COMPARISON_WEAKEST_CANDIDATES@@’: 3-5 sentences comparing the weakest candidates and identifying the main failure modes.
- ‘@@COMPARISON_CROSS_CUTTING_LESSONS@@’: 3-5 sentences giving cross-cutting lessons from the full set.

Confidence and unknowns section:

- ‘@@OVERALL_CONFIDENCE@@’ must begin with ‘Overall confidence: ‘ and use one of these ratings: ‘High’, ‘Medium’, or ‘Low’.
- ‘@@OVERALL_CONFIDENCE@@’ should be 3-4 sentences.
- ‘@@KEY_UNKNOWNS@@’ should be 3-4 sentences listing the most important unresolved unknowns and explaining how they affect certainty.

Supporting code evidence section:

- Use the provided template block for the first 9 evidence items and keep the same format if you add more.
- Provide at least 9 evidence points total across all candidates.
- Provide at least 2 evidence points for the top-ranked candidate.
- Provide at least 2 evidence points for the bottom-ranked candidate.
- For every candidate you classify as having a fatal flaw, include at least 1 explicit evidence item tied to that fatal-flaw claim.
- Do NOT use a table in this section.
- For each evidence item, fill the file, line(s), snippet, and why-it-matters placeholders.
- Keep snippets short; use ‘...’ when quoting only the relevant segment.
- Do not use absolute paths.

Decision rules for consistency:

- Keep ranking and scores exactly consistent across comparison_report.tex and decision.json.
- Do not invent numerical values.
- Fill ‘decision.json’ by replacing the template placeholders rather than rewriting the schema.
- All total and dimension scores in ‘decision.json’ must be integers from 1 to 100 inclusive.
- ‘@@RATIONALE@@’ should be a short justification.

review_log.md:

- Use a short bullet list of key files inspected and key audit steps.
- Keep it concise and factual.

Formatting constraints:

- ASCII only in all output files.
- Keep the prose formal, direct, and economical.

Begin the review.

References

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

Fabrizio Dell’Acqua, Edward McFowland, Ethan R Mollick, Hila Lifshitz-Assaf, Katherine Kellogg, Saran Rajendran, Lisa Kraymer, 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.

Ruijiang Gao and Steven Chong Xiao. Nonstandard errors in AI agents. *arXiv preprint arXiv:2603.16744*, 2026.

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.

Serafin Grundl. Claude code as an empirical economist: Like humans but without the tails. Technical report, 2026. Available at SSRN: <https://ssrn.com/abstract=6219138>.

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.

- Wenqian Huang, Albert J. Menkveld, and Shihao Yu. AI “errors”. 2026. Working paper, Bank for International Settlements.
- 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.
- 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.
- 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>.
- 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.