

# Can Language Models Boost the Power of Randomized Experiments Without Statistical Bias? \*

Xinrui Ruan<sup>1</sup>    Xinwei Ma<sup>2</sup>    Yingfei Wang<sup>3</sup>    Waverly Wei<sup>4</sup>    Jingshen Wang<sup>1</sup>

<sup>1</sup>Division of Biostatistics, University of California, Berkeley

<sup>2</sup>Department of Economics, University of California, San Diego

<sup>3</sup>Michael G. Foster School of Business, University of Washington

<sup>4</sup>Department of Data Sciences and Operations, University of Southern California

October 7, 2025

Revised: August 23, 2026

## Abstract

Randomized controlled trials (RCTs) are widely adopted for causal inference, yet cost and sample-size constraints limit power. We introduce **CALM** (*Causal Analysis leveraging Language Models*), a statistical framework that integrates insights generated by large language models (LLMs) into the analysis of RCTs using established causal estimators to increase precision while preserving statistical validity. In particular, CALM treats LLM-generated outputs as auxiliary prognostic information and corrects their potential bias via a heterogeneous calibration step that residualizes and optimally reweights predictions. We prove that CALM remains consistent even when LLM predictions are biased and achieves efficiency gains over augmented inverse probability weighting estimators for various causal estimands. In particular, CALM develops a few-shot variant that aggregates predictions across randomly sampled demonstration sets. The resulting U-statistic-like predictor restores i.i.d. structure and also mitigates prompt-selection variability. Empirically, in simulations calibrated to a mobile-app depression RCT, CALM delivers lower variance relative to other benchmarking methods, is effective in zero- and few-shot settings, and remains stable across prompt designs. By principled use of LLMs to harness unstructured data and external knowledge learned during pretraining, CALM provides a practical path to more precise causal analyses.

*Keywords:* Artificial intelligence; Few-shot learning; Large language models; Randomized controlled trials; Treatment effects.

---

\*We thank the seminar and conference participants for their insightful comments, from which this paper has greatly benefited. Xinwei Ma acknowledges support from the National Institutes of Health (R01-AG089512). Jingshen Wang acknowledges support from Neuromatch, the National Science Foundation (NSF DMS-2239047), and the National Institutes of Health (R01-AG089512).

# 1 Introduction

Randomized controlled trials (RCTs) are widely adopted to evaluate the effectiveness of socioeconomic, business, biomedical, and behavioral interventions. This is achieved either by estimating an average effect or by performing subgroup analyses to explore treatment effect heterogeneity and thereby facilitate targeted interventions. Despite their prevalence, most RCT datasets are analyzed in isolation without incorporating external information. Additionally, regression analyses remain the standard practice with a set of hand-picked pre-intervention covariates, primarily due to the simplicity of such models. This approach, however, inevitably leaves considerable baseline information unexploited, including multimodal variables such as open-ended survey responses, product images, clinical notes, facial and gestural cues.

In this paper, we introduce a versatile framework for analyzing randomized controlled trials, accommodating pre-intervention covariates of multimodal nature without requiring specialized computational architectures. As such, our methods are simple to implement and remain accessible to a broad audience of applied researchers. Specifically, the first step in our framework leverages modern large language models (LLMs) to generate predictions of the potential outcomes using design information and baseline covariates that can be tabular or unstructured. LLMs are self-supervised models trained on broad, heterogeneous corpora, yielding general-purpose representations adaptable to many downstream tasks. Integrating LLM-generated predictions into causal inference methods therefore offers opportunities to boost the power of RCTs by leveraging both knowledge acquired during pretraining and information extracted from baseline covariates.

Despite the popularity of language models in prediction tasks, it remains a critical challenge to develop rigorous and statistically valid causal inference methods incorporating LLM-generated predictions. This is because these predictions are typically biased for the true potential outcomes, and moreover, their accuracy can exhibit substantial heterogeneity depending on individual characteristics. As a result, naively plugging LLM-based predictions into causal estimators can induce bias and lead to problematic statistical inference. Our framework, **Causal Analysis leveraging Language Models (CALM)**, tackles these issues via three key components. First, we employ few-shot learning with demonstrative examples resampling to improve prediction relevance and precision. We then introduce a residualization step coupled with heterogeneous calibration weighting to avoid intro-

ducing statistical bias and deliver more efficient treatment effect estimation. We demonstrate such efficiency gains both through rigorous theoretical investigations and via an array of numerical evidence, including a carefully crafted simulation study based on calibrated data and a re-evaluation of a behavioral trial. Finally, the estimator takes a Neyman-orthogonal form so that it admits an asymptotically normal distribution, thereby enabling valid statistical inference. In the remainder of this introduction, we elaborate on these features of our methods, further situating our contribution within the literature.

