What to Do When Humans Are No Longer the Gold Standard

Large Language Models, State of the Art and Robustness for Politics Research*

James Bisbee
Vanderbilt University
james.h.bisbee@vanderbilt.edu

Arthur Spirling
Princeton University
arthur.spirling@princeton.edu

Abstract

We consider the research implications of large language model (LLM) capabilities approaching, perhaps exceeding, those of highly-trained humans in political science. We note that frontier LLMs demonstrate near-expert performance for many data annotation tasks, and they are getting better over time. We show what this will mean for inference in downstream tasks: optimistically, it is that estimated treatment effects will become larger, although claimed null effects may be more dubious. We argue authors should focus more on sensitivity and robustness with respect to future technological change, and we demonstrate how to use local calibration for such problems. We discuss how our findings, combined with the fact that performance is inherently bounded above (at 100%), should affect debates on the importance of using proprietary "State of the Art" versus open-weight, replicable LLMs. We make available fast and free software (futureProofR) for implementing our suggestions.

*First draft: June 13, 2025. This draft: February 10, 2026. Leonardo Dantas provided excellent research assistance. Walter Mebane and Mitchell Bosley provided very helpful comments on earlier drafts. We thank audiences at Polmeth, University of Tokyo, UNC Charlotte, APSA and Waseda University for feedback.

1 Introduction

In a short space of time, Large Language Models (LLMs) have gone from a potentially helpful technical novelty to a standard part of the social science research toolkit. Analysts use them because they work: for coding data, especially text, they are fast, cheap and demonstrate excellent performance (e.g. Gilardi, Alizadeh and Kubli, 2023). This paper is about what happens, or should happen, in light of two developments related to this use-case. The first is that today's LLMs are approaching or surpassing human quality standards, including those of experts, for at least some tasks (e.g. Bosley, 2024; Choi, Peskoff and Stewart, 2025; Heseltine and Clemm von Hohenberg, 2024; Wu, 2025). The second development is consistent improvement in State of the Art (SOTA) performance of these models over time: they are getting better and better.

These trends raise dilemmas for researchers in political science. This is because typically we rely on human experts to calibrate the performance of our automatic machine-based coding methods. That is, we believe there is a true position of a party, a true number of people at a protest, a true topic of a news story and that humans are the best assessors of these things. But if we no longer believe the "gold standard" is human, then by extension we do not believe the ideal validation set is human, either. Otherwise put, human-level performance on a task may no longer be the target metric.[1] Yet with no fixed human standard to compare to, we will find it hard to comment meaningfully on how accurate or precise a given LLM is. And if we cannot know where, or in what ways they make errors, we cannot "correct" these errors or the biases they induce for our downstream estimates based upon their coding decisions. This problem is compounded by the fact that we suspect future models—which by definition we cannot yet see—will make different but improved classification decisions on the same data.[2]

[1] We assume here that the goal is not to emulate humans *qua* humans with all their errors and biases (cf Argyle et al., 2023; Bisbee et al., 2024).

[2] This challenge is an especially rapid example of the broader challenge of temporal validity in social

We have in mind two scenarios. The first is where a researcher does not have access to the best performing SOTA model. This might be a matter of cost, or because that SOTA option is proprietary and the researcher wants to use a replicable and versionable open-weight alternative (e.g. Palmer, Smith and Spirling, 2024). Or it may be that the researcher *was* using a SOTA model when the project began, which has now been surpassed and perhaps no longer supported by its original producer. This latter case is quite general. Given the speed of LLM development, it is common to read papers using an older, even discontinued model, and it is unclear how findings that rely on coding with that model would hold up today.

This paper examines how researchers can and should deal with this changing landscape. At a high-level, our argument is that researchers should move their focus to robustness in the face of technological change. Obviously we are not the first political scientists to note that varying specifications of different models matter for inference (e.g. Blackwell, 2014; Duarte et al., 2024; Imai and Yamamoto, 2010), but we innovate by making these points in the specific context of LLM coding relative to humans. In particular, we propose a sensitivity framework to bound effects for this context. We argue that for many social science contexts, improvements in LLM performance will mostly strengthen the substantive conclusions drawn in research that rejects a null hypothesis. Furthermore, we show that the ceiling effects implicit in high performance models mean that point estimates of interest are unlikely to change dramatically. This does not mean one should be unwilling to update one's model choices as technology advances; but it may mean that earlier results are more robust than commonly thought and that one can be precise about this idea.

To fix ideas, we hone in on a long-standing, common scenario in the discipline that allows us to speak precisely to the issues. That set up is measurement error in a binary treatment variable and the associated estimate of the average treatment effect (e.g. Aigner, 1973; Bollinger, 1996; Card, 1996). The key is that the treatment is potentially mis-coded

science research (Munger, 2023).

for some observations, could be classified by an LLM, and may be increasingly well classified over time. And unlike many previous approaches (e.g. Egami et al., 2023; Fong and Tyler, 2021; TeBlunthuis, Hase and Chan, 2024; Zhang, 2021), we explore the plausible case where there is no available (human) gold standard that obviously surpasses LLM performance such that bias corrections can be meaningfully made. The tools and results we provide for this case are helpful *per se* but also aid in making broader points about an increasingly common problem. We will demonstrate this wider usefulness by also applying the techniques to the more straightforward miscoded *dependent* variable case, as well as an extension to a multi-class categorical setting.

In the next section, we will define terms more precisely, and explain why we believe humans may not now constitute the gold standard in many cases. We then provide evidence that, for our focus problem at least, improved LLM accuracy is mostly good news. This is followed by a broader discussion of robustness—wherein we introduce some tools for others to use. We conclude with a discussion on related and open research questions.

2 Gold Standards, Improvements and Robustness

We will assert and then demonstrate that LLMs are good and getting better. But we first provide some simple definitions to structure our presentation. We say a model, algorithm or technique is **State of the Art** (SOTA) if it achieves current best performance for a given set of tasks. We will be specific about what "best performance" or "frontier performance" means in our binary treatment application, but it could refer to many different metrics like recall or precision. We obtain these metrics via a set of generally accepted correct answers or resolutions of tasks that we give the model to complete. If they exist, we will refer to these authoritative answers as the "**gold standard**" for a specific problem. Examples include academic expert ratings of manifesto sentences (see e.g. Benoit et al., 2016). A **validation**

set is some portion (perhaps all) of the data classified in a way that meets the gold standard, and that a given model or techniques predictions are checked against to assess performance. Any model that is not SOTA is **inferior**, by definition.

2.1 What if there is no gold standard?

Of course, we acknowledge that whether there has ever existed a truly human gold standard for some tasks is debated. This point reflects the underlying terrain of annotation tasks in social science, which we conceive of as a three-part typology, visualized in Figure 1.

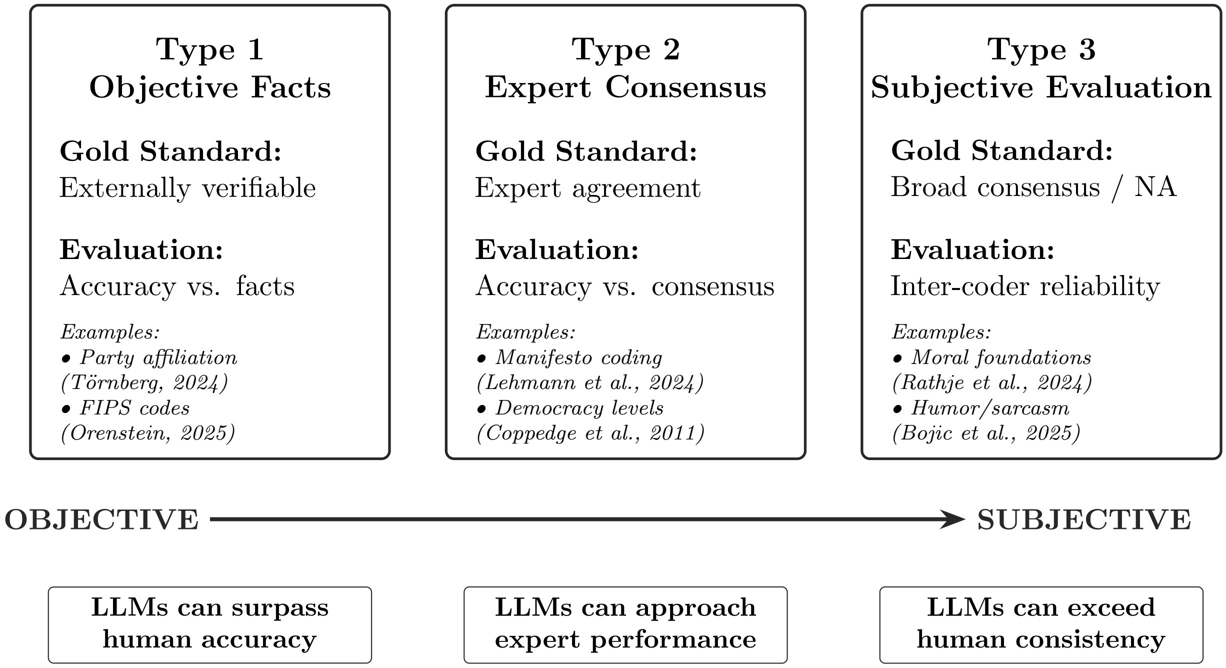

Figure 1: Typology of annotation tasks

At one end are objectively verifiable facts, such as the partisanship of a politician based on voter registration (Törnberg, 2024), the probabilistic record linkage tasks like finding the county FIPS code for an address (Ornstein, 2025). Basic information retrieval tasks have this feature. For these problems, there exists an external, verifiable source of truth against which both human and LLM annotations can be assessed for accuracy.

At the other extreme are inherently subjective or multidimensional annotation tasks for which expecting a single consistent label is impossible or inappropriate. Examples include the emotional content of a social media post (which may simultaneously evoke multiple emotions), aesthetic judgments of creative works (e.g., Xu et al., 2024), or the detection of subtle forms of humor or sarcasm where reasonable observers may legitimately disagree (Bojić et al., 2025).

In between are annotation tasks for which expert consensus is theoretically possible but the labels are not necessarily objective. This includes many core measurement tasks in political science: coding party positions from manifestos (Mikhaylov, Laver and Benoit, 2012), measuring levels of democracy (Coppedge et al., 2011), classifying policy topics from legislative texts (Gunes and Florczak, 2025), or identifying the presence of specific rhetorical strategies in political communication.[3] For these tasks, disagreement among expert coders does not necessarily indicate that some coders are correct and others are wrong; rather, it may reflect genuine ambiguity in the underlying concept, differences in how coders weight various dimensions, or reasonable variation in interpretation. Nevertheless, if it exists, a gold standard standard here would be built on multiple coders having high agreement (in the sense of Krippendorff, 2018) as to how to treat a given observation. But as political science authors regularly point out, for many problems such inter-coder measures are worryingly weak. For instance Mikhaylov, Laver and Benoit (2012) find "rater agreement is exceptionally poor" for manifesto sentence coding. Spinde et al. (2021) report alarmingly low Krippendorff's α values in coded biased language, ranging from 0.144 among crowdworkers to 0.419 among experts, well below the rule-of-thumb threshold of 0.80. In this sense, studies may be built on nominally "gold standard" labels that are nowhere near as certain or as fixed as we would like.

[3] Indeed, whether a given annotation task should be considered Type 2 or Type 3 is a matter of debate (see Little and Meng, 2024).

The robustness and sensitivity tools we provide apply equally across all three annotation types, but our conceptual claims about LLM improvement have differential applicability. For Type 1 tasks (objective ground truth), our central claim—that improvements in LLM accuracy should strengthen rather than weaken downstream estimates under standard attenuation bias assumptions—applies most directly. For Type 2 tasks (expert consensus), this claim holds but requires the additional assumption that LLM improvements represent movement toward consensus expert judgment rather than toward some alternative but equally valid interpretation. For Type 3 tasks (inherently subjective), our conceptual framework about strengthening estimates is less applicable since no gold standard exists, yet our sensitivity analysis tools remain valuable for understanding how conclusions would change under different annotation schemes. We do not claim that LLMs will "solve" subjective annotation problems; rather, they help researchers assess the robustness of their findings to unavoidable annotation ambiguity.[4]

2.2 LLMs are good and getting better

With this typology of annotation ground truth established, we turn to the question of LLM improvements. We are not the first to assert that LLMs are getting better and are approaching (sometimes exceeding) human expert standard. But it is instructive to summarize some evidence for such claims. Figure 2 presents four examples of LLM performance on annotation tasks associated with the three categories discussed above, arranged by the version of the LLM which we use or the date at which it was publicly released. The top three facets summarize twenty one versions of Open AI's GPT models, seven versions of Anthropic's Claude models, and six versions of Mistral's models, tasked with annotating social media

[4] We emphasize that the researcher should be clear on whether they are interested in measuring a latent quality of some item, or if instead they are interested in measuring the latent quality of how the item is perceived by different people. If the latter, then the ability of an LLM to perform "well" in some sense is akin to the research on LLMs as synthetic humans (Argyle et al., 2023; Bisbee et al., 2024), and beyond the scope of our discussion.

posts for authors' partisanship (left facet, Törnberg 2024); party manifestos for policy positions (center facet, Lehmann et al. 2024); and social media posts for moral foundations (right facet, Rathje et al. 2024).

Summarizing first the top row of plots, we view the left facet as an example of a Type 1 classification task, in which there is an objective ground truth insofar as each author is a US senator with a known party affiliation, and document improvements in LLM performance in terms of accuracy (i.e., what proportion of politicians were correctly annotated for partisanship). We view the center facet as an example of a Type 2 classification task, in which the policy classifications are based on expert consensus (Lehmann et al., 2024) using a highly refined codebook from the longest-running dataset in political science, and again document model performance in terms of accuracy. We view the right facet as an example of a Type 3 classification task, in which the interpretation of the moral foundations of a social media post is highly subjective. Here we do not believe accuracy is a meaningful metric, and instead characterize model improvements by comparing the Krippendorff's alpha score among the original human annotators to that we would obtain were we to randomly replace one of the humans with an LLM. The bottom panel summarizes the "Arena" scores of 204 large language models, which are based on a pairwise competition, evaluated by a human judge (Chiang et al., 2024).

While these plots tell roughly the same story (namely, models are generally improving), there are details worth emphasizing. First, there are ceiling effects evident in the top left panel in which all versions of the LLM perform well on average, and exceedingly well for a subset of the quantities of interest (e.g. GPT 5.0 achieves an accuracy of almost 95% when tasked with identifying the partisanship of a U.S. member of congress using only ten of their tweets). While there is room to improve in these specific contexts (i.e., where the ground truth is known), the range is narrow.

