Benchmarking Debiasing Methods for LLM-based Parameter Estimates

Nicolas Audinet de Pieuchon[1,4] **Adel Daoud**[1,2] **Connor T. Jerzak**[3]
Moa Johansson[1,4] **Richard Johansson**[1,4]
[1]Chalmers University of Technology, Sweden [2]Linköping University, Sweden
[3]University of Texas at Austin, USA [4]University of Gothenburg, Sweden
{nicolas.audinet, daoud, moa.johansson, richajo}@chalmers.se,
connor.jerzak@austin.utexas.edu

Abstract

Large language models (LLMs) offer an inexpensive yet powerful way to annotate text, but are often inconsistent when compared with experts. These errors can bias downstream estimates of population parameters such as regression coefficients and causal effects. To mitigate this bias, researchers have developed debiasing methods such as Design-based Supervised Learning (DSL) and Prediction-Powered Inference (PPI), which promise valid estimation by combining LLM annotations with a limited number of expensive expert annotations.

Although these methods produce consistent estimates under theoretical assumptions, it is unknown how they compare in finite samples of sizes encountered in applied research. We make two contributions: First, we study how each method's performance scales with the number of expert annotations, highlighting regimes where LLM bias or limited expert labels significantly affect results. Second, we compare DSL and PPI across a range of tasks, finding that although both achieve low bias with large datasets, DSL often outperforms PPI on bias reduction and empirical efficiency, but its performance is less consistent across datasets. Our findings indicate that there is a bias-variance tradeoff at the level of debiasing methods, calling for more research on developing metrics for quantifying their efficiency in finite samples.

1 Introduction

Large language models (LLMs) are transforming disciplines that use text as a form of evidence in testing theories, something particularly evident in computational social science (Ziems et al., 2024; Törnberg, 2024; Bail, 2024; Argyle et al., 2023). LLMs are being used to extract features critical for substantive research questions, across a myriad of domains, from measuring political ideology (Sim et al., 2013), style and tone of writing (El-Haj et al., 2016), level of politeness (Priya et al., 2024), the likelihood of epidemiological events (Kino et al., 2021), to describing neighborhoods' health and living conditions (Murugaboopathy et al., 2025), and beyond. Although the use of LLMs promises to speed up the process of annotating these variables, which would previously have required time-consuming hand annotation by experts, if LLMs provide a wrong or suboptimal answer (i.e., a biased reply), downstream scientific estimates will also be biased (Egami et al., 2024; Angelopoulos et al., 2023a).

Thus, although LLMs are powerful, these models often annotate in a way that is inconsistent with expert annotators (Audinet de Pieuchon et al., 2024; Lin and Zhang, 2025). The distribution of LLM annotation errors can be heterogeneous or correlated with other variables of interest. These errors then lead to misleading substantive interpretations (McFarland and McFarland, 2015).

To handle these biases, *debiasing methods*[1] have been developed, most prominently Prediction-Powered Inference (PPI) (Angelopoulos et al., 2023a) and Design-based Supervised Learning (DSL) (Egami et al., 2023, 2024). Both frameworks produce an unbiased estimate by combining the LLM annotations with a smaller set of expert annotations. The biases in LLM-based estimates are then compensated for by introducing a *rectifier* created by comparing the two sets of annotations for the subset of samples that have both the LLM (predicted) annotation and the expert's annotation.

Debiasing methods have been shown to work in large (population) samples (Angelopoulos et al., 2023a; Egami et al., 2024), yet there is a lack of knowledge about *when* and *how much* debiasing methods provide added value in finite samples—

[1] We stress that in this paper, the term *bias* refers to an incorrectly estimated parameter in a statistical model, and a *debiasing* method corrects the misestimation. We do not consider *bias* in the sense of e.g. demographic biases in NLP representations.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 19757–19772
November 4-9, 2025 ©2025 Association for Computational Linguistics

which is what most domain researchers have at their disposal. There are no closed-form expressions that relate a debiasing method's efficacy to the allocation of expert versus model-generated annotations, leaving practitioners without analytic guidance on when one should prefer DSL or PPI over simply collecting more expert annotations. This lack of guidance will, in turn, hamper the uptake of debiasing methods or likely reinforce ill use of LLMs in applied scientific domains.

Accordingly, to address this lack, we articulate the following research questions:

RQ1: When is a debiased, large-scale LLM annotation dataset statistically preferable to a finite expert-only dataset for unbiased estimation of a population parameter?

RQ2: What are the performance differences between the debiasing methods, and how do they vary across datasets and LLM-based annotators?

We tackle these questions by comparing PPI and DSL across four datasets and four annotation procedures. To our knowledge, ours is the first effort to compare debiasing methods empirically. In foreshadowing our results, our analysis shows that compared to PPI, DSL achieves better debiasing results on average, but it is also the most variable in performance. Thus, PPI has a higher degree of stability; DSL is less consistent in gains. Our findings call for more research into the advantages and disadvantages of various biasing methods with respect to data forms.

2 Background: Methods for Debiasing LLM-based Estimates

Let $\mathcal{D} = \{(d_i, \mathbf{x}_i, \hat{y}_i)\}_{i=1}^{N}$ be a corpus of N documents d_i, with associated independent variables $\mathbf{x}_i \in \mathbf{X} = \{\mathbf{x}_i\}_{i=1}^{N}$ and LLM annotations $\hat{y}_i \in \widehat{Y} = \{\hat{y}_i\}_{i=1}^{N}$. A subset of $\mathcal{D}$ of size n also has additional expert annotations $y_j \in Y = \{y_j\}_{j=1}^{n}$, where $n \leq N$. Expert annotations are taken to be the ground truth and are generally costly (Gilardi et al., 2023).

Next, we focus on a general parameter of interest θ, which represents the result of the downstream statistical analysis. For example, this could be a regression coefficient or a class prevalence rate. The goal of the debiasing methods is to create an estimator f which estimates θ based on $\mathbf{X}$, Y, and $\widehat{Y}$. Ideally, the estimator should be *consistent*, meaning that $f(\mathbf{X}, Y, \widehat{Y}) \to \theta$ as $N \to \infty$, and *precise*, meaning that we want to keep the variance and confidence intervals as small as possible.

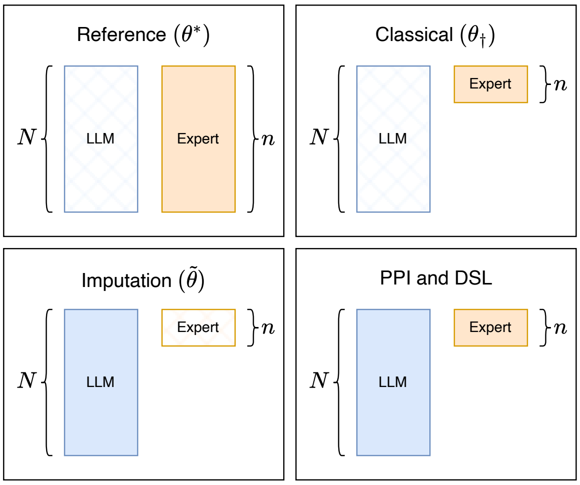

Figure 1: The reference model (top-left) is estimated from expert annotations (orange) for all N samples in the dataset (i.e., $n = N$, with full expert labeling). The classical model (top-right) uses only the n expert samples for downstream estimation ($n \ll N$). The imputation model (bottom-left) uses only the generated annotations for all N samples (blue). The *debiased* models (bottom-right) use *both* LLM annotations for all N samples and expert annotations for the subset of n samples.