*Few-shot Learning with Demonstrative Examples Resampling.* A key strength and novelty of CALM is its ability to incorporate few-shot learning (a.k.a. in-context learning) to improve prediction relevance and precision. Specifically, the idea is to include a few demonstrative examples in the prompt, which has been well-documented in the literature due to its simplicity and effectiveness for rapid task adaptation [5, 19, 31, 41]. However, few-shot learning is not without its limitations: the generated predictions can be sensitive to the examples and their ordering [49]. Such dependence and lack of invariance may introduce unnecessary noise and bias into the predictions, threatening downstream treatment effect estimation and statistical inference. Another challenge of incorporating few-shot learning is that the demonstrative examples have to be chosen in-sample, and as a result, this induces correlations among the predicted potential outcomes. Unfortunately, it is generally not possible to quantify such correlations, rendering any post hoc adjustment to the statistical inference procedure infeasible.

Our proposed CALM tackles these challenges by augmenting few-shot learning with resampling-based prediction aggregation. On the practical side, the method remains easy to implement: one only needs to regenerate predictions a few times, each with randomly sampled demonstrative examples. This can easily be done through a chat-based platform for small datasets, or otherwise via a standard API. Theoretically, we show that averaging predictions across multiple small, randomly drawn sets of demonstrative samples yields an LLM-based predictor with a U-statistic-like structure, which is not only robust to demonstrative sample selection but also behaves in an iid manner in large samples. As a result, both our treatment effect estimator and its companion inferential method remain straightforward, making them accessible to practitioners.

*Residualization with Heterogeneous Calibration Weights.* Despite their popularity, language model predictions are by no means perfect: they can be imprecise and even worse, biased in ways

that are difficult to characterize. To remain agnostic about the nature of such bias, our proposed CALM incorporates the residualized version of the LLM predictions, where tabular covariates are partialled out from the predicted potential outcomes via nonparametric regressions. A natural concern is whether residualization leads to information loss. Fortunately, this is not the case. Our final estimator builds on the classical augmented inverse probability weighting method, which already incorporates the tabular covariate information efficiently. Residualization therefore preserves the predictive information captured by the unstructured covariates while simultaneously removing any potential bias in the language model predictions.

Incorporating mean-zero quantities into an estimator does not, by itself, guarantee efficiency gains. To ensure that our approach improves upon classical methods, we recognize that prediction accuracy from language models may be heterogeneous, varying across treatment arms and covariate values. We therefore introduce an additional calibration weighting scheme. The idea is simple yet powerful: by assigning weights that are both treatment- and covariate-specific, the estimator fully exploits the predictive power of the unstructured covariates and the underlying language model, leading to improved estimation efficiency and statistical power. This is borne out in both our theoretical analysis and our numerical evidence.

*Neyman Orthogonal Estimating Equation.* Our proposed CALM requires the estimation of several nuisance functions, including the conditional first and second moments of the true and predicted potential outcomes. Following the literature, we recommend the use of sample splitting together with cross-fitting, which is enabled by our carefully crafted estimating equation that is fully Neyman orthogonal with respect to the nuisance functions. It is worth noting that sample splitting and cross-fitting may not be necessary, depending on the specific methods employed for first-step estimation. Although we do not pursue such directions, see [8, 9, 10, 23, 39] for relevant examples.

In addition to its methodological and theoretical advances, our work provides systematic empirical evidence on the practical utility of CALM. Using synthetic data calibrated to the BRIGHTEN study with both structured and unstructured information, we demonstrate that CALM yields more accurate estimates of causal effects, achieving lower bias and reduced variance compared to benchmark methods. We further demonstrate that CALM remains effective in both zero- and few-shot settings without domain-specific fine-tuning, adapts well to heterogeneous covariate strata, and

avoids the under-coverage issues that arise when LLM predictions are naively incorporated as covariates in benchmark methods. In the BRIGHTEN case study, CALM not only yields tighter confidence intervals than benchmark methods, but also detects subgroup treatment effects (e.g., among female Hispanic participants) that benchmark methods miss, indicating its potential to enhance the discovery of treatment heterogeneity. Finally, robustness checks across multiple LLMs and diverse prompt-engineering strategies confirm that CALM’s empirical performance is stable to design choices, thereby reducing practical barriers to applying LLMs in real-world experiment analysis.