Second, a monotonic increase in performance over time is most evident in the bottom

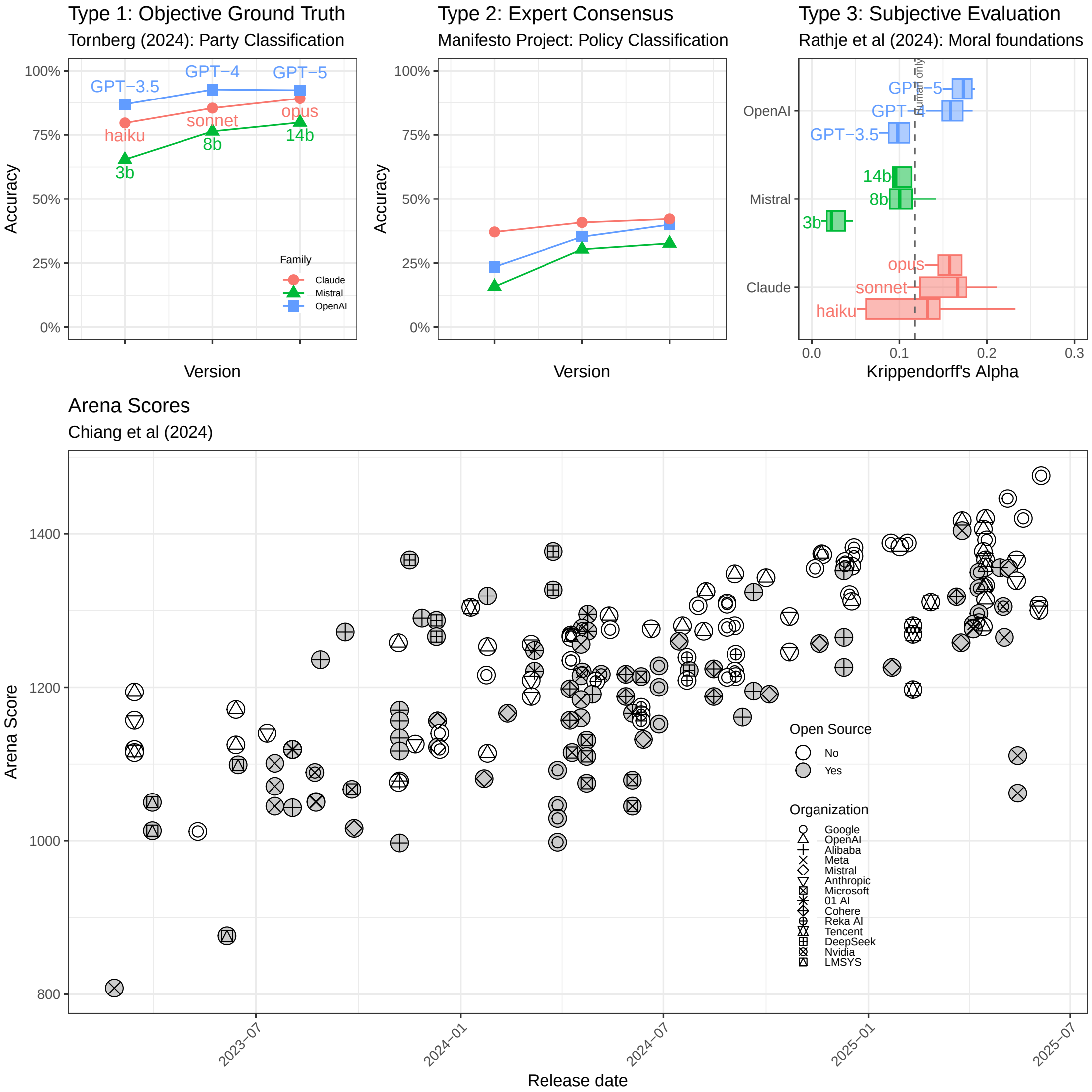

Figure 2: *Top row, left and center panels:* Average accuracy (y-axis) of 34 different large language models by version (x-axis and labels) and family (colors and shapes). Accuracy is defined as the proportion of politicians whose partisanship was correctly coded on the basis of 10 tweets (top-left panel, Törnberg 2024), or the proportion of manifesto sentences correctly assigned to a policy topic category (top-center panel, Lehmann et al. 2024). *Top row, right panel:* Krippendorff's alpha distributions (x-axis) calculated by randomly replacing one human coder with an LLM annotator 1,000 times, disaggregated by version (labels) and family (y-axis and colors). Intercoder reliability associated with annotating social media posts according to moral foundations codebook (Rathje et al., 2024). *Bottom row:* Arena scores (y-axis) for 204 large language models (points) by release date (x-axis), open source status (colors) and organization (shapes). Arena scores are based on a combination of human and GPT 4 annotators presented with two model outputs to the same prompt and asked to evaluate which performs better. Details can be found in Chiang et al. (2024).

panel, where the Arena scores overcome issues with ceiling effects. Here, the y-axis is a score derived from hundreds of pairwise comparisons across a range of tasks in which a human (or in some cases, a different LLM) simply votes on which model performed best. Each new model is, on average, improving over its predecessors in a roughly linear, clearly monotonic fashion.

Third, in Type 3 classification tasks (and potentially Type 2 tasks), it makes more sense to evaluate LLM performance not with "accuracy" per se, but rather with intercoder reliability. To operationalize this concept, we implement a bootstrapped analysis in which we start with the original human annotations and measure intercoder reliability via Krippendorff's alpha (vertical dashed gray line). We then randomly replace one of the human coders with one of the LLMs, and recalculate Krippendorff's alpha. We repeat this process 1,000 times, generating a distribution of intercoder reliability measures against which to compare to the original human-derived value. As illustrated, LLMs exhibit increasingly strong ICR values with more advanced models, exceeding that of the human (furthermore, *a propos* of the preceding discussion, the notion of a human "gold standard" seems dubious here, as a Krippendorff's alpha value of 0.118 is very low). We include a more thorough description of our replication analysis, as well as a more comprehensive review of recent research using LLM-assisted annotation, in the online Appendix.

2.3 When Humans are No Longer the Gold Standard

Recent experience has suggested that even if a gold standard does exist it might no longer be human, or at least not uniquely so. There are several ways this can happen. First, human performance need not be the upper bound on a given task (Bowman, 2024). In the context of text analysis it is clear that LLMs can outperform crowdworkers (e.g. Gilardi, Alizadeh and Kubli, 2023), and perform at a level approaching or identical to experts for certain matters (Bosley, 2024; Heseltine and Clemm von Hohenberg, 2024; Wu, 2025). Indeed they may surpass experts for at least some tasks where the ground truth is objective and known (e.g. Törnberg, 2024). Otherwise put, even if one recruits sufficient human talent to provide a baseline comparison, it is not obvious that matching those judgments is the correct metric for assessing performance (Hohenwalde et al., 2025, make this point explicitly).

Second, even when human experts could still be the foundation of a gold standard, there may be reasons not to expect the existence of a full or extensive validation set. Some data is in principle amenable to coding, but as a practical matter is difficult to work with. It would be unsurprising in such circumstances if humans were to spot-check a few cases but otherwise allow the LLM to code as it saw fit—especially if it had shown high accuracy previously on similar tasks. Some complex event data has this form (see Halterman et al., 2023)—it can be prohibitively expensive to put together validation sets, and "trust but verify" is a rational strategy. In other work, the data is sufficiently numerous and time-consuming to read that using an LLM as a first cut is sensible when combined with some expert judgement after the fact (e.g. Fang, Li and Lu, 2025; Wu, 2025). A related idea is that if an LLM is sufficiently trusted, then it can evaluate other LLMs on behalf of human researchers (see Bai et al., 2023).

Finally, we have type 3 situations where there is no objective standard but we wish nonetheless to comment on the quality of outputs. In this context, it makes little sense to talk of a "gold standard" at all. For instance, Xu et al. (2024) consider the case where one wants to grade creative student projects, but there is no *a priori* right answer. For these reasons, we assert that the notion of a human gold standard may not exist in some situations of interest to applied researchers.

2.4 What does this mean for applied research?

Based on this evidence, we believe that many LLM-powered annotation cases are such that the current models are already performing well—perhaps on par with humans—and we face a future where SOTA model improvements will be modest, constrained by an upper bound of 100% accuracy. What are the implications of these facts?

First, if there is no human gold standard, it will make little sense to attempt to "correct" estimates predicated on there being one (cf Fong and Tyler, 2021; Zhang, 2021; TeBlunthuis, Hase and Chan, 2024; Egami et al., 2023). But even if there is still a plausible human gold standard, we advocate for thinking more seriously about the robustness of findings under current LLMs as new (better) ones come online. Second, to the extent we think improvements will be generally modest and bounded above, we should perhaps not be overly concerned about using SOTA models or worrying that our results are sensitive to those choices. This has implications for both how robust we think current research findings might be, and how much we trust estimates from the past.

Of course, these empirical claims depend on the specific scenarios in which an inferior model is used. So we now focus on a challenging subset of the measurement error methodological literature and use simulations calibrated to real data. Specifically, we build on a well-known and long-standing literature in econometrics and political methodology: estimating (average) treatment effects when the treatment is binary and subject to measurement error. This focus is for ease of exposition, and we demonstrate that our robustness framework works equally well for binary outcomes, and for multi-class variables, in extensions.[5]

3 Focus Problem: ATE when treatment is (mis)classified

In practice, researchers might be interested in using an LLM to classify a variable that would be used as either an outcome or a predictor in some downstream regression analysis. Our sensitivity analysis can handle both use-cases, but we focus here on the harder scenario where the misclassified variable is on the right-hand side of a regression equation.

[5] We do not address continuous-valued variables; these have been shown to be less amenable to LLM-assisted annotation for now (Geng et al., 2024).

To set up the problem, we follow the presentation of Nguimkeu, Rosenman and Tennekoon (2021) with minor changes. We assume that the researcher is interested in a regression model of the following form:

$$Y_i = c + \beta D_i^* + X_i'\gamma + \epsilon_i \tag{1}$$

Here, Y_i is some value of a continuous outcome variable, like a vote share in an election. X_i is a set of predictors which could be continuously or otherwise valued for the ith observation. This might be the candidate's age and/or gender and/or incumbency status and/or money raised in the current election cycle. D_i^* is binary regressor or "dummy variable" that can take one of two values for a given observation, $D_i^* \in \{0, 1\}$. This might be whether the politician's manifesto was 'populist' (or not) or was 'liberal' (or conservative). The error term, ϵ_i is uncorrelated with any of the variables (D^* or the Xs) and has mean zero. We want to estimate the coefficients c (the constant), β and γ. Researchers focus on β because, assuming *inter alia* that conditional ignorability holds and the Xs are all correctly measured, it is the effect of the treatment on the outcome. But what makes the problem more complicated here is that D^* is not observed. That is, there exist some D (the observed measure, also a binary variable) that is sometimes misclassified, such that there is—for at least some units—a discrepancy between the measured value D_i and the true value D_i^*. For example, the problem could be that deciding whether a manifesto is populist is hard to get right. Put otherwise: there is some kind of misclassification of D_i^*, perhaps via a language model or human coding, that results in the realized D_i^* not being identical to the underlying truth.

The assumption about the relationship between the truth (D^*) and what we have in the data set, D, is

$$D_i = D_i^* + U_i. \tag{2}$$

Here U_i is a measurement error term. For any specific observation we are trying to classify, U_i can take one of three values. First, it can be 0, meaning no error. Second, when the true

D_i^* is 0, the error can be 1. Finally, U_i can be -1 when the true D_i^* is 1.

For reasons that will become clear, following the presentation in Meyer and Mittag (2017), we say there is a binary random variable M which takes the value 1 if the ith unit is misclassified when the treatment is recorded. That is,

$$m_i = \begin{cases} 0 \text{ if } D_i^* = D_i \\ 1 \text{ if } D_i^* \neq D_i \end{cases}$$

The proportion of all the N total observations for which $D \neq D^*$—the misclassification rate—is just the mean of M which we write as $R = \frac{\sum_{i=1}^{N} m_i}{N}$

This is a measurement error problem, but not of the 'classical' form. That is, the error here is not random and it is not unrelated to the true value of the variable (D^*) in question. Indeed, if there is error, it is perfectly negatively correlated with the value of D^*. Furthermore, it is possible that this error might also be "differential", meaning that M is not independent with respect to Y or other parts of the data generating process. Finally, it is possible that the controls $\mathbf{X}$ might correlated with the true treatment D^*, meaning that they must also be correlated with the error term U_i, thereby introducing a "hidden bias" (Nguimkeu, Rosenman and Tennekoon, 2021).

Even if one assumes away differential measurement error and correlated controls, a researcher who wants to evaluate the ATE of D^* but only has access to D will find that the ATE of interest is attenuated. That is, β is estimated to be closer to zero than the truth. The direction of the bias is ambiguous (thus worse) in the case where these other problems are not assumed away.

Mapping the preceding discussion onto our substantive focus on state of the art large language models, we will define **best performance** as that which has the smallest proportion of misclassifications. That is, the best performance is the one that minimizes the misclassification rate R as defined above. Equivalently, the best performance—which we will denote as R_{SOTA}—is the one that has the smallest value of R. The model that yields this performance is demarcated as **state of the art**.

Clearly, R_{SOTA} need not take the value 0 for it to be state of the art. And the numerator of the fraction makes no weighted distinction between a false positive or false negative. Moreover, there are many constellations of values that could yield a minimum for the fraction, such that it is possible that multiple different models could all be SOTA despite the fact they give somewhat different realizations of M. We return to this idea later, when we shift our focus from *mis*classification to *re*-classification. Nevertheless, the real quantity of interest for our discussion is the difference between R_{SOTA} and R_{LLM}, where the latter is misclassification rate of whatever LLM we happened to use.

3.1 A simulated example

We now return to the motivating set of questions from S2.4 which can be summarized into the general "how much should we worry about SOTA changing?" To fix ideas, consider the performance of GPT-4 from Figure 2 above. We find that, in the context of US politicians, the March 1st, 2025 version of GPT 4.0 outperforms the November 20, 2024 version by 1 percentage point, increasing from 92.6% to 93.6% in accuracy terms. A reasonable instinct would be that this is unlikely to dramatically alter the substantive conclusions drawn from a downstream regression using annotations generated by the older vintage of the model. But measurement error involves multiple moving pieces; it is not just the number of observations which are misclassified (sum of M) that matters, but also the underlying distribution of 1's and 0's (the skew of the treatment), as well as the degree to which the misclassified observations are correlated with other parts of the data generating process (differential measurement error and correlated controls).

To demonstrate, we simulate a data generating process in which an outcome Y is a linear combination of a binary treatment D and continuous covariates X_1 and X_2, expressed as

$Y = c + \beta D + \gamma_1 X_1 + \gamma_2 X_2 + \varepsilon$. As above, we denote misclassification with a vector M that takes on the value $m_i = 0$ if the ith observation is accurately classified or $m_i = 1$ if not. We explore the attenuation bias associated with different values of the following dimensions of interest:

- The misclassification rate $R = \text{mean}(M)$ (i.e., how many observations are misclassified, or 1 - accuracy).
- The skew in the binary treatment variable $\pi = \text{mean}(D)$ (i.e., how many 1's and 0's make up the treatment vector).
- The extent to which the measurement error and the outcome are correlated $\rho_{M,Y}$ (i.e., how much differential measurement error exists).
- The variance-covariance matrix, Σ, describing the relationship between D and the controls (i.e., the correlation between the treatment and the controls).

The full simulation results are in the online appendix, where we find a useful hierarchy in how much each dimension biases $\hat{\beta}$ in a downstream regression. Here, we vary the three most influential dimensions: the misclassification rate R, the skew π, and the differential measurement error $\rho_{M,Y}$. The first two dimensions produce attenuation bias, whereas the third can produce bias in either direction, potentially causing coefficients to cross the null.

Figure 3 visualizes the problem with estimating a coefficient based on data annotated by an inferior LLM ($\hat{\beta}_{t1}$ where $t1$ captures the idea that the LLM might have been SOTA initially, but has since been surpassed), and what might happen to this estimate if it was re-calculated on data annotated by a current state-of-the-art LLM ($\hat{\beta}_{t2}$). We denote the misclassification rate R on the x-axes and the skew in the annotated treatment vector π on the y-axes. Dashed contour plots chart the terrain of the ATE estimates of interest, denoted by a solid black circle. Building on our claims in Section 2.2 above, we assume that the future

LLM will improve on the current (in other words, $R_{t2} < R_{t1}$), we move leftward across the x-axis. What these future models might yield in terms of π is unknown, so we allow the skew in the updated treatment vector D to vary arbitrarily.[6] The updated coefficient estimates are indicated by a white circle, with an arrow highlighting the movement between the current ($t1$) and future ($t2$) models.