One way to achieve this would be to ignore $\widehat{Y}$ entirely and only use the unbiased expert annotations Y. We call this the *classical estimator* $\theta_\dagger = f(\mathbf{X}, Y)$, which is usually generated by minimizing a loss. Although this estimator produces unbiased estimates, it can have a large variance if we have few expert annotations. We call the classical estimator trained with expert annotations for *all* N samples the *reference estimator* θ^*, which corresponds to the ideal but costly model that the debiasing methods are aiming towards.

Another approach would be to only use LLM annotations $\widehat{Y}$ and ignore the expert annotations Y. We call this the *imputation estimator*, $\tilde{\theta} = f(\mathbf{X}, \widehat{Y})$. Here, we rely on the assumption that we can exchange the expert annotations for the LLM annotations. The hope is that, while LLM annotations might be noisier than expert annotations, we can counteract the noise by simply generating as many labels as needed, given a large enough corpus. However, the LLM may exhibit systematic biases different from those of the expert human annotators, meaning that $|\tilde{\theta} - \theta^*| > 0$ as $N \to \infty$, and therefore this assumption does not hold in general. In turn, this leads to a biased downstream estimate, and one runs the risk of being "precisely inaccurate"

(McFarland and McFarland, 2015).

A third approach claims to be both unbiased and more precise than $\theta_{\dagger}$. Such methods typically work by estimating parameters on LLM annotations, with a *rectifier* constructed from the difference between the generated and expert annotations for the subset of the corpus for which we have both (see Figure 1). In this paper, we investigate PPI and DSL as two of the most prominent among these methods.

Prediction-Powered Inference (PPI). PPI offers a protocol for integrating LLM predictions into downstream statistical inference via first-order debiasing (Angelopoulos et al., 2023a). It begins by treating the LLM predictions as if they were true labels and forming the "imputation estimate": $\tilde{\theta} = \text{argmin}_{\theta} \frac{1}{N}\sum_{i=1}^{N} \ell_{\theta}(\mathbf{x}_i, \hat{y}_i)$, where ℓ_{θ} is the loss defining our estimand, such as the binary cross-entropy for a logistic regression. In general $\tilde{\theta}$ is biased, so PPI introduces the *rectifier*, which, in the one parameter case equals

$$r_{\theta} = \mathbb{E}\big[\nabla_{\theta}\ell_{\theta}(\mathbf{x}_i, y_i) - \nabla_{\theta}\ell_{\theta}(\mathbf{x}_i, \hat{y}_i)\big],$$

the gradient terms capturing the systematic distortion from substituting $\hat{y}_i$ for the true y_i (the gradient difference reveals the bias direction in parameter space, which we then offset to debias). We estimate r_{θ} on the labeled sample and estimate the imputed gradient on the unlabeled set using plugin estimators. The final, first-order debiased estimate is then $\tilde{\theta} - \widehat{r}_{\theta}$. Because $\widehat{r}_{\theta}$ is estimated from sample averages, confidence sets can be readily obtained.

Design-based Supervised Learning (DSL). DSL (Egami et al., 2023, 2024) adopts a *design-based sampling* scheme, which assumes $\pi(\hat{y}_i, \mathbf{x}_i) = \Pr(b_i = 1 \mid \hat{y}_i, \mathbf{x}_i) > 0$, where $b_i \in \{0, 1\}$ denotes whether document i is labeled by experts and where $\pi(\cdot)$ is known. The data is partitioned into K folds and used to cross-fit $\widehat{g}_k$, a model to predict y_i as a function of $\hat{y}_i$ and $\mathbf{x}_i$:

$$\widetilde{y}_i^k = \widehat{g}_k(\hat{y}_i, \mathbf{x}_i) + \frac{b_i}{\pi(\hat{y}_i, \mathbf{x}_i)}\big(y_i - \widehat{g}_k(\hat{y}_i, \mathbf{x}_i)\big).$$

Then, $\mathbb{E}\big[\widetilde{y}_i \mid \hat{y}_i, \mathbf{x}_i\big] = \mathbb{E}\big[y_i \mid \hat{y}_i, \mathbf{x}_i\big]$ regardless of misspecification of $\widehat{g}_k$ via double robustness.

Many estimands admit a moment equation form: $\mathbb{E}\big[m(y_i, \mathbf{x}_i; \theta)\big] = 0$ (e.g., maximum likelihood). DSL solves the empirical analogue of the moment condition with the debiased outcome, using $\sum_{i=1}^{N} m(\widetilde{y}_i, \mathbf{x}_i; \theta) = 0$, where each $\widetilde{y}_i$ is constructed as above. Cross-fitting and M-estimation theory then yield consistent "sandwich" estimators of variance, giving valid confidence intervals.

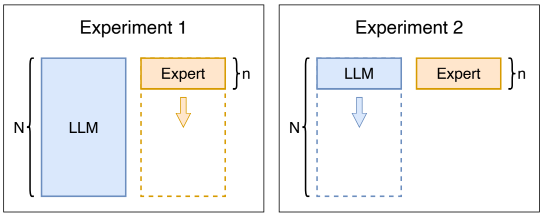

Figure 2: Setup for Experiments 1 and 2. Left panel (Experiment 1): Fixed N total samples with LLM annotations (blue); vary $n \ll N$ expert annotations (orange) for debiasing. Right panel (Experiment 2): Vary N total samples with LLM annotations (blue); fixed n expert annotations (orange) for debiasing.

Table 3 summarizes the key inferential properties of PPI and DSL, highlighting similarities and differences.

3 Methodology

Our analysis focuses on two experiments, which we use to benchmark and contrast the $\theta_{\dagger}$, PPI, and DSL estimators (see Figure 2). In both experiments, we focus on the coefficients of a binary logistic regression as our particular parameter of interest θ. Specifically, for each dataset, we create a downstream task relating four independent variables $x_1 \ldots x_4$ to a binary outcome y. The independent variables are either categorical or integers computed from text features. Each logistic regression, therefore, produces four coefficients $\beta_1 \ldots \beta_4$ and a y-intercept β_0 for a total of five parameters. See Appendix D for package use details and a link to the code.

Experiment 1. Our first experiment involves varying the number of expert annotations while keeping the total number of samples constant (see Figure 2, left). Our goal here is to answer the question: how do the debiasing methods improve with an increasing proportion of expert annotations? In other words, if one has a fixed number of data samples, how much budget should one allocate towards the expert annotations for debiasing?