Before closing this introduction, we briefly discuss several related strands of literature to which this paper contributes. To start, CALM is motivated by recent advances in LLMs, which have demonstrated strong in-context learning abilities, encompassing zero-shot and few-shot prediction, that make them adaptable to new tasks with minimal supervision [5, 15, 44, 40]. In scientific and biomedical domains, domain-adapted models further improve relevance and accuracy [28, 42], while multimodal extensions enable joint reasoning over text, images, and audio [1, 24, 26, 32, 36]. These capabilities make LLMs well-suited for extracting rich information from the unstructured data increasingly collected in modern RCTs, including clinical notes, medical images, and participant narratives, now standard across oncology, mental health [35, 38], dementia [20], and healthcare delivery [6, 22]. Yet most LLM-based prediction pipelines offer little uncertainty quantification or statistical guarantees, which are precisely what causal inference in high-stakes biomedical settings demands.

As CALM incorporates LLM-based predictions in a principled manner without introducing additional bias, it also contributes to the growing literature on causal inference with potentially biased auxiliary information and prediction-powered inference (PPI) [2, 4, 7, 11, 16, 27, 47]. Originally developed for settings with missing labels, PPI leverages black-box predictions of unobserved outcomes to improve estimation efficiency, and has since been extended to randomized experiments [14, 34], with refinements including optimally tuned parameters [3], stratified tuning [17], and cross-fitted machine learning predictions [50]. Unlike CALM, however, these methods do not guarantee efficiency gains and may even incur efficiency loss when auxiliary predictions are poor; moreover, they are not designed to handle the correlated predictions that arise from LLM few-shot learning.

Our work also relates to the literature on improving the efficiency of randomized experiments

through covariate adjustment. Classical approaches such as regression adjustment and stratification are well known to reduce variance without compromising statistical validity [18, 25], but they are limited in their ability to utilize unstructured data and thus may not achieve full efficiency. More recently, double/debiased machine learning [12] leverages flexible machine learning together with cross-fitting. However, existing flexible machine learning tools, such as random forests and boosting, are primarily limited to structured data and are not directly applicable to unstructured data [13, 20, 35]. In addition, the inclusion of many covariates may lead to biases in the asymptotic normal distribution and therefore renders standard normal-based inference invalid [10].

The rest of the paper unfolds as follows. Section 2 introduces our theoretical framework and notation before presenting the proposed method. To keep the exposition clear, the section focuses on a single potential outcome, allowing us to highlight the key components of our approach without cumbersome notation. Section 3 expands on this setup to cover zero-shot prediction with CALM, and the estimation of conditional and unconditional average treatment effects. Section 4 presents our theoretical results alongside their formal assumptions. Finally, Sections 5 and 6 showcase our numerical results, which include a carefully designed simulation study with synthetic data and a re-analysis of a randomized controlled trial.

## 2 Setup and Methodology

We discuss in Subsection 2.1 the heuristic motivation for CALM, introduce the residualized, calibration-weighted correction term that incorporates LLM-based counterfactual predictions, and characterize the resulting efficiency gain over AIPW. Next, Subsection 2.2 details the step-by-step implementation of CALM under few-shot learning. Finally, Subsection 2.3 extends this framework to settings where outcome models are parsimonious and potentially misspecified, showing that CALM retains a guaranteed efficiency gain over AIPW even when calibration is restricted to a constrained class. We then specialize this result to the practically important case of covariate discretization.

Consider a randomized controlled trial with  $n$  units, labeled by  $i = 1, 2, \dots, n$ . For each participant, we observe their treatment status,  $T_i \in \{1, 2, \dots, k\}$ , a scalar outcome,  $Y_i \in \mathbb{R}$ , and pre-intervention covariates, including both structured  $X_i \in \mathcal{X} \subseteq \mathbb{R}^p$  (such as age, educational

background, etc.) and unstructured  $Z_i \in \mathcal{Z}$  (e.g., open-ended surveys). As such, the information observed by the analyst is  $O_i = (Y_i, T_i, X_i, Z_i)$ . Following the standard potential outcomes framework, the observed outcome is  $Y_i = \sum_{t=1}^k \mathbb{1}(T_i = t)Y_i(t)$ , and  $(Y_i(1), \dots, Y_i(k))$  denote the  $k$  potential outcomes for unit  $i$ . To facilitate further discussion, we also introduce the propensity scores and the conditional expectations of the potential outcomes:

$$e_t(X_i) = \mathbb{P}[T_i = t|X_i] \quad \text{and} \quad \mu_t(X_i) = \mathbb{E}[Y_i(t)|X_i] \quad \text{for } t = 1, 2, \dots, k.$$