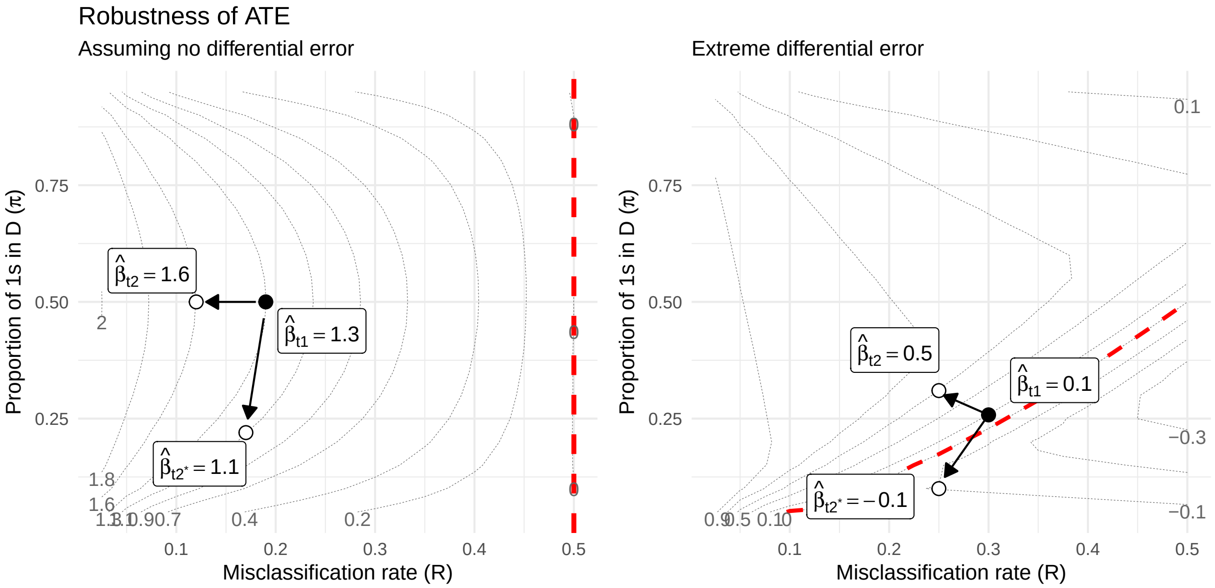

Figure 3: Monte Carlo simulation results demonstrating sensitivity of coefficients (points) due to misclassification rates (x-axes), skew in binary variable being coded (y-axes) assuming changes along both dimensions when comparing today's LLM ($\hat{\beta}_{t1}$, black circles) to tomorrow's ($\hat{\beta}_{t2}$, white circles). Left-panel displays sensitivity assuming no differential measurement error ($\rho_{M,Y} = 0$) while right-panel displays sensitivity assuming large differential measurement error ($\rho_{M,Y} = 0.7$).

The figures highlight the importance of skew in the underlying treatment variable, as well as the relative robustness of ATEs, assuming that tomorrow's LLM reduces the misclassification rate (x-axes). Assuming no differential measurement error (left panel of Figure 3), the reduction in the misclassification rate (x-axis) would need to be very modest and the change in skew (y-axis) very extreme such that using the newest SOTA LLM would reduce

[6] The range of possible values of π_{future} under the assumption that $R_{\text{future}} < R_{\text{current}}$ are constrained only by R_{current}, since it is possible in the extreme to flip all incorrectly coded zeros to ones, while flipping all but one correctly coded ones to zeros.

the strength of the ATE ($\hat{\beta}_{t2^*}$). But if the skew doesn't change, newer models should only strengthen the statistical and substantive strength of the ATE ($\hat{\beta}_{t2}$). Even when the differential error is extreme (right panel of Figure 3), ATE estimates would not flip sign unless the treatment distribution is highly skewed and grows even more extreme ($\hat{\beta}_{t2^*}$). In sum then, simulated data suggests that better annotations should typically strengthen conclusions, even where the underlying assumptions are relaxed.

But how unlikely is it that a new model could produce shifts in both R and π to weaken an ATE? We return to the replication data introduced above, and plot the change in both the misclassification rate and the skew for every model compared to all newer versions, using the (Type 1) partisanship annotation task from Törnberg (2024), where the ground truth is an objective fact. Figure 4 plots the difference in misclassification rate (x-axes) and skew (y-axes) when moving from an older model (columns) to a newer model (rows). Arrows indicate movement, with red arrows highlighting an increase in the misclassification rate, while black arrows indicate a reduction in the same. While there are a handful of contexts in which tomorrow's model actually performs worse than today's (especially when moving from a model in OpenAI's "turbo" family to one of their cheaper "mini" models), and a handful of contexts in which the skew of the updated D variable shifts dramatically, the vast majority of calibration data lie within the curves of the contour lines. Although not a comprehensive summary of all published social science work that uses LLMs for annotation, the evidence here indicates that conclusions would be weakened with a new model in only 13.4% of cases, a proportion that drops to 8% if we ignore the inexpensive "mini" family of LLMs. In none of the calibration data would an ATE be overturned in terms of sign, even were we to assume extreme differential error.

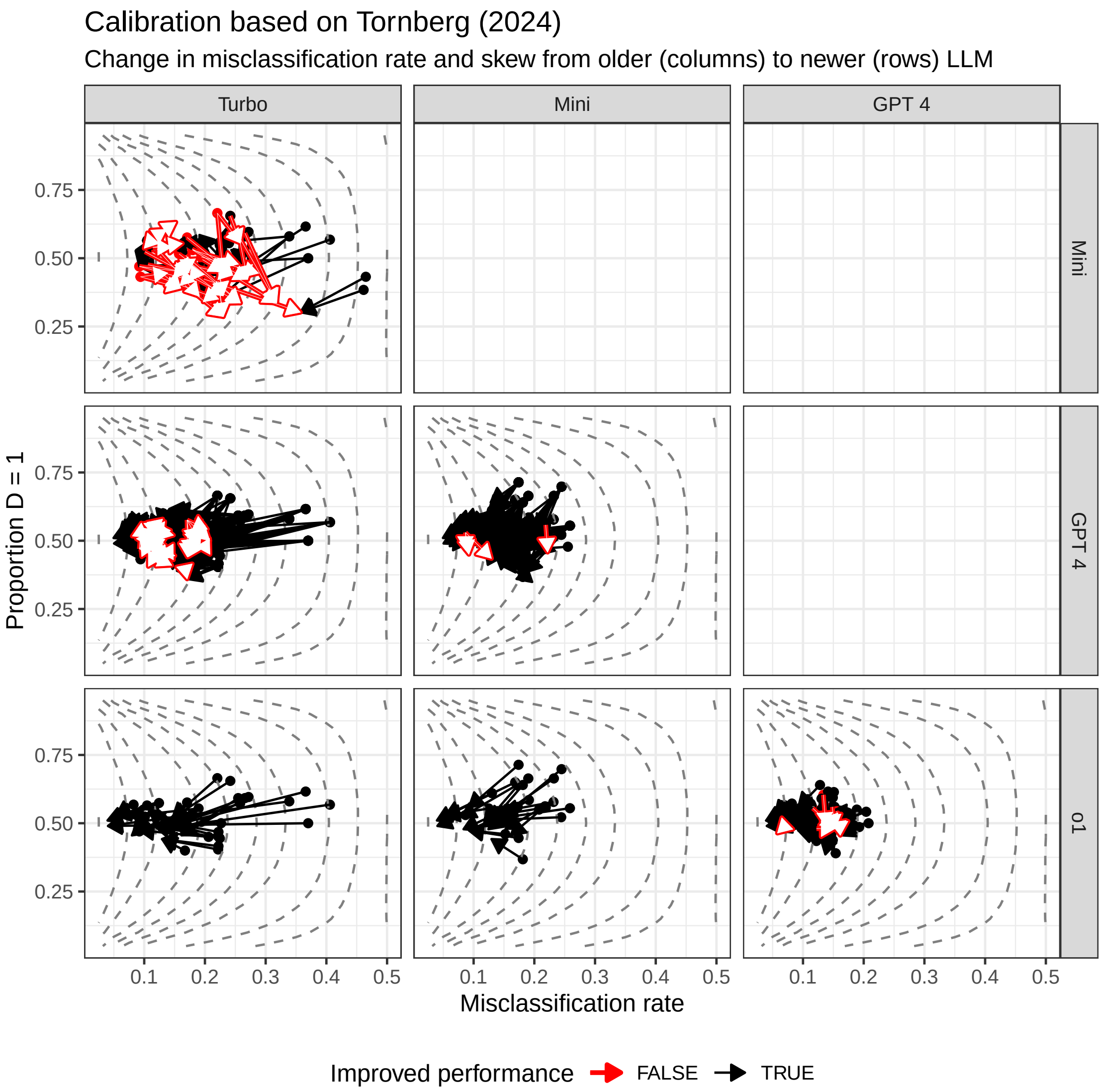

Figure 4: Calibration evidence based on Törnberg (2024). Each solid black point represents an older model's performance in predicting the partisanship of a politician based on their tweets in terms of misclassification rate (x-axes) and imbalance in the binary partisanship vector (y-axes). Each hollow white point represents a newer model's performance in the same task. Black arrows highlight improvements in accuracy (reductions in the misclassification rate, x-axes). Red arrows highlight reductions in performance when moving to a new LLM. Dashed gray contour lines summarize attenuation bias from misclassification, assuming no differential error.

In sum then, simulated evidence calibrated to real-world values suggest that ATE estimates of interest will be mostly insulated from changes in the annotation method, at least as far as sign and substantive meaning are concerned. The differential measurement error would need to be large, the improvement in LLM performance small, and the shift in skew substantial in order for tomorrow's estimate of an ATE to be pushed toward zero, relative to today's. The specific value of the point estimate is less protected, although we argue that the majority of social scientific research interested in ATEs are primarily concerned with evaluating an alternative hypothesis against a null of no relationship. We include an extensive set of simulated calibrations in the online appendix, along with a wholly different data generating process borrowed from Egami et al. (2023). In general, for misclassification rates below 20%, and for changes in the same of less than 15 percentage points, changes in the observed treatment vector D stemming from a new SOTA LLM would need to be just-so to dramatically overturn empirical conclusions about coefficient signs and significance. Put differently, applied scholars need not be overly concerned about whether their conclusions would survive were their data to be re-coded by the current cutting edge AI. For the same reason, we think reviewers might pause before requiring manuscripts to regenerate all data with a newer LLM.

4 Robustness and sensitivity analysis for LLMs

The preceding analysis is based on simulated data in which a misclassification rate can be measured; this presupposes the existence of a gold standard against which to compare. But in a world without (human) gold standard data, or for annotations that fall under the type 2 or type 3 categories described above, how should an applied researcher deal with rapid technological change? For the field as a whole, the equivalent issue is how to assess the value of papers that—due to the length of our usual publication cycle—rely on non-SOTA LLMs to annotate data. These questions become even more pressing if our core assertion that LLMs are getting better does not hold.

We argue that sensitivity analyses can help us understand how fragile our findings are, or how reliable our published results are, in the face of changing annotations by some future LLM. To do so, we move away from our preceding focus on *misclassification*, and pivot to the related but more general and agnostic *reclassification.* In other words, we drop the assumption that tomorrow's LLM will be strictly *better* than today's and instead only assert that its annotations might be *different.* We then examine how brittle today's ATE estimates are when we know that some proportion of the observed treatment vector D might be reclassified tomorrow.

Our approach builds on methods originally developed by Blackwell (2014) and others, which were then generalized by Duarte et al. (2024). We calculate three sets of bounds subject to a proportion of the data which could be reclassified. The first set of bounds are **extreme bounds**, for which we identify the specific observations for which flipping the observed treatment values would maximize or minimize the ATE (Duarte et al., 2024). Specifically, for each assumed misclassification rate $R \in \{0.01, 0.02, \ldots, 0.5\}$, we identify the subset of observations whose treatment assignment—if flipped—would most strongly increase or decrease the estimated treatment coefficient. To accommodate covariates, we first residualize both the outcome and treatment on the set of control variables using linear regression. We then greedily identify the $R \times N$ observations for which flipping the residualized treatment variable would most increase (or decrease) the mean difference in residualized outcomes between treated and control groups. These flipped values are used to re-estimate the full regression model, and the resulting coefficients are recorded as the "extreme" upper and lower bounds for each misclassification rate.

Practically, denote M with typical value $m_i \in \{0, 1\}$ to be an indicator variable for whether an observation is reclassified and denote $\tilde{Y}$ to be the residualized values of Y from the regression of $Y = \mathbf{X}\beta$. To calculate the extreme upper bound on $\hat{\beta}_{\text{extreme}}$, we identify the largest $\tilde{Y}$ values where $D = 0$ and the smallest $\tilde{Y}$ values where $D = 1$ and then proceed down the list, flipping D until we have hit the limit of $R \times N$ observations to be reclassified.

This ensures that we maximize any potential skew in the outcome variable, but also could result in only $D = 0$ being flipped to $D = 1$ if all the largest values of $\tilde{Y}$ are associated with $D = 0$ observations. The extreme lower bound follows the identical procedure, except that we reverse the $D = 0$ and $D = 1$ ordering of $\tilde{Y}$. The resulting $\hat{\beta}_{\text{extreme}}$ estimates thus reflect a highly unlikely extreme bound on the furthest β might range from the observed $\hat{\beta}$. The intuition of this approach is visualized in Figure 5 which shows a normal (jittered) representation of the data in the left panel, a sorted visualization of the same data in the center panel, with the largest values of $D = 1$ and smallest values of $D = 0$ highlighted in gray diamons, and then the new difference in means when these observations are reclassified to $D = 0$ and $D = 1$ respectively, flipping the sign of the observed coefficient from positive to negative.

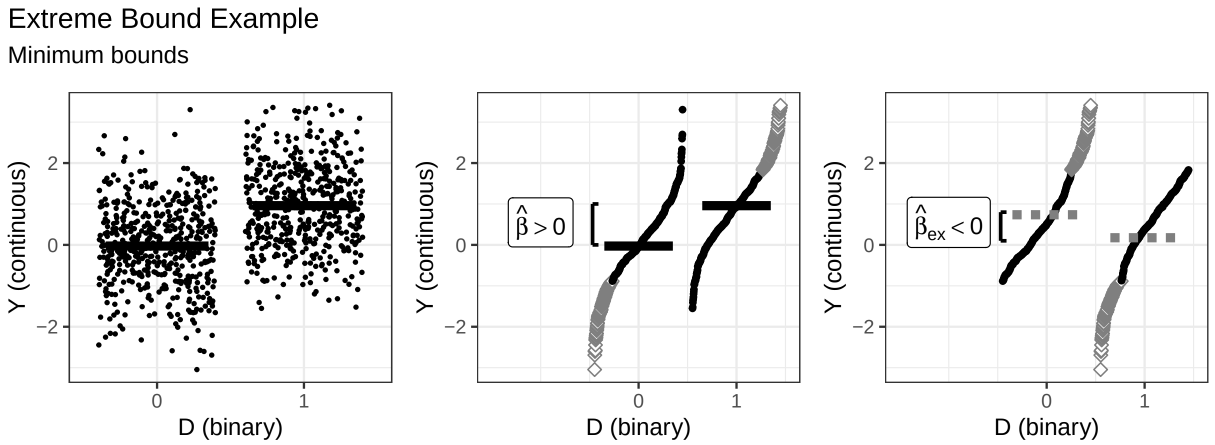

Figure 5: Demonstration of extreme minimum bounds obtained by sorting data based on outcome value (or residualized values thereof when there are controls), selecting the observations with the largest values of $D = 1$ and the smallest values of $D = 0$ in order to produce the smallest possible coefficient. Flipping these observations yields an estimate of $\hat{\beta}$ which flips sign.

The preceding approach is fast and will perfectly recover the most extreme estimates of the coefficient when there are equal numbers of $D = 1$ and $D = 0$ in the data. However, if the data is skewed, then a more extreme coefficient is possible if we flip more observations for which ever treatment status ($D = 0$ or $D = 1$) is rarer. In other words, if only 25%

of the observations have $D = 1$ initially, flipping more of these observations will produce a bigger "bang for the buck" since the updated average will move further. In practice, we look across all possible splits of $R \times N$ observations between $D = 0$ and $D = 1$ and choose the split which produces the most extreme estimate of the ATE. A detailed description of this algorithm is included in the online Appendix, Section C.

The second set of bounds are **naïve bounds** in which we select the observations at random (denote the indicator variable as M_{naive}) to then flip the observed treatment and re-estimate the ATE of interest. We repeat this process 100 times and plot the top and bottom 5% of these observations. These bounds are "naïve" in the sense that they show what the researcher could expect, on average, if they know their treatment is wrongly coded but not in any particular way.