For this experiment, we vary the number of expert samples logarithmically. We use a minimum of 200 expert annotations (below that threshold, debiasing methods became unstable). We additionally report the proportion of expert samples n_i/N rather than the absolute number in order to compare datasets of different sizes. We run 250 repetitions

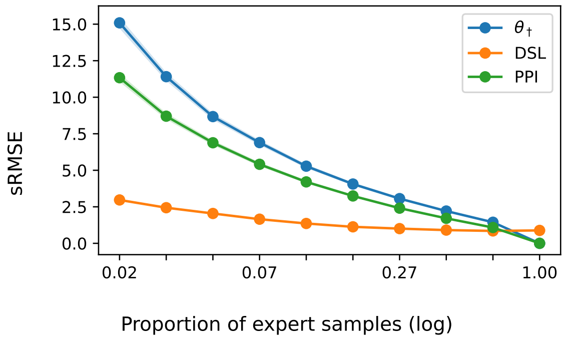

Figure 3: Results for Experiment 1 averaged over all datasets and annotation methods.

and report 2σ confidence intervals for each entry, dataset, and annotation procedure.

Experiment 2. In our second experiment, we vary the total number of samples while keeping the number of expert annotations fixed (see Figure 2, right). This setup targets scenarios where the expert annotation budget is limited but unlabeled data is abundant—such as in large news or social-media corpora—enabling practitioners to evaluate how additional unlabeled data enhances the effective sample size (i.e., the equivalent number of expert annotations needed to match the debiased estimator's precision) via debiasing methods. Specifically, we ask: given a fixed expert budget, how much does the effective sample size increase with more generated annotations? We repeat these experiments using 200, 1,000, and 5,000 expert annotations.

Like in Experiment 1, we vary the total number of samples logarithmically. The minimum number of total samples is defined by the number of available expert samples. The maximum number of total samples is determined by the size of the available dataset, which varies. We report the proportion of total samples with respect to the total number of available samples to facilitate comparison between datasets. We use 250 repetitions to estimate the 2σ confidence interval.

Datasets and Annotations. We replicate our experiments over four datasets: Multi-domain Sentiment, Misinfo-general, Bias in Biographies, and Germeval18 (see Appendix A). We also compare performance across four LLM-model classes: BERT, DeepSeek v3, Phi-4, and Claude 3.7 Sonnet (see Appendix C). Input variables are either additional annotations available from the original dataset or quantities derived from the text, such as the text length in characters. We compare PPI, DSL, and $\theta_\dagger$ with the same number of annotations. The datasets are available at `https://huggingface.co/datasets/nicaudinet/llm-debiasing-benchmark`.

Evaluation Metrics. We evaluate performance of debiasing methods by comparing the respective models against the reference model θ^*. Comparison between models is done using a standardized Root Mean Squared Error (sRMSE), which captures both bias and variance for a holistic performance assessment (see Appendix B). We standardize by scaling according to the reference model coefficients.

4 Results

In our experiments, we contrasted $\theta_\dagger$, PPI, and DSL with the reference model θ^*. The only difference between $\theta_\dagger$ and θ^* is that they are trained on a different number of expert annotations — $\theta_\dagger$ is trained on only the expert annotations that would have been given to one of the debiasing methods. Accordingly, the smaller the proportion of expert annotations given to the debiasing methods, the more inaccurate $\theta_\dagger$ becomes, which is reflected as a high sRMSE. As we increase the proportion of expert annotations, $\theta_\dagger$ converges towards θ^*, and we observe a monotonically decreasing sRMSE. At a proportion of 1, there is no difference between $\theta_\dagger$ and θ^* (the sRMSE is 0).

Results of Experiment 1 are displayed in Figure 3. We observe that PPI has a lower sRMSE than $\theta_\dagger$ for all data points. This is expected and guaranteed by theory under assumptions. DSL exhibits a significantly lower sRMSE than both PPI and $\theta_\dagger$ for almost all data points, showing that it is able to use the expert annotations more efficiently than both. However, the crossing at the end, when virtually all expert annotations contribute to the debiasing procedure, is curious: why do the DSL and $\theta_\dagger$ curves cross? When analyzing the performance of DSL by dataset (see Appendix F), we notice that the crossing phenomenon in the performance of DSL seems dataset-dependent. In particular, for the Misinfo-general dataset, DSL performs worse than both PPI and $\theta_\dagger$ for all samples.

A complete explanation of this phenomenon is still unknown. We have ruled out hypotheses related to preprocessing (e.g., centering); we have not identified obvious properties of the dataset that predict anomalous DSL estimates (e.g., agreement between expert and LLM annotations). Emerging

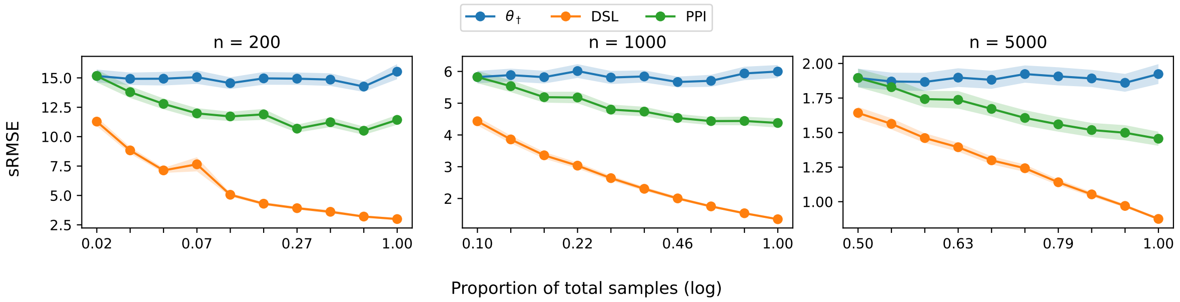

Figure 4: Results from the set of experiments varying the total number of samples, averaged over datasets and annotation methods. The x-axis shows the total number of samples (N) as a proportion of the total available samples in each dataset. The y-axis shows the sRMSE. The plots show results for $n = 200$, $n = 1000$, and $n = 5000$.

evidence, however, points to multicollinearity in the feature set as a contributing factor: DSL appears more sensitive to it than PPI, which is comparatively robust. In particular, we find in Figure 10 in Appendix G that the detrimental scaling of DSL in the Misinfo-general dataset is greatly improved when we remove highly collinear features. We also find in Figure 14 in Appendix H that bias decreases for all three methods when removing highly collinear features. Another remaining explanation is that, although PPI debiasing via subgradients leverages less information compared to DSL (which uses external sampling design knowledge), it avoids instabilities commonly associated with weighting estimators (Zubizarreta, 2015). Future work should explore these and related explanations, including how feature correlations interact with debiasing stability.

The results of Experiment 2 are displayed in Figure 4. Since $\theta_\dagger$ does not use generated annotations, its sRMSE remains constant as the dataset size grows. We also observe that PPI and DSL both outperform $\theta_\dagger$ in each of the three cases; performance of both tends to improve as we increase the total dataset size.