Since the data arise from a randomized trial, we assume that the propensity scores are known. We note, however, that our proposed method readily extends to settings with unknown  $e_t(X_i)$  with minor changes to the underlying technical conditions. The framework accommodates both complete randomization, in which the treatment probability is constant (say,  $e_t(X_i) = 1/k$ ), and stratified experiments, where units with different pre-intervention covariates may receive different treatment probabilities. For expositional clarity, the formal analysis below adopts an i.i.d. superpopulation formulation, while the CALM construction itself remains applicable under fixed-allocation complete or stratified designs. In addition, our methodology allows some of the unstructured covariates to be missing, thanks to the versatility of language models in handling partially observed inputs. Such flexibility is particularly relevant, as it is common for participants not to complete a survey.

This paper develops estimation strategies for both unconditional and conditional average treatment effects that incorporate structured and unstructured information by leveraging language models. The parameters of interest are

$$\tau_{t,t'} = \mathbb{E}[Y_i(t) - Y_i(t')] \quad \text{and} \quad \tau_{t,t'}(x) = \mathbb{E}[Y_i(t) - Y_i(t')|X_i = x],$$

for two treatment levels  $t$  and  $t'$ . However, we focus in this section on the simpler estimand  $\mu_t = \mathbb{E}[Y_i(t)]$  to keep the exposition clear. Once the main idea and implementation details have been introduced, the same strategy carries over naturally to the estimation of  $\tau_{t,t'}$  (see Section 3).

## 2.1 Motivation and Efficiency Gains of CALM

To provide some heuristic justification for our proposed CALM, first consider  $n^{-1} \sum_{i=1}^n Y_i(t)$ . This “estimator” is ideal as it employs the sample average to estimate a population expectation. However, it is clearly infeasible:  $Y_i(t)$  is only observed for individuals who are assigned to treatment arm  $t$ . Now rewrite it into  $n^{-1} \sum_{i=1}^n \{Y_i(t) - \mu_t(X_i) + \mu_t(X_i)\}$ . Importantly,  $\mu_t(X_i)$  is estimable for each unit, so it is  $Y_i(t) - \mu_t(X_i)$  that is subject to missingness. This is the motivation behind the augmented inverse probability weighting (AIPW) approach

$$\bar{\mu}_{t,\text{AIPW}} = \frac{1}{n} \sum_{i=1}^n \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} (Y_i - \mu_t(X_i)) + \mu_t(X_i),$$

which is widely adopted in practice. Strictly speaking, the above is still infeasible as it involves the unknown nuisance function  $\mu_t(X_i)$ . See below for implementation details of CALM, which also applies to AIPW. In the absence of unstructured covariates, the AIPW estimator is known to be efficient [8, 21].

As the discussion so far has made clear, the information in  $Y_i(t)$  can be partitioned into the part captured by the structured covariates through  $\mu_t(X_i)$  and the part that cannot, and it is the latter that is only available for units in treatment arm  $t$ . Our proposal is to employ language models to predict such information for units in other treatment arms via unstructured data  $Z_i$ . For now, assume we are able to generate LLM-based predictions of the potential outcomes, denoted  $Y_i^\dagger(t)$  for each  $i = 1, 2, \dots, n$  (see Step 2 below for implementation details, including our novel few-shot learning prediction with demonstrative examples resampling). The proposed estimator takes the form (strictly speaking, below is the asymptotically linear representation of our CALM estimator)

$$\bar{\mu}_{t,\text{CALM}} = \frac{1}{n} \sum_{i=1}^n \underbrace{\frac{\mathbb{1}(T_i = t)}{e_t(X_i)} (Y_i - \mu_t(X_i)) + \mu_t(X_i)}_{\text{AIPW}} + \underbrace{\left(1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)}\right) (Y_i^\dagger(t) - \mu_t^\dagger(X_i)) \omega_t(X_i)}_{\text{CALM improvement}}. \quad (1)$$

We start with the middle term,  $Y_i^\dagger(t) - \mu_t^\dagger(X_i)$ , which incorporates LLM-based counterfactual predictions without introducing statistical bias. Importantly, our method does not require the predicted counterfactuals  $Y_i^\dagger(t)$  to be “consistent” or “unbiased” for the true  $Y_i(t)$ , as such conditions are difficult to justify in practice. Such robustness is achieved by residualizing the prediction

against its own conditional expectation,

$$\mu_t^\dagger(X_i) = \mathbb{E}[Y_i^\dagger(t)|X_i],$$

so that this residual is mean zero by construction, avoiding bias in the estimating equation regardless of how  $Y_i^\dagger(t)$  relates to  $Y_i(t)$ . This construction is also motivated by our earlier discussion:  $Y_i(t) - \mu_t(X_i)$  is exactly what is missing for units not in treatment arm  $t$ , and our approach leverages language models and unstructured information to “recover” it.