The third set consists of **permutation bounds** which characterize how bad an extreme bound might be if some proportion of M_{extreme} was randomly flipped. That is, we calculate extreme bounds for a given proportion of the data reclassified, and then randomly select 20% of the observations which were reclassified in the extreme setting, and replace them with a random selection of other observations, the indicator variable for which we denote M_{perm}. We repeat this process 100 times and take the average of the resulting vector of coefficient estimates. These bounds help the researcher understand how sensitive the extreme bound is. Recall that the extreme bound is a "just-so" set of observations which uniquely maximize (or minimize) the estimated coefficient. If modest permutations of the set of observations chosen produce far less extreme coefficient estimates, we would conclude that our model and data are relatively insulated from reclassification concerns.[7]

Examples of the sensitivity analysis are presented in Figure 6 which uses a combination

[7] While it has the flavor of an inferential statement, the permutation bounds are better understood as a measure of how sensitive the extreme bound is to modest permutations of the M_{extreme} vector. As such, we choose a threshold of 80-20: if 80% of the observations required to generate the extreme bound were still flipped, but the other 20% were chosen at random, how bad could it be?

of simulated data and two real-world datasets, based on replication material from Grumbach and Sahn (2020) and Heseltine and Clemm von Hohenberg (2024). The simulated data uses the same data generating process described above, while the latter two datasets demonstrate the usefulness when the binary variable appears on the right hand (Grumbach and Sahn, 2020) or left hand (Heseltine and Clemm von Hohenberg, 2024) sides of a downstream regression equation. The replication material from Grumbach and Sahn (2020) predicts the share of contributions from white Republican donors as a function of the recipient candidate's ethnorace. Their predictor variable is based on traditional methods of annotating the ethnorace of a politician as a function of their name and geolocation (Imai and Khanna, 2016). The replication material from Heseltine and Clemm von Hohenberg (2024) predicts the share of negative tweets as a function of politician characteristics, where the outcome of interest is based on an LLM-assisted annotation process. In this setting, we demonstrate the usefulness of the sensitivity analysis to settings where the LLM-labeled variable is on the left-hand side (the outcome) of the regression equation.[8]

The core concern for the applied researcher is where different bounds cross the null (zero on the y-axis) along the x-axis. Consider again the applied researcher contemplating rerunning analyses using a more up-to-date model for coding their treatment variable than the non-frontier LLM they actually used. For them, the point on the x-axis where each set of bounds crosses the null are a useful reference. In the simulated data, the relevant thresholds are a 12% reclassification rate for the extreme bounds, a 15% reclassification rate for the permutation bounds, and more than a 40% reclassification rate for the naïve bounds. In this case, we would argue that the researcher should be reasonably confident in the robustness of their results. This is because—as an empirical matter—it is unlikely that updating the coded data with the most cutting-edge LLM will reclassify as much as 12% of the existing

[8] The specifications presented below are based only on the replication data provided by the original authors, but do not replicate their specific findings *per se*. We use these data simply to demonstrate sensitivity analysis in a real setting.

treatment vector, let alone 40%. Conversely, more care should be taken with the replication data from Grumbach and Sahn (2020), where the extreme bounds cross zero if even only 1% of the data were reclassified. That said, the naïve simulated bounds don't cross the null until more than 10% of the data is reclassified. Similarly, the finding that Republicans tend to tweet more negatively found in Heseltine and Clemm von Hohenberg (2024) is robust up to 10% of the observations being reclassified in the permutation bound setting.

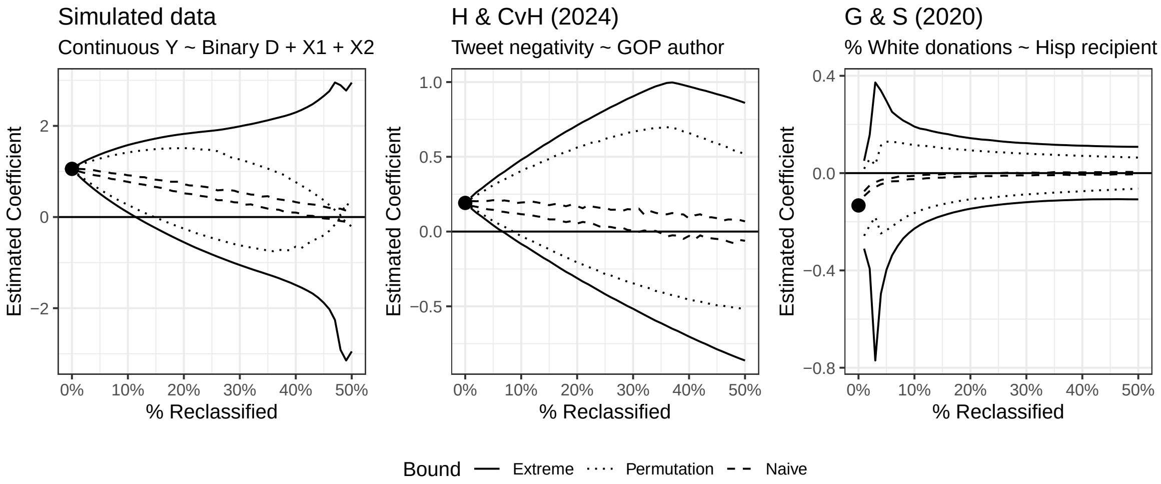

Figure 6: Sensitivity analytic results for simulated data (left panel), replication data from Heseltine and Clemm von Hohenberg (2024) where the binary variable of interest is on the left-hand side of the regression equation (center panel), and replication data from Grumbach and Sahn (2020) where the binary variable appears on the right-hand side of the regression (right panel). Solid lines indicate extreme values that the estimated ATE could take on given the proportion of reclassified observations (x-axes). Dotted lines indicate the extremity of the extreme bound if 20% of the observations were randomly permuted. Dashed lines indicate the 95% confidence interval given the proportion of reclassified observations chosen at random.

4.1 Extension: Multi-class Categorical Annotations

While our main analysis focuses on binary treatment variables, many important political science applications involve multi-class categorical variables, such as policy topics, ideological positions, and so on. Extending the sensitivity analytic-framework introduced above in the binary variable case to multi-class K-categorical variables, where $K \geq 2$, is conceptually straightforward, although computationally challenging as K grows. The crucial component to recognize is that a K-categorical treatment variable $D^* \in \{0, 1, 2, \ldots, K-1\}$ is translated into a $K-1$ set of binary variables in a downstream regression of the form:

$$Y_i = c + \sum_{j=1}^{K-1} \beta_j \cdot \mathbb{I}(D_i^* = j) + \mathbf{X}_i'\boldsymbol{\gamma} + \epsilon_i \tag{3}$$

We now have $K-1$ coefficients of interest: $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_{K-1})'$ where the reference category or "hold out" is absorbed in the constant c.

Unlike in the binary case where misclassification will attenuate estimates of interest, with a K-categorical variable a single misclassification in D induces two simultaneous indicator errors (a "false negative" in one category and a "false positive" in another), so the induced measurement error is correlated across the indicator regressors. Translating this intuition into a sensitivity analytic framework is theoretically straightforward, although computationally intensive as K increases. For a given coefficient of interest, the core intuition remains the same: identify the observations for which switching $\mathbb{I}(D_i = j)$ maximizes (minimizes) the associated coefficient estimate $\hat{\beta}_j$. Using the maximum extreme bound as an example, this entails identifying the observations for which $\mathbb{I}(D_i = j) = 0$ and $\tilde{Y}_i$ is large, and for which $\mathbb{I}(D_i = j) = 1$ and $\tilde{Y}_i$ is small, where $\tilde{Y}$ is the residualized value of Y. As above, the mask vector identifying these observations is denoted M_{extreme}.

However, swapping $\mathbb{I}(D_i = j) = 0$ to a 1 requires swapping some other $\mathbb{I}(D_i \neq j) = 1$ to a zero. This will impact the extreme $\hat{\beta}_j$ of interest in one of two related ways. First, if we "borrow" the swap from the reference category (or "hold-out"), this will change the composition of the comparison group. Second, if we "borrow" the swap from some other category, this will change the residualized $\tilde{Y}$ values as per the Frisch-Waugh-Lovell theorem.

Theoretically, it is possible to iterate through all possible observations, swapping both the $\mathbb{I}(D_i = j)$ value of interest and the required donor category. This greedy deterministic algorithm will identify M_{extreme} and produce both minimum and maximum bounds for a given $\hat{\beta}_j$ of interest, subject to some proportion of the data to be reclassified R. However, as K grows, the deterministic algorithm grows computationally intractable. As such, we also describe a probabilistic solution in the online appendix. Finally, as above, the permutation bounds are simply a mixture of the extreme bounds identified here and the naïve bounds.

5 Advice to Practitioners

We now summarize our arguments as advice to the applied research community.

1. Think carefully about whether the validation data is truly a gold standard. If you are confident it is, then bias correction methods (i.e., Egami et al. 2023; TeBlunthuis, Hase and Chan 2024) can be fruitfully applied. But if the validation data is based on human annotators with modest to poor reliability, sensitivity analysis may be more useful.

2. Though LLM performance matters, for a given inference problem other factors do too: in the case we discussed, this includes skew and differential misclassification. The way those quantities interact with coding performance is also worthy of attention.

3. Not using a "State of the Art" large language model need not be fatal, and work should not be dismissed for this reason; indeed, signs and significance of coefficients may become *more* not less pronounced as technology improves. We are skeptical "State of the Art" is a good enough reason to use proprietary rather than open weight models for this reason.

4. Rather than seeking (onerous) reanalysis of data with a more modern LLM, reviewers should request sensitivity analysis—and authors can use our software to produce that.

6 Software

We provide an `R` package, `futureProofR`, for conducting sensitivity analysis of regression coefficients when binary or categorical variables are subject to misclassification. The package implements fast, transparent algorithms that compute worst- and best-case bounds on treatment effects by strategically reclassifying observations. Its core function, `misclass_sens()`, uses a greedy residual-based procedure to identify which observations to flip in order to maximize or minimize a target coefficient, accommodating settings where the misclassified variable is either the treatment or the outcome and offering multiple bound types, including extreme, differential, and naive bounds, along with permutation-based inference. For multinomial variables, `futureproofR` extends this logic via `multiclass_sens_greedy()`, a deterministic greedy algorithm that ranks category reassignments by their marginal impact on the coefficient, and `multiclass_sens_optim()`, an optimization-based approach suitable for larger category sets. The package integrates seamlessly with standard `lm` and `fixest` workflows and includes visualization tools for inspecting how estimates vary with assumed misclassification rates, allowing researchers to assess whether substantive conclusions remain robust to plausible levels of classification error in both human- and LLM-annotated data.

7 Discussion

The progress of LLMs is remarkable. Now it is not just that machines can approximate what humans can do, albeit faster; for some tasks at least, LLM performance is at or above human expert standard. This paper is about what happens to downstream inference problems when this is broadly true of coding and measurement problems. In general, we argued that the news is good: as misclassification rates fall, inferences should sharpen. That is, for the specific but common problem we studied, claims of statistically significant treatment effects should be preserved; indeed we might well expect the sizes of the relevant coefficients to get larger.

Our broader argument is that the way researchers think about annotation problems should shift. Rather than trying to “correct” annotations back to a (potentially dubious) human standard, we should focus instead on what our findings look like as models get better and annotations change. The general idea of thinking seriously about sensitivity and robustness of coding decisions is not new to us, but the speed of LLM change induces new applications and urgency. Based on earlier efforts, we provided some new techniques for this purpose.

There are broader, meta-science lessons from our work. We think it is easy to overemphasize the importance of “State of the Art” models when conducting or reviewing research. This matters a great deal as particular LLMs are replaced and/or deprecated quickly over time. Models often cease to be frontier in a period shorter than the usual publication cycle in political science, and this potentially means everything we write is “out of date” as soon as it is published. On the empirical question, we showed that findings using non-frontier LLMs are perhaps not as vulnerable to technological change as initially thought. This may mean that appeals to State of the Art are not necessarily a compelling reason to prefer (newer, more expensive) proprietary models over open weight versionable ones.

Of course, there are limitations to our efforts. First, there will likely always be questions where a human gold standard exists and may far surpass any LLM efforts. But even so, we would argue robustness is still a helpful frame in such situations, and a clarifying partner to the bias-correction methods that rely on a human gold standard (e.g. Egami et al., 2023; TeBlunthuis, Hase and Chan, 2024).

Second, our analysis was limited to one particular use-case of the linear model. Many other problems do not have this exact form. More broadly, LLM “interaction”—rather than annotation—may be the treatment of interest in, say, experimental work.

Third, we do not deny that there are other benefits to research that come from exploring the frontier of this exciting new technology. In particular, agentic AI can be used to explore unstructured data and generate summaries and labels that go beyond the traditional annotation workflow discussed here.

But to reiterate we believe the core assertion that non-SOTA need not be disqualifying for high quality social science research is worth emphasizing, particularly in light of other considerations that might be discounted, such as the importance of reproducible research (Palmer, Smith and Spirling, 2024). We leave the other related questions for future work.

References

Aigner, Dennis. 1973. "Regression with a binary independent variable subject to errors of observation." *Journal of Econometrics* 1(1):49–59.

Argyle, Lisa P, Ethan C Busby, Nancy Fulda, Joshua R Gubler, Christopher Rytting and David Wingate. 2023. "Out of one, many: Using language models to simulate human samples." *Political Analysis* 31(3):337–351.

Bai, Yushi, Jiahao Ying, Yixin Cao, Xin Lv, Yuze He, Xiaozhi Wang, Jifan Yu, Kaisheng Zeng, Yijia Xiao, Haozhe Lyu et al. 2023. "Benchmarking foundation models with language-model-as-an-examiner." *Advances in Neural Information Processing Systems* 36:78142–78167.

Benoit, Kenneth, Drew Conway, Benjamin E Lauderdale, Michael Laver and Slava Mikhaylov. 2016. "Crowd-sourced text analysis: Reproducible and agile production of political data." *American Political Science Review* 110(2):278–295.

Bisbee, James, Joshua D Clinton, Cassy Dorff, Brenton Kenkel and Jennifer M Larson. 2024. "Synthetic replacements for human survey data? the perils of large language models." *Political Analysis* 32(4):401–416.

Blackwell, Matthew. 2014. "A selection bias approach to sensitivity analysis for causal effects." *Political Analysis* 22(2):169–182.

Bojić, Ljubiša, Olga Zagovora, Asta Zelenkauskaite, Vuk Vuković, Milan Čabarkapa, Selma Veseljević Jerković and Ana Jovančević. 2025. "Comparing large Language models and human annotators in latent content analysis of sentiment, political leaning, emotional intensity and sarcasm." *Scientific reports* 15(1):11477.

Bollinger, Christopher R. 1996. "Bounding mean regressions when a binary regressor is mismeasured." *Journal of Econometrics* 73(2):387–399.

Bosley, Mitchell. 2024. Towards Qualitative Measurement at Scale: A Prompt-Engineering Framework for Large-Scale Analysis of Deliberative Quality in Parliamentary Debates. Technical report Working Paper.

Bowman, Samuel R. 2024. "Eight things to know about large language models." *Critical AI* 2(2).

Card, David. 1996. "The effect of unions on the structure of wages: A longitudinal analysis." *Econometrica: journal of the Econometric Society* pp. 957–979.

Chiang, Wei-Lin, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez et al. 2024. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.

Choi, Dahyun, Denis Peskoff and Brandon Stewart. 2025. "Fine-tuned Large Language Models Can Replicate Expert Coding Better than Trained Coders: A Study on Informative Signals Sent by Interest Groups.".

Coppedge, Michael, John Gerring, Staffan I. Lindberg, Svend-Erik Skaaning, Jan Teorell, David Altman, Michael Bernhard et al. 2011. "Conceptualizing and Measuring Democracy: A New Approach." *Political Science Research and Methods* 2(2):247–267.

Duarte, Guilherme, Noam Finkelstein, Dean Knox, Jonathan Mummolo and Ilya Shpitser. 2024. "An automated approach to causal inference in discrete settings." *Journal of the American Statistical Association* 119(547):1778–1793.

Egami, Naoki, Musashi Hinck, Brandon Stewart and Hanying Wei. 2023. "Using imperfect surrogates for downstream inference: Design-based supervised learning for social science applications of large language models." *Advances in Neural Information Processing Systems* 36:68589–68601.

Fang, Hanming, Ming Li and Guangli Lu. 2025. Decoding China's Industrial Policies. Technical report National Bureau of Economic Research.