To translate these empirical findings into practical guidance for resource allocation in computational social science (Daoud and Dubhashi, 2023), we adapt the budgeting template from Broska et al. (2025)'s mixed subjects design, which optimizes the mix of costly expert annotations and cheaper LLM predictions based on their correlation and error profiles. Expert labeling on Amazon Mechanical Turk currently averages \$0.10 per label as of 2025 (W., 2025). For LLM inference on our largest corpus (Bias in Biographies, $N = 10{,}000$), assuming 300 input tokens per document (3 million total input tokens) and 10 output tokens per prompt (0.1 million total output tokens): Phi-4 incurs \$0.06 per million input tokens and \$0.14 per million output tokens, yielding a total cost of $\approx$ \$0.20 (equivalent to 2 expert labels). DeepSeek v3, at \$0.56 per million input tokens and \$1.68 per million output tokens, costs $\approx$ \$2 (20 expert labels). Claude 3.7 Sonnet, at \$3 per million input tokens and \$15 per million output tokens, costs $\approx$ \$10.50 (105 expert labels). BERT fine-tuning adds negligible cloud costs ($\approx$ \$0.50 USD, 5 expert labels), with debiasing computations (DSL/PPI) under \$1 total on standard hardware. In this 10,000-document scenario, the break-even n for cost (where n expert labels cost as much as full-model inference) is thus 2 for Phi-4, 20 for DeepSeek, 105 for Claude, and 5 for BERT—far below full expert annotation. Given our results (e.g., sRMSE < 0.2 at $n = 200$), we encourage practitioners to supplement their analyses with LLM predictions starting at these thresholds.

5 Conclusion

This study has investigated the performance of two LLM debiasing methods. On average, both debiasing methods produce models closer to a reference model than just using a small number of expert annotations. We also observe that DSL seems to significantly outperform PPI across datasets and annotation methods. However, DSL performance appears more inconsistent and dataset-dependent. Both DSL and PPI are more efficient than relying solely on a small, human-annotated dataset, so we encourage researchers to integrate debiasing methods into their analyses for improved estimation. While DSL outperforms PPI on most datasets, its performance is more inconsistent across them; therefore, we recommend reporting results from both methods until DSL's potential variability is better understood.

Limitations

Our study focuses on two specific debiasing methods, DSL and PPI, leaving out several other emerging techniques such as the recently proposed predict-then-debias (e.g., Kluger et al., 2025) and prediction-powered inference with inverse probability weighting (Datta and Polson, 2025). We only consider scenarios where the outcome variable requires annotation, thereby restricting the scope to single-task classification; we focus on binary outcomes as a simplifying assumption to facilitate benchmarking of the debiasing methods, though future work should extend this to multi-class or continuous outcomes. Future work should also consider situations where input variables are LLM-annotated or there is information leakage among variables (Daoud et al., 2022). In addition, while DSL and PPI can be applied to any M-estimator, our experimental evaluation of downstream tasks is currently limited to logistic regression (corresponding to the type of annotation we have considered). Future work should consider a variety of other statistical estimators, such as survival or hierarchical models.

Moreover, our experiments also concentrate on four datasets with relatively short texts in English or German, so further evaluation is needed in other languages, domains, and text lengths. Lastly, we assume expert-labeled data to be the ground truth; in practice, human annotations can also be noisy or inconsistent (Artstein and Poesio, 2008). Future work should examine how to extend or adapt methods such as DSL and PPI when the expert labels themselves may be subject to significant measurement error or domain shifts. We also acknowledge that the robustness of these debiasing methods under worst-case or adversarial settings remains an open problem.

Acknowledgments

We thank Naoki Egami, Katherine Keith, Brandon Stewart, and anonymous reviewers for helpful comments. This research was supported by the project *Countering Bias in AI Methods in the Social Sciences* under the Wallenberg AI, Autonomous Systems and Software Program – Humanity and Society (WASP-HS), funded by the Marianne and Marcus Wallenberg Foundation and the Marcus and Amalia Wallenberg Foundation. We acknowledge support from Alvis and Vera compute systems, provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS).

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Anastasios N. Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I. Jordan, and Tijana Zrnic. 2023a. Prediction-powered inference. *Science*, 382(6671):669–674.

Anastasios N Angelopoulos, John C Duchi, and Tijana Zrnic. 2023b. Ppi++: Efficient prediction-powered inference. *arXiv preprint arXiv:2311.01453*.

Anthropic. 2025. Claude 3.7 sonnet system card.

Lisa P. Argyle, 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.

Ron Artstein and Massimo Poesio. 2008. Survey article: Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Nicolas Audinet de Pieuchon, Adel Daoud, Connor Jerzak, Moa Johansson, and Richard Johansson. 2024. Can large language models (or humans) disentangle text? In *Proceedings of the Sixth Workshop on Natural Language Processing and Computational Social Science (NLP+CSS 2024)*, pages 57–67, Mexico City, Mexico. Association for Computational Linguistics.

Christopher A. Bail. 2024. Can generative AI improve social science? *Proceedings of the National Academy of Sciences*, 121(21).

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 440–447, Prague, Czech Republic. Association for Computational Linguistics.

David Broska, Michael Howes, and Austin van Loon. 2025. The Mixed Subjects Design: Treating Large Language Models as Potentially Informative Observations. *Sociological Methods & Research*, page 00491241251326865.

Adel Daoud and Devdatt Dubhashi. 2023. Statistical Modeling: The Three Cultures. *Harvard Data Science Review*, 5(1).

Adel Daoud, Connor T. Jerzak, and Richard Johansson. 2022. Conceptualizing Treatment Leakage in Text-based Causal Inference. In *Proceedings of the 2022 Conference of the North American Chapter of the*

Association for Computational Linguistics: Human Language Technologies, pages 5638–5645, Seattle, United States. Association for Computational Linguistics.

Jyotishka Datta and Nicholas G Polson. 2025. Prediction-powered inference with inverse probability weighting. *arXiv preprint arXiv:2508.10149*.

Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. 2019. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 120–128.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, United States. Association for Computational Linguistics.

Naoki Egami, 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. In *Advances in Neural Information Processing Systems*, volume 36, pages 68589–68601. Curran Associates, Inc.

Naoki Egami, Musashi Hinck, Brandon M. Stewart, and Hanying Wei. 2024. Using large language model annotations for the social sciences: A general framework of using predicted variables in downstream analyses. *Preprint from November 17, 2024*.

Mahmoud El-Haj, Paul Rayson, Steve Young, Andrew Moore, Martin Walker, Thomas Schleicher, and Vasiliki Athanasakou. 2016. Learning tone and attribution for financial text mining. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1820–1825, Portorož, Slovenia. European Language Resources Association (ELRA).

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

Shiho Kino, Yu-Tien Hsu, Koichiro Shiba, Yung-Shin Chien, Carol Mita, Ichiro Kawachi, and Adel Daoud. 2021. A scoping review on the use of machine learning in research on social determinants of health: Trends and research prospects. *SSM-population Health*, 15:100836.

Dan M Kluger, Kerri Lu, Tijana Zrnic, Sherrie Wang, and Stephen Bates. 2025. Prediction-powered inference with imputed covariates and nonuniform sampling. *arXiv preprint arXiv:2501.18577*.

Hao Lin and Yongjun Zhang. 2025. The risks of using large language models for text annotation in social science research. *arXiv preprint arXiv:2503.22040*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Daniel A McFarland and H Richard McFarland. 2015. Big data and the danger of being precisely inaccurate. *Big Data & Society*, 2(2):2053951715602495.

Satiyabooshan Murugaboopathy, Connor T. Jerzak, and Adel Daoud. 2025. Platonic Representations for Poverty Mapping: Unified Vision-Language Codes or Agent-Induced Novelty? *Preprint*, arXiv:2508.01109.

Priyanshu Priya, Mauajama Firdaus, and Asif Ekbal. 2024. Computational politeness in natural language processing: A survey. *ACM Computing Surveys*, 56(9):1–42.