The residualized LLM predictions are followed by the calibration weights  $\omega_t(X_i)$ , which deliver guaranteed efficiency gains over the AIPW approach. The term  $\omega_t(X_i)$  takes the form

$$\omega_t(X_i) = \frac{\text{Cov}[Y_i(t), Y_i^\dagger(t)|X_i]}{\mathbb{V}[Y_i^\dagger(t)|X_i]} = \frac{\text{Cov}[Y_i, Y_i^\dagger(t)|T_i = t, X_i]}{\mathbb{V}[Y_i^\dagger(t)|X_i]},$$

assigning higher weights when the LLM predictions are more strongly correlated with the true potential outcomes. Together with this weighting scheme,  $(Y_i^\dagger(t) - \mu_t^\dagger(X_i))\omega_t(X_i)$  acts as a linear predictor of  $Y_i(t) - \mu_t(X_i)$ , although this is done nonparametrically by conditioning on the structured covariates. This allows the predictive precision of the language model to vary across units, a feature likely to arise in practice. We remark that this heterogeneous weighting is crucial: without it, incorporating a mean-zero term into the estimating equation can lead to efficiency loss when  $Y_i^\dagger(t)$  provides little information about  $Y_i(t)$  beyond what is already captured by  $X_i$ .

Another distinguishing feature of CALM is its principled incorporation of few-shot LLM predictions through demonstrative example resampling and aggregation. To compare, a naive approach would generate  $Y_i^\dagger(t)$  using an LLM with  $X_i, Z_i$  and a fixed set of demonstrative examples as input. This, however, not only turns out to be suboptimal from a methodological perspective but also leads to nontrivial issues for our theoretical analysis. Specifically, few-shot predictions are known to be sensitive to the composition and ordering of the demonstrative examples in the prompt. Moreover, as those examples are drawn from the same sample, they induce correlations among the predicted outcomes that are difficult to characterize. The former leads to imprecise predictions, undermining the quality and overall performance of CALM; the latter makes asymptotic analysis very challenging. The novel demonstrative examples resampling step we introduce successfully tackles these two

issues. See Step 2 below and the theoretical results in Section 4.

Finally,  $1 - \mathbb{1}(T_i = t)/e_t(X_i)$  is introduced so that the estimating equation is Neyman orthogonal to the nuisance functions, a property that ensures valid inference even when those functions are estimated from data. The current form of  $\bar{\mu}_{t,\text{CALM}}$  is nonetheless infeasible, since it depends on the unknown nuisance functions  $\mu_t(\cdot)$ ,  $\mu_t^\dagger(\cdot)$ , and  $\omega_t(\cdot)$ . As we establish in Section 4, Neyman orthogonality combined with sample splitting permits a wide range of first-step estimators for these nuisance functions under very mild regularity conditions, including standard kernel- or sieve-based nonparametric methods and more complex machine learning tools (see Steps 1, 3 and 4).

Before providing the detailed implementation, we briefly discuss the efficiency gains of CALM. Under mild regularity conditions, one can show that the formulas we provide,  $\bar{\mu}_{t,\text{AIPW}}$  and  $\bar{\mu}_{t,\text{CALM}}$ , correspond to the asymptotically linear representations of the AIPW approach and our CALM, respectively. Thus, their variances allow a straightforward comparison of statistical efficiency:

$$\begin{aligned} \mathbf{V}_{t,\text{AIPW}} &= n \cdot \mathbb{V}[\bar{\mu}_{t,\text{AIPW}}] = \mathbb{E}\left[\frac{\mathbb{V}[Y_i(t)|X_i]}{e_t(X_i)} + (\mu_t(X_i) - \mu_t)^2\right], \\ \text{while } \mathbf{V}_{t,\text{CALM}} &= n \cdot \mathbb{V}[\bar{\mu}_{t,\text{CALM}}] = \mathbf{V}_{t,\text{AIPW}} - \mathbb{E}\left[\frac{\mathbb{V}[Y_i(t)|X_i]}{e_t(X_i)}(1 - e_t(X_i))\rho_t^2(X_i)\right], \end{aligned} \quad (2)$$

and  $\rho_t(X_i) = \text{Corr}[Y_i(t), Y_i^\dagger(t)|X_i]$ . Thus, CALM delivers nontrivial efficiency gains over AIPW except in the very rare case in which the LLM-based predictions turn out to be random noise for all  $X_i$ .