Fong, Christian and Matthew Tyler. 2021. "Machine learning predictions as regression covariates." *Political Analysis* 29(4):467–484.

Geng, Jiahui, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. pp. 6577–6595.

Gilardi, Fabrizio, Meysam Alizadeh and Maël Kubli. 2023. "ChatGPT outperforms crowd workers for text-annotation tasks." *Proceedings of the National Academy of Sciences* 120(30):e2305016120.

Grumbach, Jacob M and Alexander Sahn. 2020. "Race and representation in campaign finance." *American Political Science Review* 114(1):206–221.

Gunes, Erkan and Christoffer Koch Florczak. 2025. "Replacing or enhancing the human coder? Multiclass classification of policy documents with large language models." *Journal of Computational Social Science* 8(2):31.

Halterman, Andrew, Philip A Schrodt, Andreas Beger, Benjamin E Bagozzi and Grace I Scarborough. 2023. "Creating custom event data without dictionaries: A bag-of-tricks." *arXiv preprint arXiv:2304.01331* .

Heseltine, Michael and Bernhard Clemm von Hohenberg. 2024. "Large language models as a substitute for human experts in annotating political text." *Research & Politics* 11(1):20531680241236239.

Hohenwalde, Clarissa, Melanie Leidecker-Sandmann, Nikolai Promies and Markus Lehmkuhl. 2025. "ChatGPT's potential for quantitative content analysis: categorizing actors in German news articles." *Journal of Science Communication* 24(2):A01.

Imai, Kosuke and Kabir Khanna. 2016. "Improving ecological inference by predicting individual ethnicity from voter registration records." *Political Analysis* 24(2):263–272.

Imai, Kosuke and Teppei Yamamoto. 2010. "Causal inference with differential measurement error: Nonparametric identification and sensitivity analysis." *American Journal of Political Science* 54(2):543–560.

Krippendorff, Klaus. 2018. *Content analysis: An introduction to its methodology.* Sage publications.

Lehmann, Pola, Simon Franzmann, Dalia Al-Gaddooa, Tobias Burst, Christoph Ivanusch, Sven Regel, Felix Riethmüller, Andrea Volkens, Bernhard Weßels and Lisa Zehnter. 2024. "The Manifesto Data Collection.".
URL: *https://doi.org/10.25522/manifesto.mpds.2024a*

Little, Andrew T and Anne Meng. 2024. "Measuring democratic backsliding." *PS: Political Science & Politics* 57(2):149–161.

Meyer, Bruce D and Nikolas Mittag. 2017. "Misclassification in binary choice models." *Journal of Econometrics* 200(2):295–311.

Mikhaylov, Slava, Michael Laver and Kenneth R Benoit. 2012. "Coder reliability and misclassification in the human coding of party manifestos." *Political analysis* 20(1):78–91.

Munger, Kevin. 2023. "Temporal validity as meta-science." *Research & Politics* 10(3):20531680231187271.

Nguimkeu, Pierre, Robert Rosenman and Vidhura Tennekoon. 2021. "Regression with a misclassified binary regressor: Correcting for the hidden bias.".

Ornstein, Joseph T. 2025. "Probabilistic record linkage using pretrained text embeddings." *Political Analysis* pp. 1–12.

Palmer, Alexis, Noah A Smith and Arthur Spirling. 2024. "Using proprietary language models in academic research requires explicit justification." *Nature Computational Science* 4(1):2–3.

Rathje, Steve, Dan-Mircea Mirea, Ilia Sucholutsky, Raja Marjieh, Claire E Robertson and Jay J Van Bavel. 2024. "GPT is an effective tool for multilingual psychological text analysis." *Proceedings of the National Academy of Sciences* 121(34):e2308950121.

Spinde, Timo, David Krieger, Manuel Plank and Bela Gipp. 2021. Towards a reliable ground-truth for biased language detection. In *2021 ACM/IEEE Joint Conference on Digital Libraries (JCDL).* IEEE pp. 324–325.

TeBlunthuis, Nathan, Valerie Hase and Chung-Hong Chan. 2024. "Misclassification in automated content analysis causes bias in regression. Can we fix it? Yes we can!" *Communication Methods and Measures* 18(3):278–299.

Törnberg, Petter. 2024. "Large language models outperform expert coders and supervised classifiers at annotating political social media messages." *Social Science Computer Review* p. 08944393241286471.

Wu, Patrick Y. 2025. "Large Language Models Can Be a Viable Substitute for Expert Political Surveys When a Shock Disrupts Traditional Measurement Approaches." *arXiv preprint arXiv:2506.06540* .

Xu, Shengwei, Yuxuan Lu, Grant Schoenebeck and Yuqing Kong. 2024. "Benchmarking LLMs' Judgments with No Gold Standard." *arXiv preprint arXiv:2411.07127* .

Zhang, Han. 2021. "How using machine learning classification as a variable in regression leads to attenuation bias and what to do about it.".

Online Only Supporting Information for *What to Do When Humans Are No Longer the Gold Standard*

Contents (Appendix)

A Replicating LLM-Assisted Annotation

Our assertion that LLMs are good and getting better relies on an analysis of three annotation tasks, falling into the 3-part annotation typology discussed in the paper. Here, we describe the analysis in more detail, as well as provide several additional results using other annotation tasks.

A.1 Annotation Task in Detail

We used 34 different LLMs, broken out as in Table 1, to characterize annotation performance metrics over improvements in the large language model industry. For each replication dataset, we found the original codebook and annotation instructions, and adopted these to be used with our LLM models. In most cases, we annotated all available observations found in the replication datasets. For the CAP and Manifesto Project materials, we replicated a random sample of 500 bills and 400 quasi-sentences, respectively.

As discussed in the paper, we argue that different annotation tasks require different approaches to evaluating LLM performance. For Type 1 annotation tasks where there is an objective ground truth, performance is straightforwardly evaluated with respect to this gold standard, and takes the form of either accuracy, or a weighted F1 metric and its component sensitivity and specificity measures. For Type 2 annotation tasks where we would like to think that expert agreement accurately captures some latent quantity of interest, we can again evaluate LLM performance via accuracy with respect to the expert consensus. However, it is also sensible to evaluate LLM performance with some measure of consistency, which we describe shortly. For Type 3 annotation tasks where we do not think an objective ground truth exists at all, LLM performance is instead best evaluated with measures of consistency such as intercoder reliability.

With these tools in mind, we turn to a summary of many different annotation tasks. Specifically, we apply these metrics to (Type 1) the annotation of social media posts for author partisanship, compared to either expert annotators or to mTurk workers (Törnberg, 2024); (Type 2) the annotation of congressional bills for topics (Gunes and Florczak, 2025); the annotation of party manifesto sentences for positions (Lehmann et al., 2024); and (Type 3) the annotation of social media posts for various types of sentiment, tone, emotion, and ideology (Bojić et al., 2025); or the annotation of texts for moral foundations, emotion, offensiveness, and sentiment (Rathje et al., 2024). For each replication dataset, we apply their original codebook and annotation instructions to all 34 LLMs across the OpenAI, Anthropic, and Mistral API endpoints available at the time of analysis. We then calculate the LLM performance by accuracy and by consistency, using the method described above. Starting with simple accuracy, we treat the consensus annotation as the ground truth in Type 2 and 3 tasks. We visualize each model's performance, broken out by model family, in Figure 7.

Across all tasks, the general pattern of improvement with more recent or more advanced versions of the LLMs is apparent. As illustrated, accuracy is highest for Type 1 tasks where an objective ground truth is available. For Type 2 and Type 3 tasks however, accuracy

Model	**Family**	**Provider**	**Release Date**
OpenAI (21 models)			
gpt-3.5-turbo-1106	GPT-3.5	OpenAI	2023-11-06
gpt-3.5-turbo-0125	GPT-3.5	OpenAI	2024-01-25
gpt-4-0613	GPT-4	OpenAI	2023-06-13
gpt-4-1106-preview	GPT-4	OpenAI	2023-11-06
gpt-4-0125-preview	GPT-4	OpenAI	2024-01-25
gpt-4-turbo-2024-04-09	GPT-4	OpenAI	2024-04-09
gpt-4o-2024-05-13	GPT-4	OpenAI	2024-05-13
gpt-4o-mini-2024-07-18	GPT-4	OpenAI	2024-07-18
gpt-4o-2024-08-06	GPT-4	OpenAI	2024-08-06
gpt-4o-2024-11-20	GPT-4	OpenAI	2024-11-20
gpt-4o	GPT-4	OpenAI	2024-11-20
chatgpt-4o-latest	GPT-4	OpenAI	2024-11-20
o1-2024-12-17	Reasoning	OpenAI	2024-12-17
o3-mini-2025-01-31	Reasoning	OpenAI	2025-01-31
gpt-5	GPT-5	OpenAI	2025-03-01
gpt-5-mini	GPT-5	OpenAI	2025-03-01
gpt-5.1	GPT-5	OpenAI	2025-05-01
gpt-5.1-chat-latest	GPT-5	OpenAI	2025-05-01
gpt-5.2	GPT-5	OpenAI	2025-07-01
gpt-5.2-chat-latest	GPT-5	OpenAI	2025-07-01
gpt-5.2-pro	GPT-5	OpenAI	2025-09-01
Anthropic (7 models)			
claude-3-haiku-20240307	Claude Haiku	Anthropic	2024-03-07
claude-sonnet-4-20250514	Claude Sonnet	Anthropic	2025-05-14
claude-opus-4-20250514	Claude Opus	Anthropic	2025-05-14
claude-opus-4-1-20250805	Claude Opus	Anthropic	2025-08-05
claude-sonnet-4-5-20250929	Claude Sonnet	Anthropic	2025-09-29
claude-haiku-4-5-20251001	Claude Haiku	Anthropic	2025-10-01
claude-opus-4-5-20251101	Claude Opus	Anthropic	2025-11-01
Mistral (6 models)			
ministral-3b-2512	Mistral Small	Mistral	2025-12-01
ministral-8b-2512	Mistral Medium	Mistral	2025-12-01
ministral-14b-2512	Mistral Large	Mistral	2025-12-01
mistral-small-2506	Mistral Small	Mistral	2025-06-01
mistral-medium-2508	Mistral Medium	Mistral	2025-08-01
mistral-large-2512	Mistral Large	Mistral	2025-12-01

Table 1: Large language models used in replication analysis. We test 34 models across three providers (OpenAI, Anthropic, Mistral) spanning release dates from June 2023 to December 2025. Models are grouped by provider and ordered by release date within each group. Family groupings used in analysis are summarized in the second column.

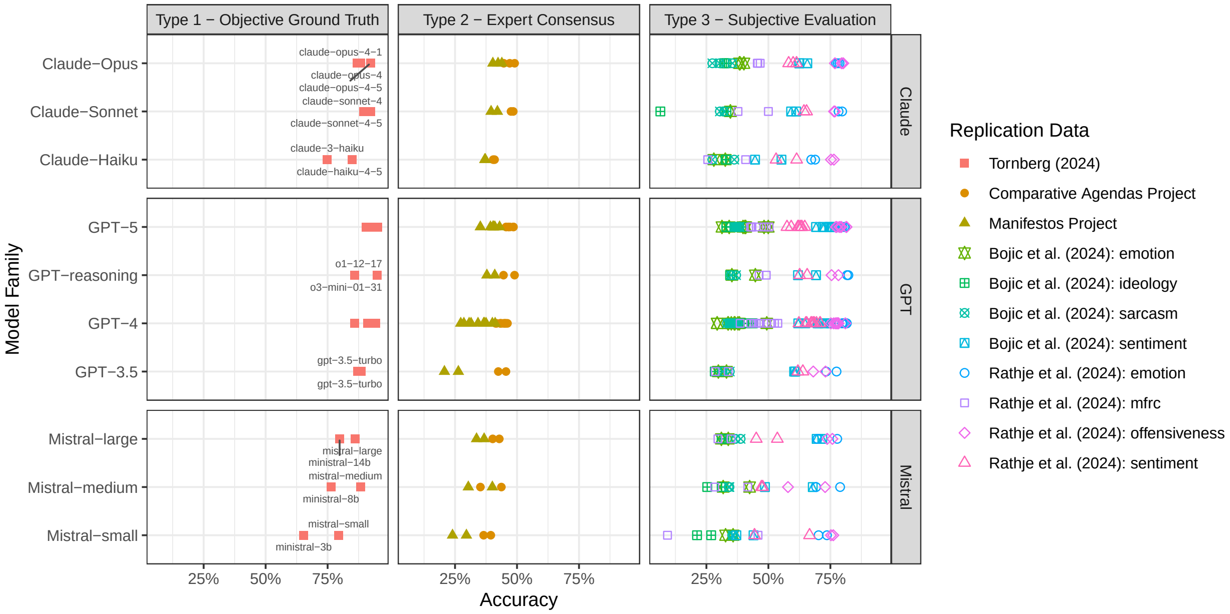

Figure 7: All models (points) broken out by task type (columns) and model family (y-axes) in terms of accuracy (x-axes) for 13 separate annotation tasks (colors and shapes)

declines, although the extent to which this reflects a failure on the part of the LLM, or less certainty about the consensus decision on the ground truth, is up for debate.

As such, we turn to an alternative metric of model performance that focuses on the consistency of annotations via intercoder reliability metrics. We operationalize consistency in one of two ways, depending on whether the publicly available replication materials include the raw annotations by multiple coders, or only the aggregate label (or, equivalently, annotations by a single coder with insufficient overlap to estimate intercoder reliability measures). If the former, we implement a bootstrapped measure of Krippendorff's alpha which compares the distribution of alpha values from randomly replacing one of the human coders with one of the LLMs and calculating the alpha value, with the original alpha value estimated with only humans. If replacing a human annotator with an LLM annotator increases the intercoder reliability, we can conclude that this LLM is "superior" to the human at least in terms of consistency.

Using the Törnberg (2024) replication materials as an example, we have annotations for author partisanship based on either a groups of mTurk workers or two political scientists. The two political scientists ("experts") labeled each author's partisanship on the basis of 500 tweets (250 from Democratic senators and 250 from Republican senators). This allows us to estimate Krippendorff's alpha for the human experts ($\alpha = 0.6481$) and then compare this to 1000 bootstrapped samples of the data in which we randomly replace one of the human annotators with one of the large language model annotations. As illustrated in Figure 8, most LLMs outperform the expert annotators' Krippendorff's alpha value. The notable exceptions include Mistral's smaller models, as well as Anthropic's Claude 3 haiku model from early March, 2024.