Yanchuan Sim, Brice D. L. Acree, Justin H. Gross, and Noah A. Smith. 2013. Measuring ideological proportions in political speeches. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 91–101, Seattle, Washington, USA. Association for Computational Linguistics.

Petter Törnberg. 2024. Best practices for text annotation with large language models. *Sociologica*, 18(2):67–85.

Ivo Verhoeven, Pushkar Mishra, and Ekaterina Shutova. 2024. Yesterday's news: Benchmarking multi-dimensional out-of-distribution generalisation of misinformation detection models. *arXiv preprint arXiv:2410.18122*.

Admon W. 2025. How much do data annotation services cost? the complete guide 2025. Accessed: 2025-09-18.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. Overview of the GermEval 2018 shared task on the identification of offensive language. In *Proceedings of GermEval 2018, 14th Conference on Natural Language Processing (KONVENS 2018)*, Vienna, Austria.

Caleb Ziems, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2024. Can large language models transform computational social science? *Computational Linguistics*, 50(1):237–291.

José R Zubizarreta. 2015. Stable weights that balance covariates for estimation with incomplete outcome data. *Journal of the American Statistical Association*, 110(511):910–922.

A Datasets Description

We here present information about the datasets used in the analysis. All datasets are constructed by extracting a balanced subset of publicly available datasets. Following the simulation experiment from Egami et al. (Egami et al., 2023) we created four features $(x1, x2, x3, x4)$ used in the downstream task to predict the annotated output y. Details of the original datasets and feature creation are reported below. All datasets and LLM annotations used in the paper are available at https://huggingface.co/datasets/nicaudinet/llm-debiasing-benchmark.

Multi-domain Sentiment. The Multi-domain Sentiment dataset is a corpus of product reviews taken from Amazon (Blitzer et al., 2007). The dataset was originally used to investigate domain adaptation in sentiment classifiers. We used a subset taken from 6 domains, consisting of 11,914 reviews with two sets of annotations: a binary sentiment label (positive or negative) and a domain label (books, camera, DVD, health, music, or software). The dataset is balanced both in sentiment and topic labels.

For the downstream task, we use the sentiment label as the outcome variable. The independent variables are: the domain label (transformed to numeric values 0-5), the number of characters in the review, the number of space-separated words in the review, and the number of repetitions of the word "I" in the review.

Misinfo-general. The Misinfo-general dataset is a large corpus of British newspaper articles (Verhoeven et al., 2024) originally used to benchmark out-of-distribution performance of misinformation models. For our experiments, we selected articles from 2022 that were published in one of two venues: The Guardian UK or The Sun. We then balanced the dataset to have 5000 articles in each class.

For the downstream task, we use the venue as the binary outcome variable. The independent variables are: the number of characters in the article, the number of space-separated words in the article, the number of capital letters in the article, and the number of characters in the title of the article.

Bias in Biographies. The Bias in Biographies dataset is a corpus of short biographies originally used to study gender bias in occupational classification (De-Arteaga et al., 2019). The corpus consists of English-language online biographies from the Common Crawl, annotated with self-identified binary gender and occupation labels (with 28 categories), enabling analysis of implicit gender biases in textual representations. Here, $N =$ 10,000.

For the downstream task, we use the gender label as the outcome variable. This variable is balanced. Independent variables are: the occupation label (transformed to a numeric value, 0-27), the number of characters in the biography, the number of space-separated words in the biography, and the number of capital letters in the biography.

Germeval18. The Germeval18 dataset is a corpus of German tweets. It was used in the GermEval shared task on the identification of offensive language in 2018 (Wiegand et al., 2018). It is composed of a training and test set of documents with associated toxicity labels, totaling 5676 documents. We use a balanced subset of the data.

For the downstream task, we use the binary toxicity label as the outcome variable. The independent variables are: the number of characters in the tweet, the number of space-separated words in the tweet, the number of capital letters in the tweet, and the number of "@" characters in the tweet.

B Details of Evaluation Metrics

We define the standardized Root Mean Squared Error ($sRMSE$) as:

$$\text{sRMSE}(\theta; d) = \sqrt{\mathbb{E}\left[\left(\frac{\theta - \theta_d^*}{\theta_d^*}\right)^2\right]}.$$

where θ are the coefficients from the model under test and θ_d^* are the coefficients from the reference model for dataset d.

C Model Details

BERT + Logistic Regression. As a representative of supervised approaches, we fine-tune a pretrained BERT encoder (Devlin et al., 2019) on the expert-labeled subset to obtain contextual representations $\mathbf{h}_i = \text{BERT}(d_i)$, which are then passed to a logistic regression head trained to predict y_i.

Large Language Models. We also generate annotations with three language models: Microsoft Phi-4 (Abdin et al., 2024), DeepSeek v3 (Liu et al., 2024), and Claude 3.7 Sonnet (Anthropic, 2025). Phi-4 is a 14B open-weight model, which we ran locally with the default temperature of 1.0. We used the paid DeepSeek and Anthropic APIs to access DeepSeek v3 and Claude 3.7 Sonnet, respectively. We paid approximately $10 for the DeepSeek API and approximately $100 for the Anthropic API (2025 USD). We used the default decoding mechanism and temperature of 1.0 for both models. The prompts used to generate the labels are available in Appendix E. In some cases, the annotations generated for a small number of the documents did not fit the annotation schema. These samples were ignored.

D Package and Code Details

For the classical logistic regression, we use the `scikit-learn` Python package. We use no regularization and set the maximum iterations to 1000.

For DSL, we use the `dsl` R package developed by the original paper authors for both experiments. We leave the parameters to their default settings.

For PPI, we use the `ppi_py` Python package (Angelopoulos et al., 2023b)—an implementation of the `PPI+` framework by the original PPI authors—for both experiments. We also leave the parameters to their default settings.

The source code for the experiments is available at `https://github.com/nicaudinet/llm-debiasing-benchmark`.

E Prompts

Figures 5, 6, 7, and 8 show the prompt templates used to make prompts for LLM annotation. The prompt templates were specialized for each dataset since each dataset corresponds to a different annotation task. However, the structure of the prompt templates was kept the same: first, a short description of the task, then an explanation of the formatting with two simple examples, and finally the document to classify. For each dataset, we also include a system prompt (see Table 1).

F Results by Dataset

Figure 9 showcases the results for Experiment 1 broken down by dataset. In all four datasets, PPI outperforms $\theta_\dagger$ for all data points. DSL outperforms PPI and $\theta_\dagger$ in most cases. However, 3 of the datasets exhibit cross-over behavior for higher proportions of expert samples, with Misinfo-general being the outlier where DSL performs significantly worse than both PPI and $\theta_\dagger$ for all data points.

G Results Removing Collinear Variables

Here we investigated the dependence of DSL on correlations between variables. Some of the features we chose for the datasets were highly collinear (e.g., the number of characters and the number of space-separated words in a piece of text). We gather the Pearson r^2 correlations in Table 2. For each dataset and annotation type, we proceeded to remove collinear features by finding feature pairs with r^2 above 0.9 and removing the latter variable (for instance, we remove $x3$ for the Multi-domain Sentiment dataset).