## 2.2 CALM with Few-Shot Learning: Implementation Details

We now describe the step-by-step procedure for estimating  $\mu_t = \mathbb{E}[Y_i(t)]$  using CALM with few-shot learning. Alongside the detailed steps, we also provide heuristics and guidance for practical use. The first step is a three-way sample split:

**Step 1 (Three-way sample splitting)** Randomly split the units  $i = 1, 2, \dots, n$  into three non-overlapping subsamples  $\mathcal{I}_1$ ,  $\mathcal{I}_2$ , and  $\mathcal{I}_3$ .

For now, one may think of the first fold  $\mathcal{I}_1$  as a reservoir of demonstrative examples for few-shot learning, the second,  $\mathcal{I}_2$ , as the fold used to estimate the nuisance functions  $\mu_t(\cdot)$ ,  $\mu_t^\dagger(\cdot)$ , and  $\omega_t(\cdot)$ , and the last  $\mathcal{I}_3$  for forming estimating equations. Later we will employ cross-fitting (Step 4), in

which the roles of the three folds are rotated.

The next step generates predictions of the potential outcomes via few-shot learning with demonstrative-example resampling.

**Step 2 (Few-shot predictions with demonstrative examples resampling)** Fix some (small)  $m$ . Let  $\mathcal{S}^* = \{(X_j^*, Z_j^*, T_j^*, Y_j^*), j = 1, \dots, m\}$  be a collection of ordered observations, randomly drawn without replacement from the arm- $t$  observations in  $\mathcal{I}_1$ . Let  $i$  denote a unit from  $\mathcal{I}_2$  or  $\mathcal{I}_3$ . Then instruct the LLM to predict the outcome for unit  $i$  as if enrolled in treatment  $t$ , given a context describing the RCT setting, the serialized covariates  $(X_i, Z_i)$ , and the serialized demonstrative examples  $\mathcal{S}^*$ . By repeating the above procedure  $B$  times with independently resampled examples  $\mathcal{S}_b^*$  for  $b = 1, 2, \dots, B$ , we obtain the aggregated prediction:

$$X_i, Z_i, \mathcal{S}^* \xrightarrow[\text{LLM few-shot}]{\text{Serialization}} Y_i^\dagger(t; \mathcal{S}^*) \xrightarrow[\text{Aggregation}]{\text{Resampling } \mathcal{S}^*} Y_i^\dagger(t; \mathcal{I}_1) = \frac{1}{B} \sum_{b=1}^B Y_i^\dagger(t; \mathcal{S}_b^*).$$

The above step involves constructing a prompt with four components to enable few-shot prediction from a pre-trained LLM. To start, we provide an instruction describing the task: predicting the potential outcome for unit  $i$  as if they were assigned to treatment  $t$ . To ensure the language model correctly interprets the data, the instruction also includes natural language descriptions of the input variables and additional contextual knowledge, such as a description of the treatment arms. Next, we provide the demonstrative examples  $\mathcal{S}^*$ , where each example contains the unit’s covariates, treatment assignment, and their observed outcome. Third, we provide the input data  $X_i$  and  $Z_i$  for unit  $i$ , where structured covariates are serialized into natural language and unstructured textual data (e.g., free-text responses) are presented directly. Other modalities, such as images, can be uploaded separately if supported by the model interface. Finally, we specify the desired output by requesting the counterfactual outcome prediction in a fixed format. A concrete example of our few-shot prompt design is provided in the Supplementary Materials.

Step 2 simultaneously tackles the two statistical challenges of integrating few-shot predictions into CALM discussed earlier. For the first challenge, it repeatedly resamples small, random sets of demonstrative examples and aggregates the resulting few-shot predictions. This averaging procedure effectively neutralizes the variability introduced by the random composition and ordering of demonstrative examples, which would otherwise persist even as the sample size grows. This step

also addresses the second challenge: a naive use of demonstrative examples can induce correlations among the predicted outcomes. More precisely, a naive implementation would stop at  $Y_i^\dagger(t; \mathcal{S}^*)$  without further resampling or aggregation. This, however, leads to correlated predictions for units  $i$  and  $i'$  whenever their demonstrative examples overlap. Such correlations are, unfortunately, very difficult to characterize. By contrast, the novel resampling and aggregation we introduce give the final prediction  $Y_i^\dagger(t; \mathcal{I}_1)$  a U-statistic structure. Together with sample splitting, this ensures that the predictions for different units  $i$  and  $i'$  are asymptotically independent, a key property we exploit in our theoretical analysis.