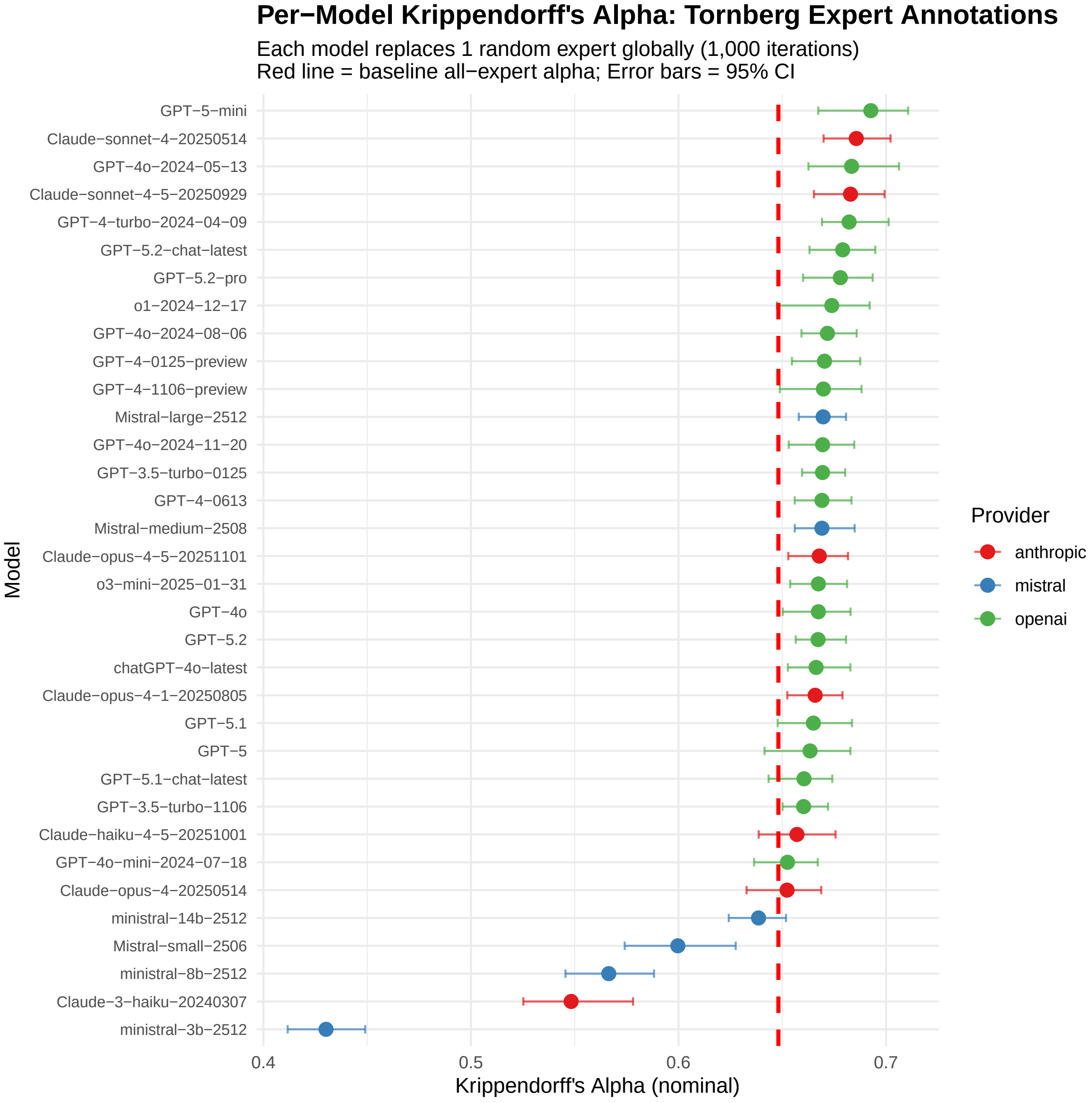

Figure 8: Average Krippendorff's alpha derived from 100 bootstrapped samples of the data, replacing one of the human expert annotators with an LLM annotator. Based on Törnberg (2024) replication materials.

We also subset the LLMs by family to explore the improvements in LLMs over time, replacing n human coders with all of the LLM models associated with a given version of an LLM. For example, we would randomly replace two human coders with Mistral-small and Ministral-3b to capture the change in intercoder reliability associated with this family of LLMs. We visualize this approach in Figure 9 for all suitable replication data.

For replication datasets that don't include the raw human annotations, we flip the above

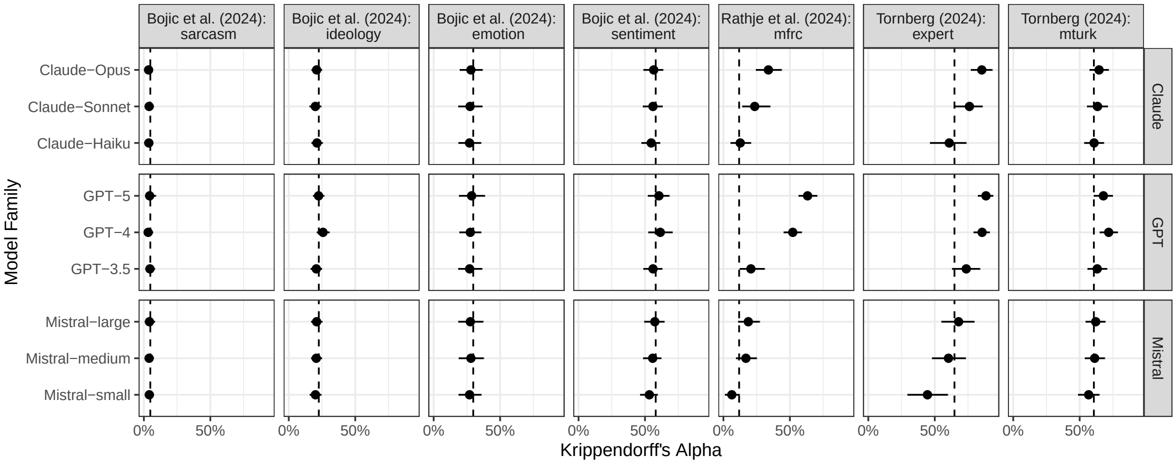

Figure 9: Krippendorff's alpha values (x-axes) for human annotators (vertical dashed line) and for randomly replacing n human coders with n LLMs from a given family (y-axes) 1,000 bootstrapped times (points). Horizontal bars indicate 95% intervals of bootstrapped values.

intuition on its head, and calculate the baseline Krippendorff's alpha using only the LLMs. We then randomly replace an LLM (or a family of LLMs) with either a human coder (if we have the raw annotations) or the human consensus value, and re-estimate Krippendorff's alpha value. Here, if we see a decline in Krippendorff's alpha value from replacing an LLM with a human, we would conclude that the human is less consistent than the LLM removed. Figure 10 summarizes these results, with vertical lines indicating the LLM-only Krippendorff's alpha value for the given annotation task. As illustrated, in most cases for most models, replacing that family of models with a human annotator reduces Krippendorff's alpha. The exceptions are for the weakest / oldest models, where replacing the LLM with a human can increase Krippendorff's alpha value.

The preceding plot is somewhat complex to read, so we instead put the difference in Krippendorff's alpha values between the full LLM value and that measured when replacing an LLM family with human annotations (see Figure 11). Now negative values on the x-axis indicate that the human replacement reduced Krippendorff's alpha from the LLM-only measure, while positive values indicate that the human improved consistency. As illustrated more clearly here, the majority of values are less than zero, consistent with our broad claim that humans are no longer the gold standard.

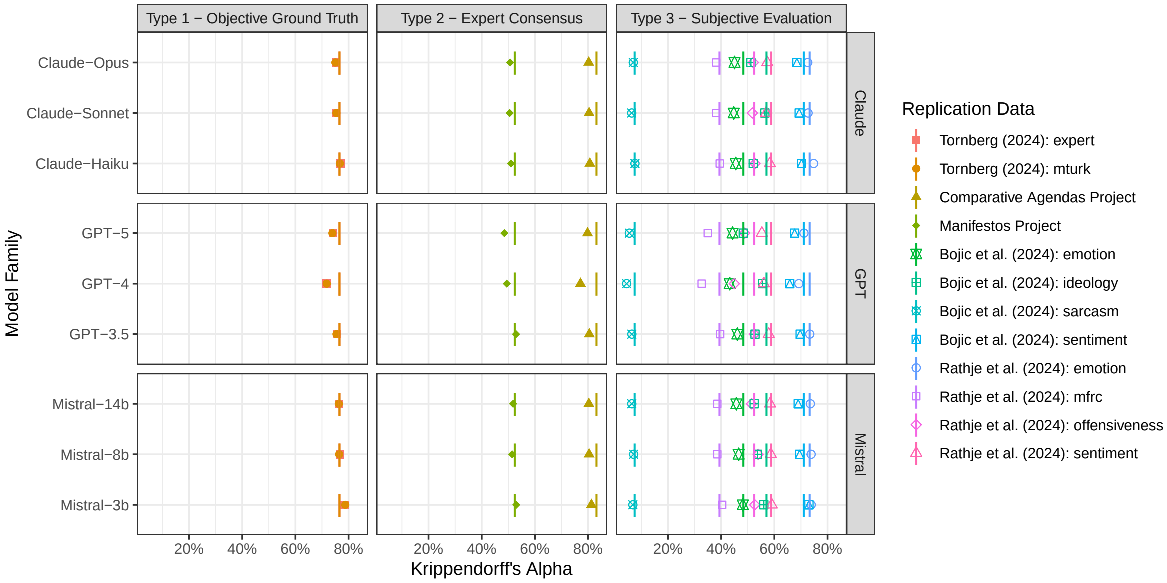

Figure 10: Krippendorff's alpha based on 34 LLM annotators (vertical lines) compared to Krippendorff's alpha value calculated when a family of LLMs (y-axes) is replaced with a human annotator.

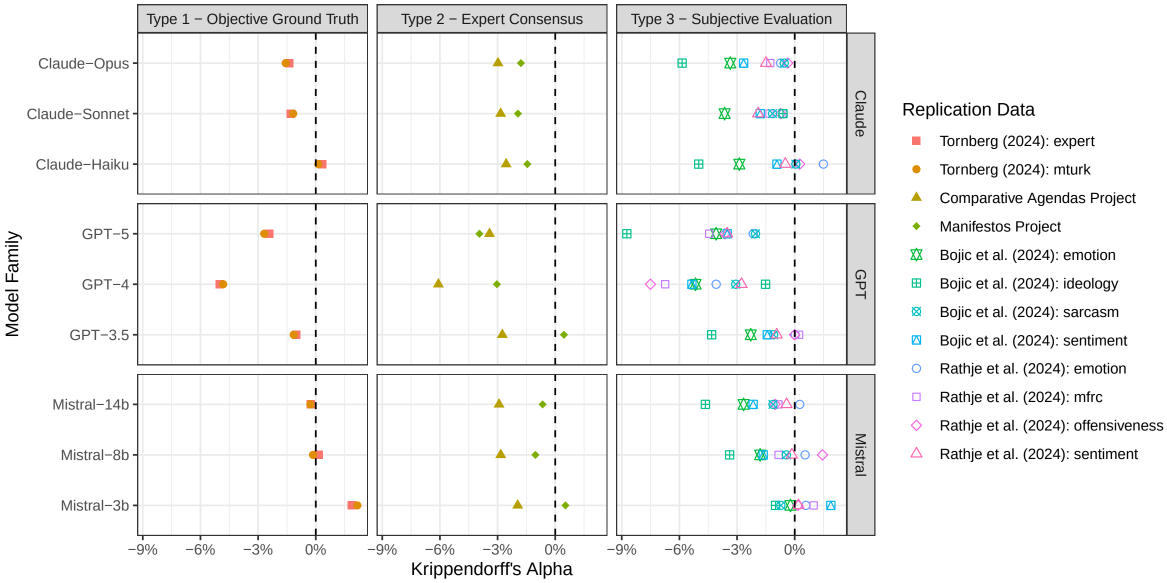

Figure 11: Average Krippendorff's alpha derived from 100 bootstrapped samples of the data, replacing one of the human expert annotators with an LLM annotator. Based on Törnberg (2024) replication materials.

B Monte Carlo Simulations

Our data simulated in Figure 3 from the manuscript is based on the following DGP.

1. Draw $D = D^*$ from a binomial distribution with size = 1 and probability π.

2. Draw $(X_1, X_2) = \mathbf{X}$ from a conditional multivariate normal distribution with

$$\Sigma = \begin{bmatrix} 1 & \sigma_{X_1,D} & \sigma_{X_2,D} \\ \sigma_{X_1,D} & 1 & \sigma_{X_2,X_1} \\ \sigma_{X_2,D} & \sigma_{X_2,X_1} & 1 \end{bmatrix}$$

3. Generate $Y \sim \mathcal{N}(\alpha + \beta D^* + \gamma_1 X_1 + \gamma_2 X_2, \sigma_Y)$ where $\alpha = 0$, $\beta = 1$, and γ are either both zero, or -2, 2. For clarity, we set σ_Y to be close to zero.

4. To form M, generate $m_i \in \{0, 1\}$. This identifies which observations are to be measured with error, subject to a differential parameter $\rho_{ME,Y}$ which captures how correlated M is with the outcome Y, and a misclassification rate $R \in [0, 0.5]$. Note that the differential parameter $\rho_{ME,Y}$ is bounded by R—i.e., the correlation between the observations measured with error and the outcome cannot be too strong if only 2% of the data is misclassified.

5. For observation i where $m_i = 1$, replace $D = 1 - D^*$.

6. Estimate $Y = \hat{\alpha} + \hat{\beta} D + \hat{\gamma}_1 X_1 + \hat{\gamma}_2 X_2 + e$ and compare with coefficients from $Y = \hat{\alpha} + \hat{\beta} D^* + \hat{\gamma}_1 X_1 + \hat{\gamma}_2 X_2 + e$.

We characterize bias from misclassification by varying R, π, $\rho_{ME,Y}$, and Σ, as well as allowing $\mathbf{X}$ to be prognostic of Y via γ. We start in Figure 12 by focusing on the two most influential dimensions: R and π, setting $\rho_{ME,Y}$, γ_1 and γ_2, and $\sigma_{X_1,D}$, $\sigma_{X_2,D}$ and σ_{X_1,X_2} all equal to zero. As expected, attenuation bias from misclassification of D declines linearly from 1 to 0 when $\pi = 0.5$, and grows more extreme as the skew in the true treatment D^* grows more extreme.

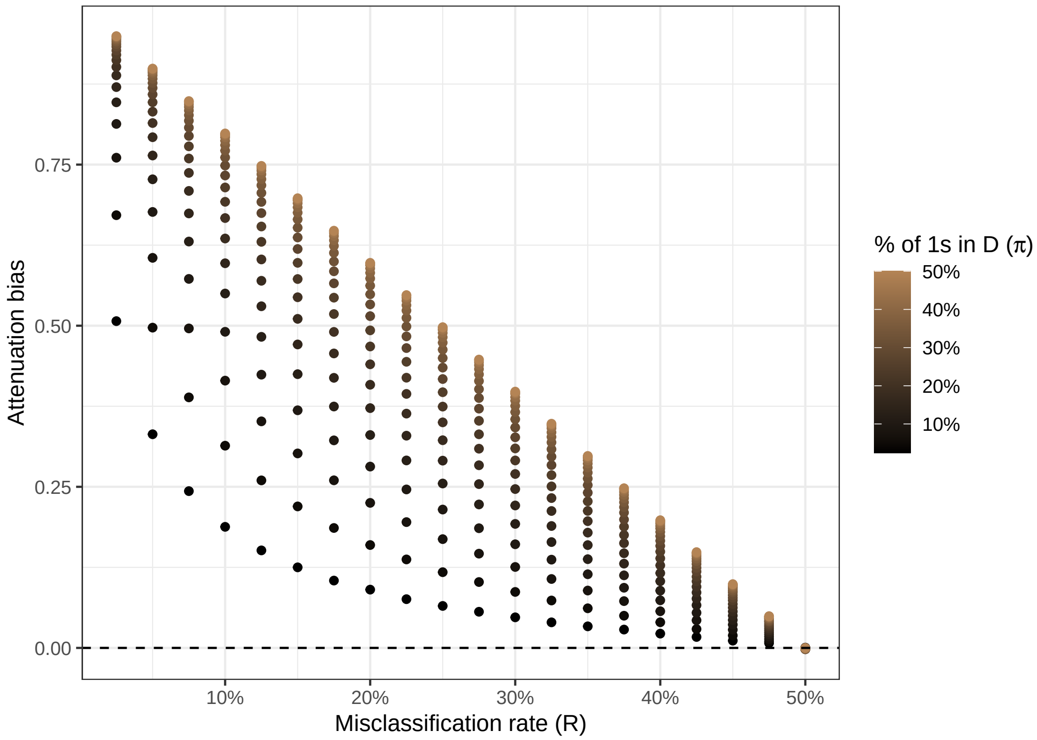

Figure 12: Attenuation bias (y-axis) from the misclassification rate R (x-axis) and skew in the true treatment π (colors).

We then explore the impact of differential measurement error across values of $\rho_{ME,Y}$ in Figure 13, revealing little to no impact unless the treatment is skewed (left panel). As is known, we see that differential measurement error can move us beyond the world of pure attenuation bias, as—in this particular setting—positive correlation between the observations measured with error and the outcome Y produce coefficient estimates that cross the null (i.e., true $\beta =$ but $\hat{\beta} < 0$). Notably, these extremes are not found with a balanced treatment variable (right panel), and remain orders of magnitude smaller for a given change in the correlation than what we observe for movements along the x-axis or skew in 12.