The results of running Experiment 1 with the reduced datasets are shown in Figure 10 and Figure 11. The results show that removing the collinear features mitigated the cross-over effect observed with DSL for higher proportions of expert samples.

H Standardized Bias Plots

We report the performance of the debiasing methods for Experiment 1 in terms of the standardized bias, following the simulation experiment from Egami et al. (Egami et al., 2023). The standardized bias is defined similarly to the sRMSE as:

$$\text{Standardized Bias}(\theta; d) = \mathbb{E}\left[\frac{\theta - \theta_d^*}{\theta_d^*}\right]$$

where θ are the coefficients from the downstream task and θ_d^* are the coefficients from the reference model for dataset d.

Figure 12 and Figure 13 show the results for the original experiment with four features. We notice that PPI consistently produces slightly less biased coefficients with smaller confidence intervals than $\theta_\dagger$. DSL is more variable, producing much less biased coefficients for some datasets (Multi-domain Sentiment, Bias in Biographies) but much more biased coefficients in others (Misinfo-general, Germeval18).

Figure 14 and Figure 15 show the results for the experiment from Appendix G where highly correlated features are removed. Compared to using all features, we notice a significant performance increase in all three methods. In particular, the coefficients produced by DSL are more stable.

I Comparison of DSL and PPI

A comparison of DSL and PPI can be found in Table 3.

```
Classify the following review as either:
- POSITIVE if the review indicates an overall positive sentiment
- NEGATIVE if the review indicates an overall negative sentiment

Give no other explanation for your classification, only output the label.

Here are two examples of the formatting I would like you to use, where
< REVIEW_TEXT > is a stand-in for the article text:

< REVIEW_TEXT >

CLASSIFICATION: POSITIVE

< REVIEW_TEXT >

CLASSIFICATION: NEGATIVE

Here's the review to classify:

{text}

CLASSIFICATION:
```

Figure 5: The prompt template used to annotate documents from the Multi-domain Sentiment dataset, where `{text}` is substituted with the document in question.

```
Classify the following article as either:
- THESUN if it is likely to have been published in the British tabloid newspaper
  The Sun
- THEGUARDIAN if it is likely to have been published in the British daily
  newspaper The Guardian

Give no other explanation for your classification, only output the label.

Here are two examples of the formatting I would like you use, where
< ARTICLE_TEXT > is a stand-in for the article text:

< ARTICLE_TEXT >

CLASSIFICATION: THESUN

< ARTICLE_TEXT >

CLASSIFICATION: THEGUARDIAN

Here's the article I would like you to classify:

{text}

CLASSIFICATION:
```

Figure 6: The prompt template used to annotate documents from the Misinfo-general dataset, where `{text}` is substituted for the document in question

```
Classify the following textual biographies as either:
- MALE if the subject is likely to be male
- FEMALE if the subject is likely to be female

Give no other explanation for your classification, only output the label.

Here are two examples of the formatting I would like you use, where < BIOGRAPHY_TEXT >
is a stand-in for the textual biography:

< BIOGRAPHY_TEXT >

CLASSIFICATION: MALE

< BIOGRAPHY_TEXT >

CLASSIFICATION: FEMALE

Here's the textual biography I would like you to classify:

{text}

CLASSIFICATION:
```

Figure 7: The prompt template used to annotate documents from the Bias in Biographies dataset, where `{text}` is substituted for the document in question

```
Classify the following German tweets as either:
- OFFENSIVE if the tweet is likely to contain an offense or be offensive
- OTHER if the tweet is _not_ likely to contain an offense or be offensive

Give no other explanation for your classification, only output the label.

Here are two examples of the formatting I would like you use, where < TWEET_TEXT >
is a stand-in for the text of the tweet:

< TWEET_TEXT >

CLASSIFICATION: OFFENSIVE

< TWEET_TEXT >

CLASSIFICATION: OTHER

{make_examples(examples)}

Here's the German tweet I would like you to classify:

{text}

CLASSIFICATION:
```

Figure 8: The prompt template used to annotate documents from the Germeval18 dataset, where `{text}` is substituted for the document in question

Dataset	**System Prompt**
Multi-domain Sentiment	“You are a perfect sentiment classification system”
Misinfo-general	“You are a perfect newspaper article classification system”
Bias in Biographies	“You are a perfect biography classification system”
Germeval18	“You are a perfect German tweet classification system”

Table 1: The system prompts used to annotate the various datasets

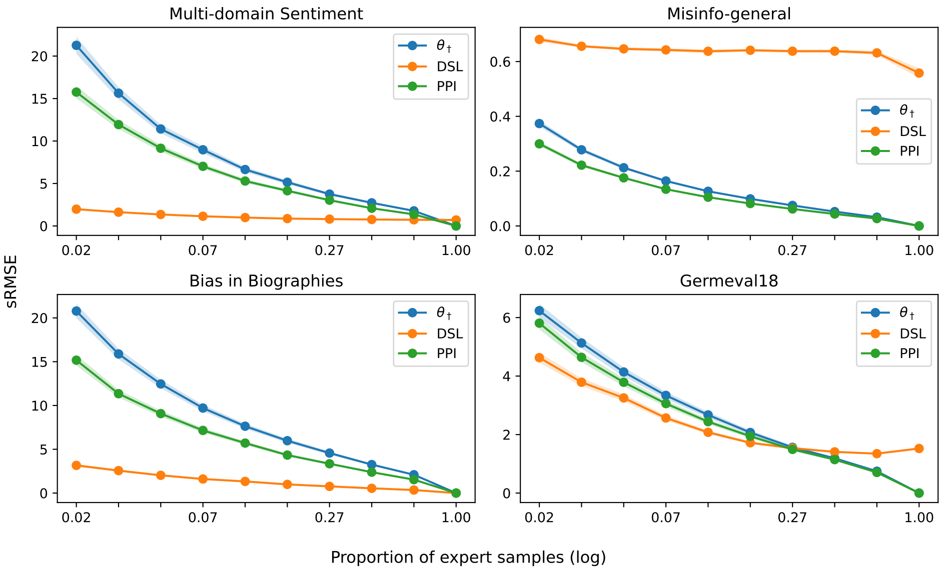

Figure 9: Results from the set of experiments varying the proportion of expert samples, aggregated per dataset.

Dataset	LLM	x1,x2	x1,x3	x1,x4	x2,x3	x2,x4	x3,x4
Multi-domain Sentiment	bert	0.008	0.007	0.013	**0.995**	0.253	0.283
	deepseek	0.008	0.007	0.013	**0.995**	0.255	0.284
	phi4	0.007	0.007	0.012	**0.995**	0.253	0.283
	claude	0.007	0.007	0.012	**0.995**	0.252	0.282
Misinfo-general	bert	**0.995**	0.617	0.026	0.618	0.021	0.002
	deepseek	**0.995**	0.621	0.025	0.622	0.021	0.002
	phi4	**0.995**	0.617	0.026	0.618	0.021	0.002
	claude	**0.995**	0.617	0.026	0.618	0.022	0.002
Bias in Biographies	bert	0.000	0.000	0.001	**0.965**	0.351	0.329
	deepseek	0.000	0.000	0.001	**0.964**	0.346	0.325
	phi4	0.000	0.000	0.001	**0.965**	0.351	0.329
	claude	0.000	0.000	0.001	**0.965**	0.351	0.329
Germeval18	bert	0.349	0.250	0.190	**0.961**	0.685	0.653
	deepseek	0.282	0.161	0.096	**0.940**	0.470	0.452
	phi4	0.332	0.223	0.161	**0.956**	0.615	0.590
	claude	0.284	0.164	0.101	**0.941**	0.487	0.468

Table 2: The Pearson r^2 correlations between each pair of features for each dataset and LLM annotations. Pairs of features with correlations above the threshold are highlighted. Correlations for the same dataset may differ slightly between LLM annotations because the LLMs failed to annotate a small portion of the samples, which we discarded.