Now that predicted potential outcomes have been generated for all units in  $\mathcal{I}_2$  and  $\mathcal{I}_3$ , the next step involves estimating the nuisance functions using subsample  $\mathcal{I}_2$  and constructing “estimating equations” for units in  $\mathcal{I}_3$ .

**Step 3 (Nuisance function and parameter estimation)** Estimate  $\mu_t(\cdot)$ ,  $\mu_t^\dagger(\cdot)$ , and  $\omega_t(\cdot)$  using subsample  $\mathcal{I}_2$ . This can be done via sieve-based or kernel-based nonparametric procedures, or other machine learning methods. Denote the resulting estimates as  $\hat{\mu}_t(\cdot; \mathcal{I}_2)$ ,  $\hat{\mu}_t^\dagger(\cdot; \mathcal{I}_2)$ , and  $\hat{\omega}_t(\cdot; \mathcal{I}_2)$ , respectively. Form the estimating equation for observations  $i$  in  $\mathcal{I}_3$ :

$$\begin{aligned} \hat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2) = & \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} \left( Y_i - \hat{\mu}_t(X_i; \mathcal{I}_2) \right) + \hat{\mu}_t(X_i; \mathcal{I}_2) \\ & + \left( 1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} \right) \left( Y_i^\dagger(t; \mathcal{I}_1) - \hat{\mu}_t^\dagger(X_i; \mathcal{I}_2) \right) \hat{\omega}_t(X_i; \mathcal{I}_2). \end{aligned}$$

As the notation suggests,  $\hat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2)$  depends on all three data folds:  $\mathcal{I}_1$  serves as a pool of demonstrative examples,  $\mathcal{I}_2$  provides the nuisance function estimates, and  $\mathcal{I}_3$  provides the unit’s own observations. Without loss of generality, we will assume that all calibration weights we construct are bounded.

The final step rotates the roles of the three data folds, a technique known as cross-fitting. We also provide a valid asymptotic variance estimator and the corresponding confidence interval construction.

**Step 4 (CALM-based estimation and statistical inference)** Rotate the roles of  $\mathcal{I}_1$ ,  $\mathcal{I}_2$ , and

$\mathcal{I}_3$  by repeating Steps 2-3, and form the final CALM estimator of  $\mu_t$ :

$$\hat{\mu}_{t,\text{CALM}} = \frac{1}{n} \left\{ \sum_{i \in \mathcal{I}_3} \hat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2) + \sum_{i \in \mathcal{I}_1} \hat{\varphi}_t(i; \mathcal{I}_2, \mathcal{I}_3) + \sum_{i \in \mathcal{I}_2} \hat{\varphi}_t(i; \mathcal{I}_3, \mathcal{I}_1) \right\}.$$

Construct the asymptotic variance estimator

$$\hat{\mathbf{V}}_{t,\text{CALM}} = \frac{1}{n} \left\{ \sum_{i \in \mathcal{I}_3} \left( \hat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2) - \hat{\mu}_{t,\text{CALM}} \right)^2 + \sum_{i \in \mathcal{I}_1} \left( \hat{\varphi}_t(i; \mathcal{I}_2, \mathcal{I}_3) - \hat{\mu}_{t,\text{CALM}} \right)^2 + \sum_{i \in \mathcal{I}_2} \left( \hat{\varphi}_t(i; \mathcal{I}_3, \mathcal{I}_1) - \hat{\mu}_{t,\text{CALM}} \right)^2 \right\},$$

leading to the  $1 - \alpha$  confidence interval  $[\hat{\mu}_{t,\text{CALM}} \pm z_{\alpha/2} \hat{\mathbf{V}}_{t,\text{CALM}}^{1/2} / \sqrt{n}]$  where  $z_{\alpha/2}$  denotes the upper  $\alpha/2$ -quantile of the standard normal distribution.

### 2.3 Misspecified Outcome Models and Constrained Calibration

Step 3 allows for the estimation of nuisance functions using general nonparametric methods. Empirical practice, however, may favor parsimonious outcome models, such as an additively separable functional form or linear specifications based on finite-dimensional basis expansions. A prominent example of the latter is covariate discretization, also known as coarsening or binning. Two primary considerations drive these modeling choices. First, a parsimonious outcome model greatly simplifies estimation, producing results that are straightforward to compute, report, and interpret, which may be particularly relevant in data-scarce applications. Second, in the specific case of discretization, many randomized controlled trials employ stratified or blocked randomization, making the treatment assignment probability  $e_t(X_i)$  piecewise constant across a finite set of strata. Aligning covariate adjustment with the granularity of the randomization design ensures internal consistency. The tradeoff behind parsimonious outcome models is that they are likely misspecified.