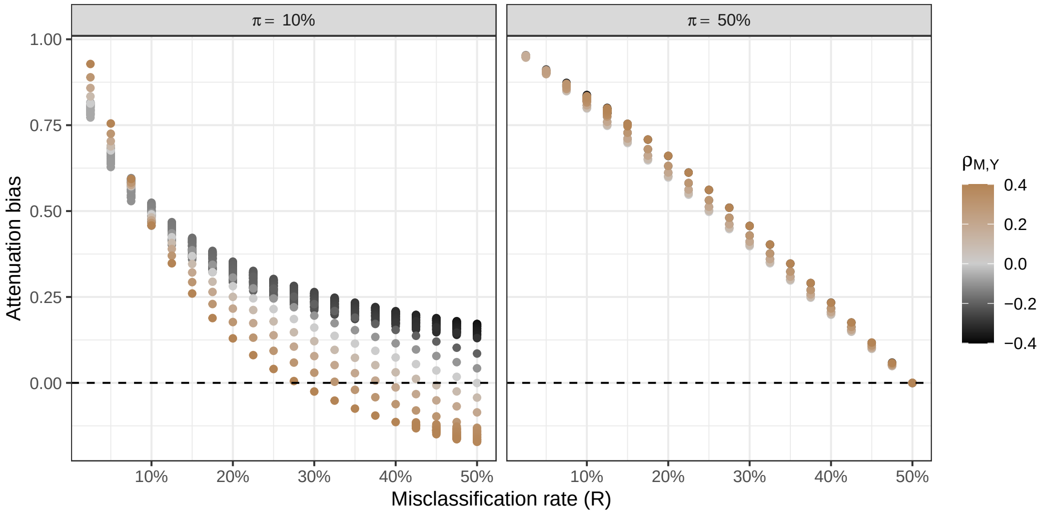

Figure 13: Attenuation bias (y-axis) from the misclassification rate R (x-axis) and correlation between the observations misclassified and the outcome (colors) for highly skewed (left panel) and symmetrically distributed (right panel) treatment vectors.

Finally, we turn to a more complicated data generating process in which there are controls which are correlated with both the treatment and the outcome. Specifically, we set $\sigma_{X_1,D} = \sigma_{X_2,D} = 0.4$, we set $\sigma_{X_2,X_1} = 0.6$, and finally welet $\gamma_1 = 2$ and $\gamma_2 = -2$. Across all tests, we compare the minimum and maximum values of $\rho_{M,Y}$ (0.4 and -0.4) which bound the range. These results are summarized for different levels of skew ($\pi = 0.1$ and $\pi = 0.5$) and presented in Figure 14. As above, we again find that the misclassification rate and the skew are most influential, although the worst case scenario with correlated controls (bottom-right panel of Figure 14) exhibits the most extreme range of possible attenuation bias.

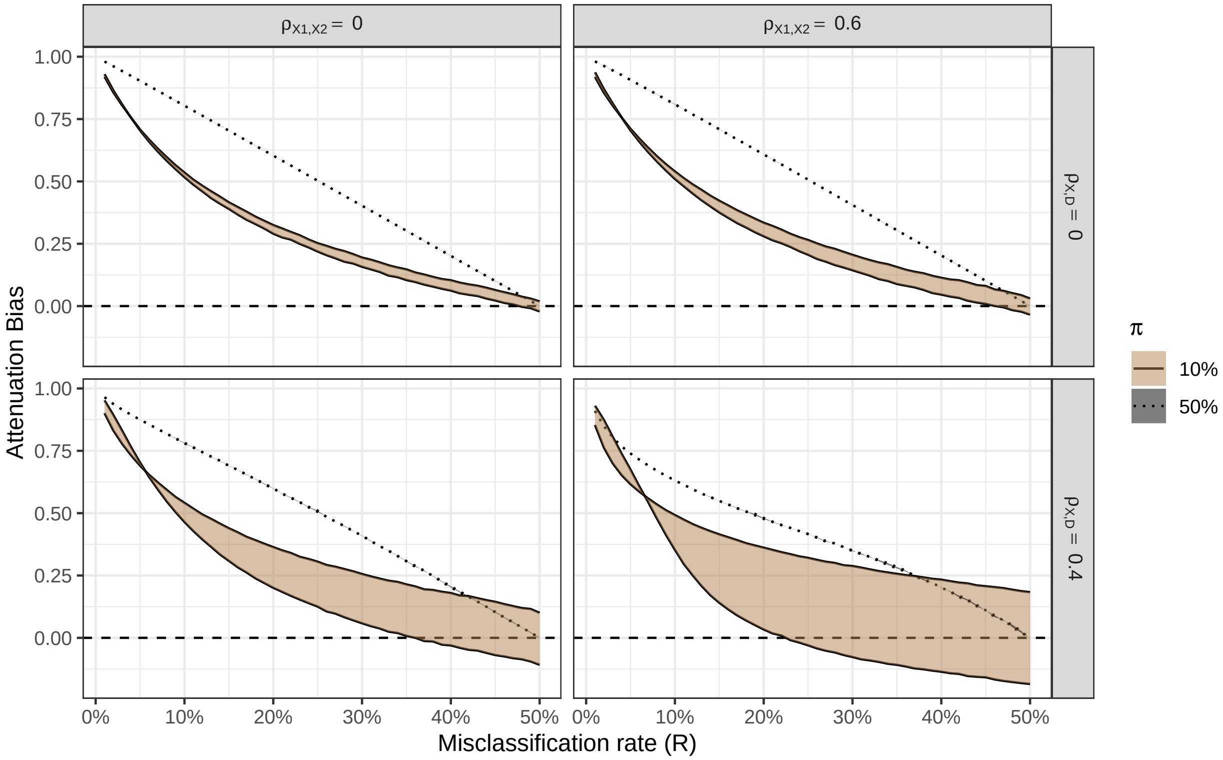

Figure 14: Attenuation bias (y-axis) from the misclassification rate R (x-axis) and correlation between the observations misclassified and the outcome (range of polygons) for skewed and unskewed data (solid and dotted lines, respectively) for varying correlation between the controls (columns) and varying correlation between the controls and the treatment (rows).

B.1 Non-linear Data Generating Process

Our main results rely on a data generating process that, while accommodating the types of characteristics of theoretical interest (skew, differential correlation with the outcome, correlated controls, etc.) is nevertheless linear in parameters by design. Here, we adopt the simulation approach found in Egami et al. (2023), summarized below.

- $\mathbf{X} \sim \mathcal{N}(\vec{0}, \sigma^X)$
  - $\mathbf{X}_i = (X_{i1}, \ldots, X_{i,10})$
  - For $\ell \in \{1, \ldots, 10\}$, $\Sigma^X_{\ell,\ell} = 1$ and for $\ell \neq \ell'$, $\sigma^X_{\ell,\ell'} = 0.3$
  - $X_{i,2} = \nVdash\{X_{i,2} > \text{qnorm}(0.8)\}$
- $W_i = \frac{0.1}{1+\exp(0.5X_{i,3}-0.5X_{i,2})} + \frac{1.3X_{i,4}}{1+\exp(-0.1X_{i,2})} + 1.5X_{i,4} + 0.5X_{i,1} + 1.3X_{i,1} + X_{i,2}$
- $Y_i \sim \text{Bernoulli}(\text{expit}(W_i))$
- $\hat{Y}_i = P_iY_i + (1 - P_i)(1 - Y_i)$ where $P_i \sim \text{Bernoulli}(P_q)$

- Model $Y_i = c + \beta_1 X_{i,1} + \beta_2 X_{i,1}^2 + \beta_3 X_{i,2} + \beta_4 X_{i,4} + \varepsilon_i$

We re-generate Figure 3 from the manuscript in Figure 15 below, illustrating substantively similar patterns in the contour plot. Even with highly non-linear data generating processes, the core claim that—assuming LLMs are generally improving over time, and that the change in the skew of the treatment cannot be too extreme —it is unlikely that using a state of the art LLM would dramatically change the findings in applied work.

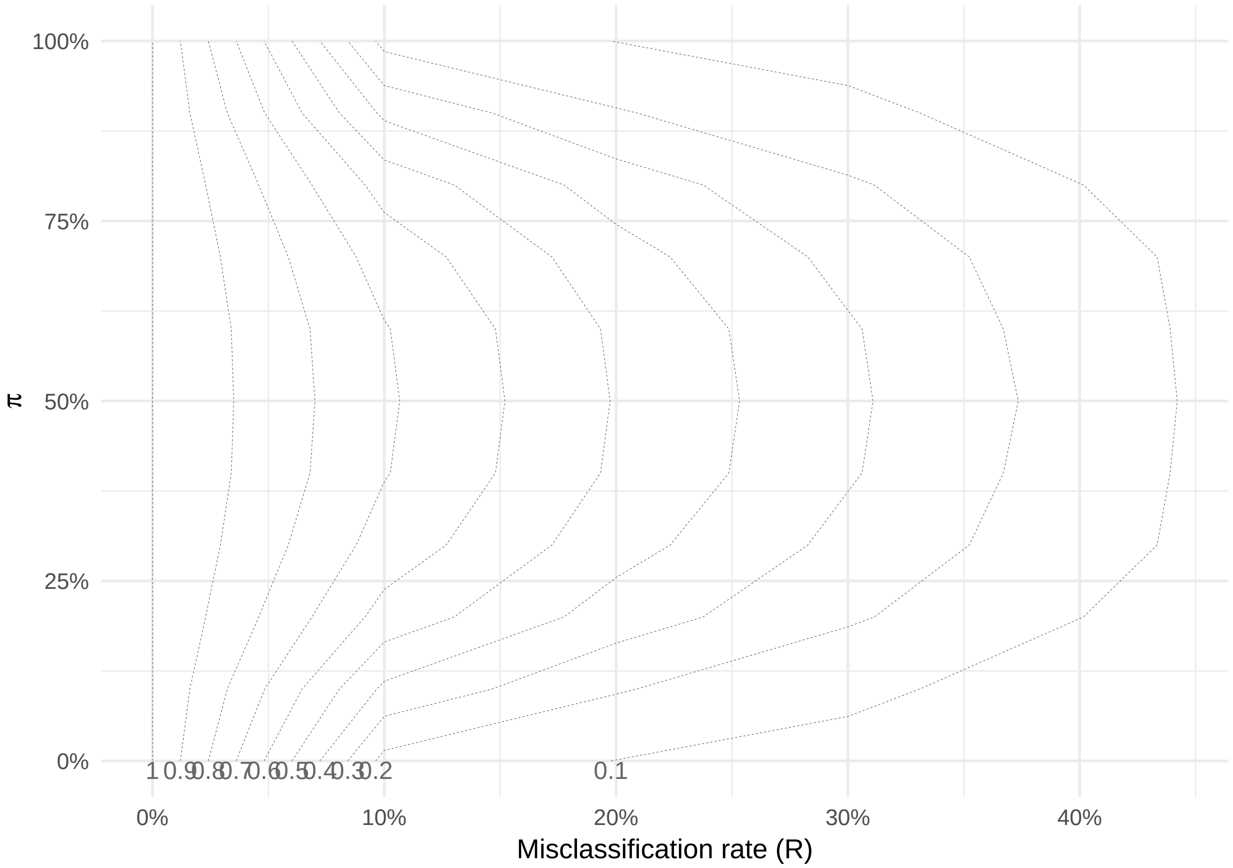

Figure 15: Contour plot of attenuation bias (dashed contour lines) by misclassification (x-axis) and skew in the treatment (y-axis) using the non-linear data generating simulation from Egami et al. (2023).

C Sensitivity Analysis for Treatment Misclassification

To assess the robustness of our estimated treatment effects to misclassification in the binary treatment variable, we implemented a simulation-based sensitivity analysis inspired by recent work on measurement error in causal inference. Our approach proceeds in three stages:

(1) estimating extreme bounds under structured misclassification, (2) simulating misclassification with controlled correlation to the outcome, and (3) comparing these results to benchmark simulations with randomly permuted treatment values. In all cases, we move away from the language of misclassification and remain agnostic about what it means in practice to reclassify an observation. Our sensitivity analyses thus only ask how far an observed coefficient might move under different scenarios when some proportion of the data is moved from one status to another; specifically, we are swapping $D = 1$ to $D = 0$ and vice versa.

Estimating Extreme Bounds. We begin by estimating the maximum and minimum ATE that could arise under a fixed rate of misclassification. These are the (most) extreme bounds. Specifically, for each assumed misclassification rate $R \in \{0.01, 0.02, \ldots, 0.5\}$, we identify the subset of observations whose treatment assignment—if flipped—would most strongly increase or decrease the estimated treatment coefficient. To accommodate covariates, we first residualize both the outcome and treatment on the set of control variables using linear regression. We then greedily identify the $R \times N$ observations for which flipping the residualized treatment variable would most increase (or decrease) the mean difference in residualized outcomes between treated and control groups. These flipped values are used to re-estimate the full regression model, and the resulting coefficients are recorded as the "extreme" upper and lower bounds for each misclassification rate.

Practically, denote M with typical value $m_i \in \{0, 1\}$ to be an indicator variable for whether an observation is reclassified and denote $\tilde{Y}$ to be the residualized values of Y from the regression of $Y = \mathbf{X}\beta$. To calculate the extreme upper bound on $\hat{\beta}_{\text{extreme}}$, we identify the largest $\tilde{Y}$ values where $D = 0$ and the smallest $\tilde{Y}$ values where $D = 1$ and then proceed down the list, flipping D until we have hit the limit of $R \times N$ observations to be reclassified. This ensures that we maximize any potential skew in the outcome variable, but also could result in only $D = 0$ being flipped to $D = 1$ if all the largest values of $\tilde{Y}$ are associated with $D = 0$ observations. The extreme lower bound follows the identical procedure, except that we reverse the $D = 0$ and $D = 1$ ordering of $\tilde{Y}$. The resulting $\hat{\beta}_{\text{extreme}}$ estimates thus reflect a highly unlikely extreme bound on the furthest β might range from the observed $\hat{\beta}$.

The intuition of this approach is visualized in Figure 16 which shows a normal (jittered) representation of the data in the left panel, a sorted visualization of the same data in the center panel, with the largest values of $D = 1$ and smallest values of $D = 0$ highlighted in red, and then the new difference in means when these observations are reclassified to $D = 0$ and $D = 1$ respectively, flipping the sign of the observed coefficient from positive to negative.

Deterministic Greedy Algorithm for Skewed Binary Treatment. The basic extreme bounds approach described above works well when the treatment is balanced (i.e., $\pi = P(D = 1) \approx 0.5$). However, when the treatment is skewed, we can achieve more extreme bounds by exploiting the asymmetry in group sizes. The key insight is that flipping observations from the minority group has a larger impact on the estimated treatment effect than flipping observations from the majority group, since each observation in the minority

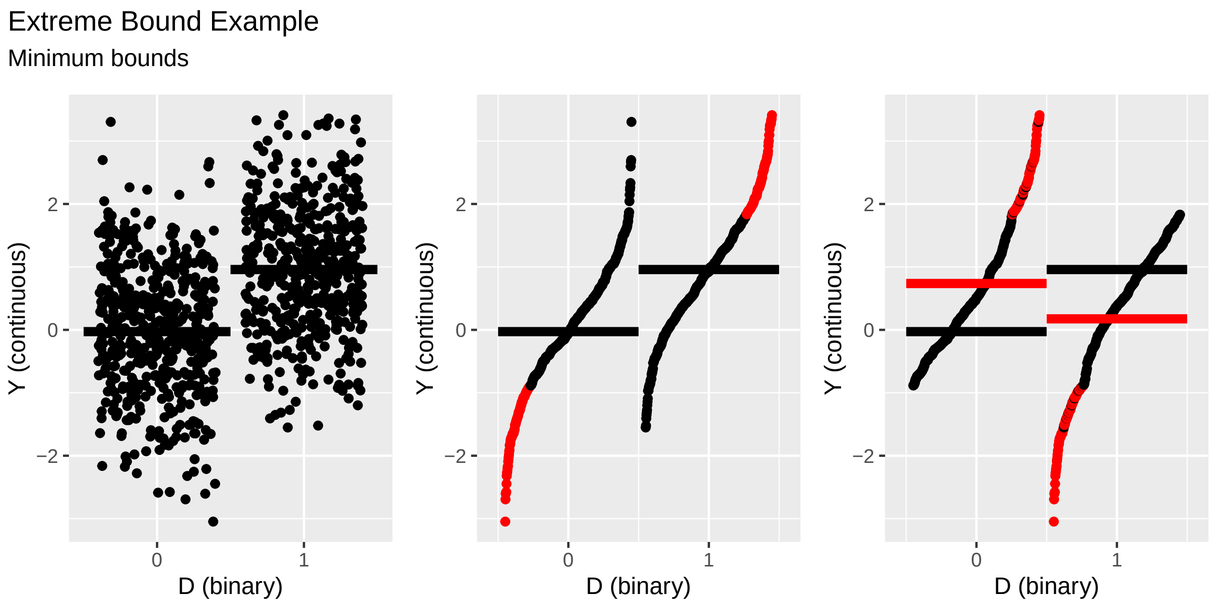

Figure 16: Demonstration of extreme bounds obtained by sorting data based on outcome value (or residualized values thereof when there are controls), selecting the observations with the largest (smallest) values of $D = 1$ and the smallest (largest) values of $D = 0$ in order to produce the smallest (largest) possible coefficient.

group contributes more weight to that group's mean.