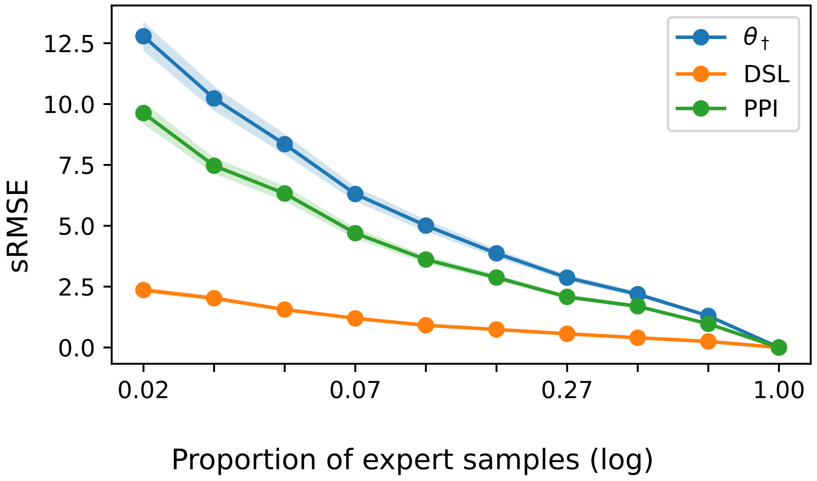

Figure 10: Performance of debiasing methods in Experiment 1 after removing highly collinear features ($r^2 > 0.9$) averaged over all datasets.

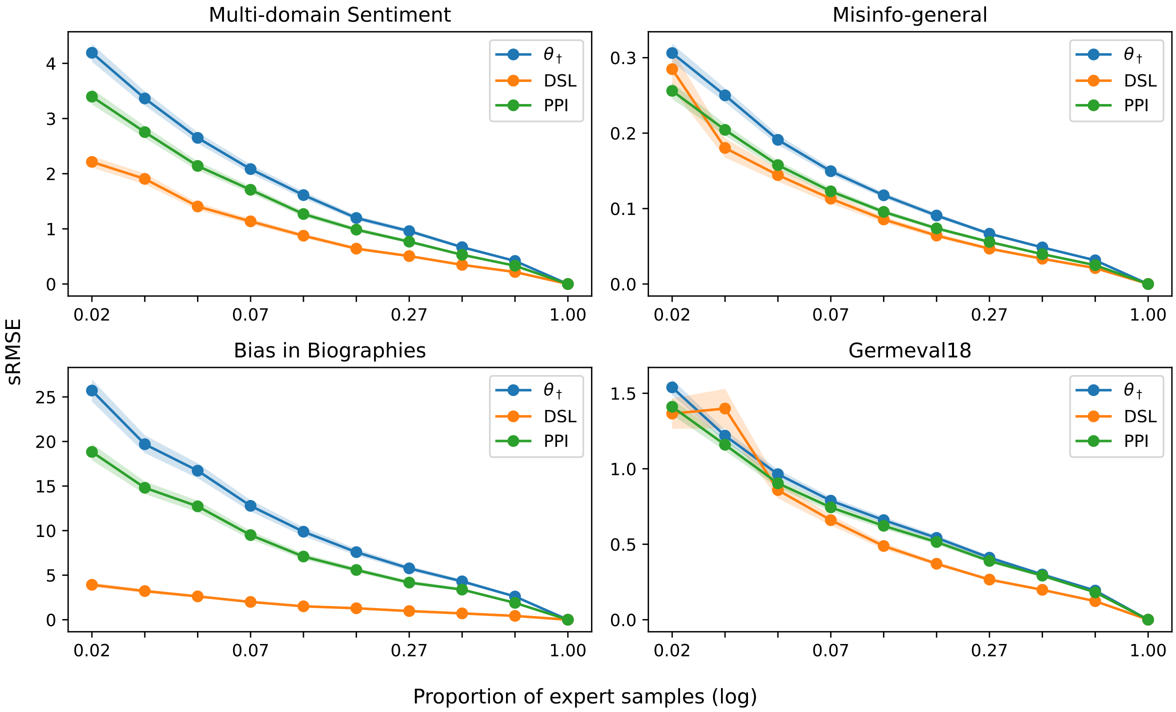

Figure 11: Performance of debiasing methods in Experiment 1 after removing highly collinear features ($r^2 > 0.9$) for each dataset.

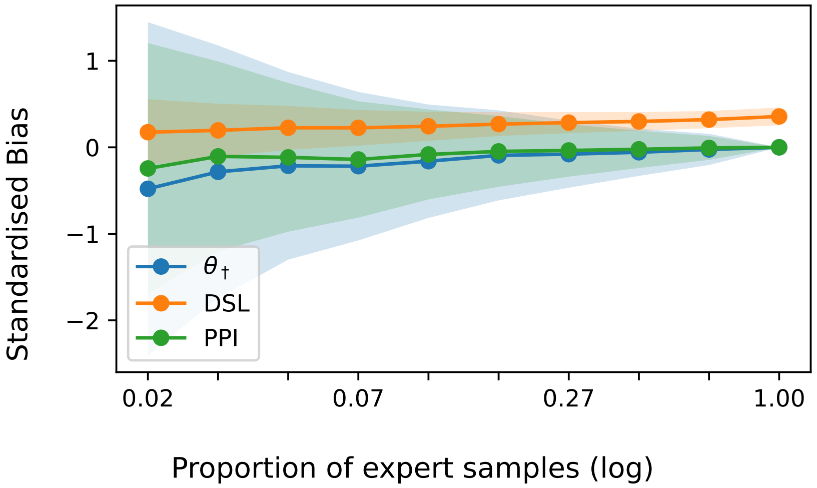

Figure 12: Performance of debiasing methods for Experiment 1 in terms of the standardized bias aggregated over all datasets.