Formally, let $n_1 = \sum_i D_i$ and $n_0 = N - n_1$ denote the number of treated and control observations, respectively. For a fixed reclassification rate R, we seek to allocate the $R \times N$ flips between two quantities: k_1 observations to flip from $D = 1$ to $D = 0$ and k_0 observations to flip from $D = 0$ to $D = 1$, where $k_1 + k_0 = R \times N$.

The algorithm proceeds as follows:

1. Sort all observations by residualized outcome $\tilde{Y}$.
2. For the **maximum** bound: Among $D = 1$ observations, identify the k_1 observations with the *smallest* $\tilde{Y}$ values. Among $D = 0$ observations, identify the k_0 observations with the *largest* $\tilde{Y}$ values.
3. For the **minimum** bound: Reverse the ordering (largest $\tilde{Y}$ for $D = 1$, smallest $\tilde{Y}$ for $D = 0$).
4. Iterate over all valid allocations (k_1, k_0) such that $k_1 + k_0 = R \times N$, $0 \leq k_1 \leq n_1$, and $0 \leq k_0 \leq n_0$.
5. For each allocation, flip the treatment assignments and re-estimate the ATE. Record the allocation that produces the most extreme bound.

When the treatment is highly skewed (e.g., $\pi = 0.1$), the optimal allocation typically involves flipping a disproportionate number of observations from the minority group. For instance, if $R = 0.05$ and $\pi = 0.1$, the algorithm may flip nearly all available $D = 1$ observations before touching any $D = 0$ observations, since each $D = 1$ observation carries ten times the weight in computing the treated group mean.

This approach guarantees that we identify the true extreme bounds for any level of treatment skew and misclassification rate, at the cost of enumerating $O(R \times N)$ candidate allocations. In practice, this is computationally feasible for moderate sample sizes (e.g., $N \leq 10{,}000$) and misclassification rates (e.g., $R \leq 0.5$).

Outcome-Correlated Misclassification Simulations. While the extreme bounds offer a worst-case scenario, these are highly unlikely. To put structure on more plausible bounds, we start by measuring the observed correlation between the reclassification indicator M from the extreme bounds setting, and the outcome Y. This correlation can be interpreted as an estimate of the differential reclassification implied by the extreme bounds. However, there are multiple combinations of observations that might achieve the same observed differential reclassification beyond those that maximize the minimum and maximum values of $\hat{\beta}_{extreme}$. We therefore simulate the distribution of coefficients that adhere to both the desired misclassification rate R, and the observed differential reclassification $\hat{\rho}_{Y,M} \mid \hat{\beta}_{extreme} = \frac{cov(Y,M)}{\sigma_Y \sigma_M}$.

In practice, for each assumed misclassification rate R, we calculate $\hat{\rho}_{Y,M} \mid \hat{\beta}_{extreme}$ and then use a greedy swapping algorithm to construct a new binary mask M' such that the correlation between M' and the outcome Y is approximately equal to a $\hat{\rho}_{Y,M} \mid \hat{\beta}_{extreme}$ (e.g., the observed correlation in the greedy-flipped data). This procedure begins by initializing M' as an indicator for the top or bottom $R \times N$ values of Y and then iteratively swaps elements in and out of M' until the target correlation is reached within a small tolerance. Taking a positive value of $\hat{\rho}_{Y,M} \mid \hat{\beta}_{extreme}$ as an example, we first order the data by Y set the top $R \cdot N$ observations in terms of Y to $M' = 1$, while the remaining observations set $M' = 0$. This gives us the maximum correlation $\hat{\rho}_{Y,M'}$. We then randomly flip M' values until $\hat{\rho}_{Y,M'} \approx \hat{\rho}_{Y,M} \mid \hat{\beta}_{extreme}$. Once we are close enough in terms of the observed differential reclassification, we then flip D for all $M' = 1$, re-estimate the regression, and repeat this process 100 times to obtain the empirical distribution of $\hat{\beta}_{diff \mid \hat{\beta}_{extreme}}$. Note that there are two measures of $\hat{\rho}_{Y,M} \mid \hat{\beta}_{extreme}$ for each value of R: the upper extreme bound and the lower extreme bound. We calculate the distribution of $\hat{\beta}_{diff}$ for both separately, and then use the 97.5% percentile associated with the upper extreme's distribution, and the 2.5% percentile associated with the lower extreme's distribution.

Benchmark: Random Flipping. Finally, we also calculate a naïve set of bounds which reflect what we might expect to see if we simply choose $R \times N$ observations at random and flip their observed value of treatment. These simulations do not target any specific relationship between the outcome and the flipping process and thus reflect the distribution of treatment effects under purely random reclassification.

C.1 Multi-Class Sensitivity Analysis

Translating this intuition into a sensitivity analytic framework is theoretically straightforward, although computationally intensive as K increases. For a given coefficient of interest, the core intuition remains the same: identify the observations for which switching $\mathbb{I}(D_i = j)$ maximizes (minimizes) the associated coefficient estimate $\hat{\beta}_j$. Using the maximum extreme bound as an example, this entails identifying the observations for which $\mathbb{I}(D_i = j) = 0$ and $\tilde{Y}_i$ is large, and for which $\mathbb{I}(D_i = j) = 1$ and $\tilde{Y}_i$ is small, where $\tilde{Y}$ is the residualized value of Y. As above, the mask vector identifying these observations is denoted M_{extreme}.

However, swapping $\mathbb{I}(D_i = j) = 0$ to a 1 requires swapping some other $\mathbb{I}(D_i \neq j) = 1$ to a zero. This will impact the extreme $\hat{\beta}_j$ of interest in one of two related ways. First, if we "borrow" the swap from the reference category (or "hold-out"), this will change the composition of the comparison group. Second, if we "borrow" the swap from some other category, this will change the residualized $\tilde{Y}$ values as per the Frisch-Waugh-Lovell theorem.

To provide a concrete example, consider a regression of wage on ethnorace where the multi-class category is divided into white, black, and other. We set white as the reference category, and create binary indicators for black and other. Then, to calculate the extreme maximum bound on $\hat{\beta}_{\text{black}}$ subject to some reclassification rate amounting to a total of ten reclassified observations, we find five non-black observations with high wages, and five black observations with low wages. However, to swap the high-wage non-black observations to black, we need to swap their observed ethnoracial dummy to a zero. If these five observations are all originally classified as white, this means the comparison group has five fewer observations and loses those high-wage Y values. This is, from the perspective of calculating an extreme bound, "good" because it augments the extremity of $\hat{\beta}_{\text{black}}$ just as it does in the binary case. But if these five observations are all originally classified as other, the consequences for the extreme coefficient estimate are harder to predict, and are a function of their impact on $\hat{\beta}_{\text{other}}$ which, subsequently, impacts the $\tilde{Y}$ values used to calculate $\hat{\beta}_{\text{black}}$. In this case, it is possible that a more extreme estimate of $\hat{\beta}_{\text{black}}$ could be obtained by finding some other observation originally classified as white with a slightly lower wage. Alternatively, depending on the underlying correlational structure, it might be that swapping observations originally classified as other is, in fact, the biggest boost to the extreme value of $\hat{\beta}_{\text{black}}$ estimate.

Transition Matrix Representation. We represent the reclassification pattern using a $K \times K$ transition matrix $\mathbf{T}$ where element T_{jk} denotes the number of observations reclassified from category j to category k. The diagonal elements T_{jj} represent observations that remain in their original category. The off-diagonal elements capture the flow of observations between categories, subject to the constraints:

- Row sums must not exceed the number of observations in each category: $\sum_{k=1}^{K} T_{jk} \leq n_j$ for all j, where $n_j = \sum_i \mathbb{I}(D_i = j)$.
- The total number of reclassifications must equal $R \times N$: $\sum_{j \neq k} T_{jk} = R \times N$.

- All entries must be non-negative integers: $T_{jk} \in \{0, 1, 2, \dots\}$.

For a multi-category regression model of the form $Y = \alpha + \sum_{j=1}^{K-1} \beta_j \mathbb{I}(D = j) + \mathbf{X}\gamma + \varepsilon$, our goal is to identify the transition matrix $\mathbf{T}$ that produces the most extreme value (maximum or minimum) of a target coefficient β_j for some category of interest j.

Deterministic Greedy Algorithm (Small K). For small K (e.g., $K = 3$ categories), we employ a deterministic greedy algorithm that directly searches over the space of feasible transition matrices. This approach guarantees finding the true extreme bounds but becomes computationally prohibitive for $K > 3$.

The algorithm proceeds as follows:

1. **Residualize:** Regress Y on the control variables $\mathbf{X}$ to obtain residuals $\tilde{Y}$.
2. **Sort within categories:** For each category $j \in \{1, \dots, K\}$, sort observations by their residualized outcome $\tilde{Y}$.
3. **Initialize candidates:** For each possible destination category $k \neq j$ (where j is the focal category whose coefficient we want to maximize), identify the observations that would most increase $\hat{\beta}_j$ if moved from their current category to category k.
    - To **maximize** $\hat{\beta}_j$: Move observations with *large* $\tilde{Y}$ from the reference category to category j, and move observations with *small* $\tilde{Y}$ from category j to the reference category.
    - To **minimize** $\hat{\beta}_j$: Reverse the ordering.
4. **Greedy allocation:** Starting with the transition matrix $\mathbf{T} = \mathbf{0}$, iteratively:
    - Compute the marginal impact on $\hat{\beta}_j$ of incrementing each feasible off-diagonal element T_{jk} by one.
    - Select the transition with the largest marginal impact.
    - Update $\mathbf{T}$ and the set of feasible transitions (respecting row sum constraints).
    - Repeat until $\sum_{j \neq k} T_{jk} = R \times N$.
5. **Apply reclassification:** Use the final transition matrix $\mathbf{T}$ to reclassify observations, re-estimate the regression, and record $\hat{\beta}_j$.

This greedy approach works well for $K = 3$ because the number of feasible transitions at each step is manageable ($O(K^2) = 9$ candidate moves per iteration). However, for $K > 3$, the combinatorial explosion of possible transition patterns makes this approach impractical.

Optimization-Based Algorithm (Large K). For $K > 3$, we employ a continuous relaxation of the transition matrix problem that can be solved using gradient-based optimization. The key insight is to relax the integer constraint $T_{jk} \in \mathbb{Z}_+$ to $T_{jk} \in \mathbb{R}_+$ and then round the solution to the nearest feasible integer matrix.

Let $\mathbf{t} = \text{vec}(\mathbf{T})$ denote the vectorized transition matrix (excluding diagonal elements). We formulate the optimization problem as:

$$\begin{aligned} \max_{\mathbf{t}} \quad & \hat{\beta}_j(\mathbf{t}) && (4) \\ \text{subject to} \quad & \sum_{k \neq j} t_{jk} \leq n_j \quad \forall j && (5) \\ & \sum_{j \neq k} t_{jk} = R \times N && (6) \\ & t_{jk} \geq 0 \quad \forall j, k && (7) \end{aligned}$$

The objective function $\hat{\beta}_j(\mathbf{t})$ is the estimated coefficient for category j after applying the (continuous) reclassification pattern $\mathbf{t}$. While $\hat{\beta}_j$ is not available in closed form, we can approximate its gradient using automatic differentiation or finite differences, enabling the use of standard constrained optimization solvers (e.g., L-BFGS-B, SLSQP).

The algorithm proceeds as follows:

1. **Initialize:** Start with a feasible initial guess $\mathbf{t}_0$ (e.g., uniform allocation across all off-diagonal elements).

2. **Optimize:** Use a gradient-based solver to find $\mathbf{t}^* = \arg\max_{\mathbf{t}} \hat{\beta}_j(\mathbf{t})$ subject to constraints (5)–(7).

3. **Round:** Convert the continuous solution $\mathbf{t}^*$ to an integer transition matrix $\mathbf{T}^*$ using randomized rounding or a greedy rounding procedure that respects the constraints.

4. **Refine:** Optionally apply local search (e.g., steepest ascent) to improve $\mathbf{T}^*$ by swapping individual observations.

5. **Evaluate:** Apply the final reclassification pattern $\mathbf{T}^*$, re-estimate the regression, and record $\hat{\beta}_j$.

In practice, we find that this optimization-based approach scales efficiently to $K = 10$ or more categories, producing near-optimal solutions in a fraction of the time required by the greedy algorithm. The key trade-off is that the optimization approach does not guarantee finding the global optimum.

C.2 Software Implementation: The `futureproofR` Package

All methods described in this paper are implemented in the `futureproofR` R package (v0.5.0), currently under development. The package provides a unified interface for sensitivity analysis of measurement error (misclassification) in binary and multi-class treatment variables, with support for control variables and multiple bound types.

Core Functions. The main exported functions are:

- `misclass_sens()`: Performs sensitivity analysis for binary treatment variables. Computes extreme bounds (maximum and minimum), outcome-correlated differential bounds, and naïve random-flipping bounds for user-specified misclassification rates. Returns a data frame with coefficient estimates across all bound types.

- `misclass_sens_plot()`: Produces publication-ready sensitivity plots for binary treatments, showing coefficient estimates as a function of misclassification rate with shaded regions for extreme, differential, and naïve bounds.

- `multiclass_sens_greedy()`: Implements the deterministic greedy algorithm for multi-class treatments. Recommended for $K \leq 3$ categories where the algorithm guarantees finding true extreme bounds.

- `multiclass_sens_optim()`: Implements the optimization-based algorithm for multi-class treatments with $K > 3$ categories. Uses continuous relaxation and gradient-based optimization to efficiently identify near-optimal reclassification patterns.

- `contour_plot()`: Creates contour plots (à la Figure 3 in the main text) showing how coefficient estimates vary as a function of both misclassification rate and differential reclassification.

Example Usage. A minimal example for binary treatment:

```
library(futureproofR)

# Fit model
m <- lm(Y ~ D + X, data = df)

# Run sensitivity analysis
results <- misclass_sens(
  dat = df,
  outcome = "Y",
  treatment = "D",
  binary = "D",
  nsims = 100,
  m = m,
  R_vect = seq(0.01, 0.3, by = 0.01)
)
```

```
# Plot results
misclass_sens_plot(results, m)
```

For multi-class treatments:

```
# Greedy algorithm (K=3)
results <- multiclass_sens_greedy(
  D_obs = df$category,
  Y = df$outcome,
  Z = df[, c("X1", "X2")],
  K = 3,
  target_beta = 3,
  prop_reclassified = 0.2
)

# Optimization algorithm (K>3)
results <- multiclass_sens_optim(
  D_obs = df$category,
  Y = df$outcome,
  Z = df[, c("X1", "X2")],
  K = 5,
  target_beta = 3,
  prop_reclassified = 0.2
)
```

Computational Performance. The package is optimized for moderate-sized datasets ($N \leq 50{,}000$). For binary treatments, extreme bounds computation scales as $O(R \times N^2)$. For multi-class treatments with $K > 3$, the optimization-based approach scales approximately as $O(K^2 \cdot N \cdot \log N)$ per misclassification rate, making it feasible for datasets with tens of thousands of observations and up to $K = 10$ categories.

The package uses progress bars for long-running computations and includes built-in parallelization support to speed up computation across multiple misclassification rates and simulation draws.