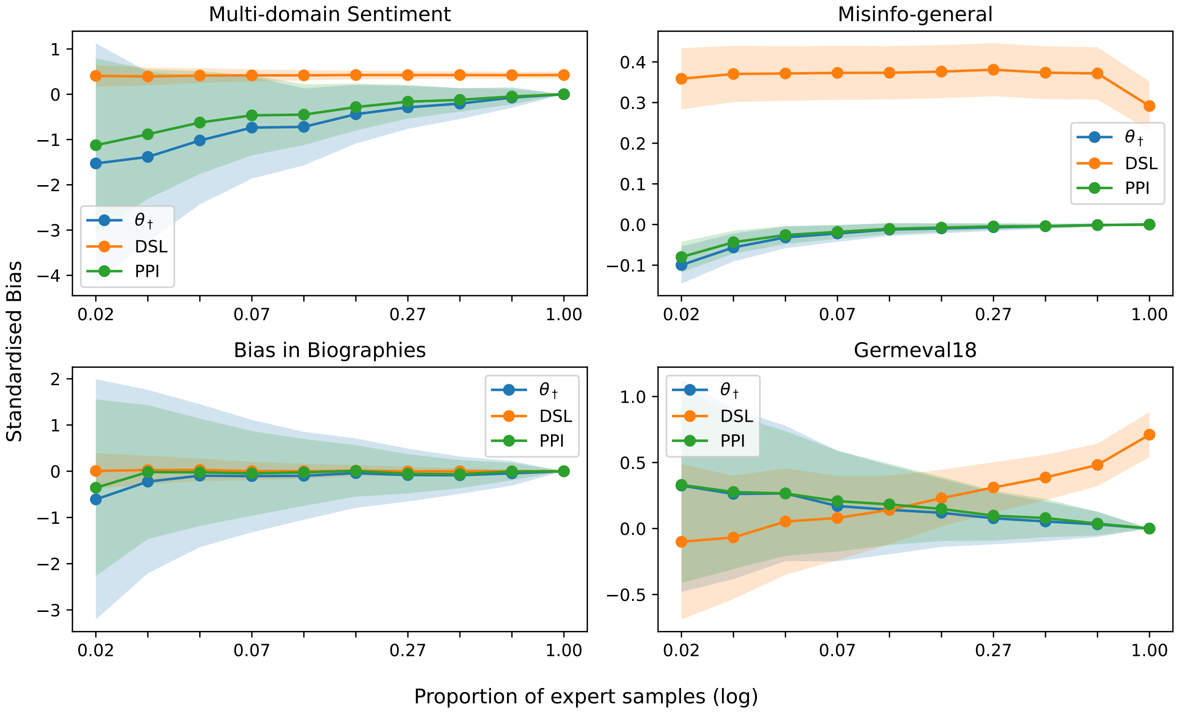

Figure 13: Performance of debiasing methods for Experiment 1 in terms of the standardized bias for each dataset.

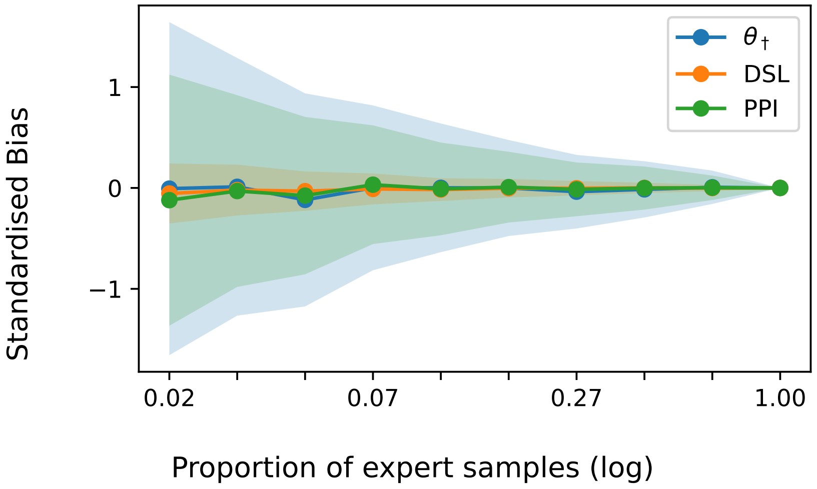

Figure 14: Performance of debiasing methods for Experiment 1 where highly correlated variables are removed, in terms of the standardized bias and aggregated over all datasets.

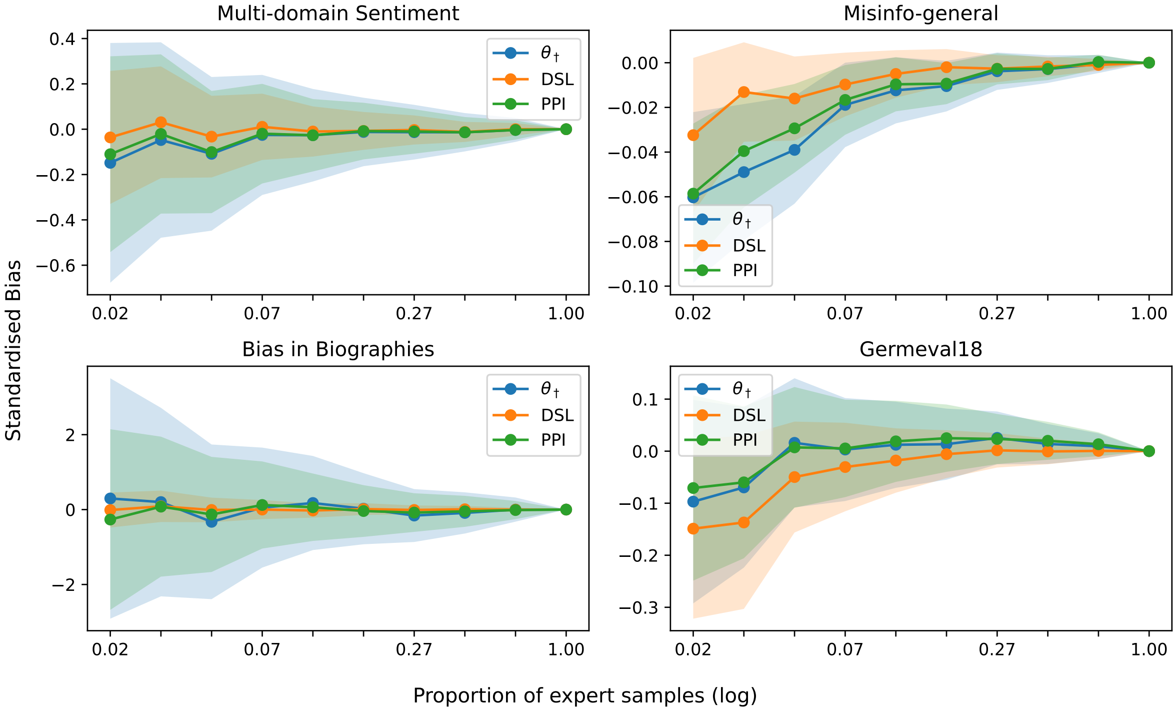

Figure 15: Performance of debiasing methods for Experiment 1, where highly correlated variables are removed, in terms of the standardized bias.

Table 3: Comparison of PPI (Angelopoulos et al., 2023a) and DSL(Egami et al., 2023) for debiasing ML predictions in downstream parameter estimation.

Method	**Bias Correction Mechanism**	**Guarantees**	**Variance Components**
PPI	First-order adjustment using estimating equation gradients (influence function) to offset systematic prediction errors.	Asymptotic validity holds irrespective of prediction model specification, provided large-sample coverage.	Aggregates prediction uncertainty and gradient-based correction variability; no design-specific terms.
DSL	Post-estimation pseudo-outcome via doubly robust imputation, relying on specified selection probabilities.	Consistency if at least one of outcome regression or selection model is accurate; requires positive bounded probabilities.	Includes augmented regression variance plus inverse-probability weighting effects, which may be amplified under irregular sampling (e.g., covariate overlap, multicollinearity).