Therefore, in this subsection, we discuss how outcome model misspecification affects our proposed CALM framework; that is, we consider a scenario where  $\hat{\mu}_t(\cdot; \mathcal{I}_2)$  and  $\hat{\mu}_t^\dagger(\cdot; \mathcal{I}_2)$  converge to  $\tilde{\mu}_t(\cdot)$  and  $\tilde{\mu}_t^\dagger(\cdot)$ , where these limits may differ from the true conditional expectations. In line with parsimonious modeling choices, we also allow the calibration weights to be estimated within a constrained class  $\mathcal{W}$ . Depending on the specific application and the analyst's modeling preferences, this class can range from fully nonparametric to, at the other extreme, parametric functions of the covariates depending on a finite-dimensional calibration parameter.

We first present a general result on the guaranteed efficiency gain relative to the AIPW approach, even if the outcome models are misspecified and calibration is constrained. To keep the exposition accessible, the discussion in this subsection is heuristic, and formal results are presented in Section 4. We again start with the asymptotically linear representation: under very mild regularity conditions, our CALM estimator is equivalent to

$$\bar{\mu}_{t,\text{CALM}} = \frac{1}{n} \sum_{i=1}^n \underbrace{\frac{\mathbb{1}(T_i = t)}{e_t(X_i)} (Y_i - \tilde{\mu}_t(X_i)) + \tilde{\mu}_t(X_i)}_{\text{AIPW with misspecification}} + \underbrace{\left(1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)}\right) (Y_i^\dagger(t) - \tilde{\mu}_t^\dagger(X_i)) \omega_t(X_i)}_{\omega_t(\cdot) \in \mathcal{W}: \text{constrained calibration}}, \quad (3)$$

for some calibration weight that may not be the optimal one. Asymptotic variances of the CALM and AIPW approaches can be found from the above representation. Let  $\lambda_t(X_i) = (1 - e_t(X_i))/e_t(X_i)$ . Then

$$\begin{aligned} \mathbf{V}_{t,\text{AIPW}} &= \mathbb{E} \left[ \frac{\mathbb{V}[Y_i(t)|X_i]}{e_t(X_i)} + (\mu_t(X_i) - \mu_t)^2 + \lambda_t(X_i) (\tilde{\mu}_t(X_i) - \mu_t(X_i))^2 \right], \\ \mathbf{V}_{t,\text{CALM}} &= \mathbf{V}_{t,\text{AIPW}} + \mathbb{E} \left[ \omega_t^2(X_i) \lambda_t(X_i) \mathbb{E}[(Y_i^\dagger(t) - \tilde{\mu}_t^\dagger(X_i))^2 | X_i] \right. \\ &\quad \left. - 2\omega_t(X_i) \lambda_t(X_i) \mathbb{E}[(Y_i(t) - \tilde{\mu}_t(X_i))(Y_i^\dagger(t) - \tilde{\mu}_t^\dagger(X_i)) | X_i] \right]. \quad (4) \end{aligned}$$

If  $\omega_t(\cdot)$  is chosen to minimize the last term over  $\mathcal{W}$ , then, because the AIPW approach corresponds to  $\omega_t(\cdot) = 0$ , the optimally calibrated CALM estimator is weakly more efficient than AIPW whenever  $0 \in \mathcal{W}$ . This is an extremely mild requirement, and is satisfied, for example, when  $\mathcal{W}$  consists of linear models of the covariates, or basis transformations thereof.

We now discuss a few implementation details and some associated technical considerations. First, a potential change is to employ only two data folds, where  $\mathcal{I}_1$  continues to serve as the reservoir for demonstrative examples (i.e., Step 2 remains the same), and  $\mathcal{I}_2$  is used for nuisance function estimation, calibration weights construction, and forming an estimator for  $\mu_t$ . In this way, cross-fitting only needs to be done twice by flipping the roles of  $\mathcal{I}_1$  and  $\mathcal{I}_2$ , thus significantly reducing computation. Formally,

**Step 4 (reduced cross-fitting)** Flip the roles of  $\mathcal{I}_1$  and  $\mathcal{I}_2$  by repeating Steps 2–3, and form the

CALM estimator of  $\mu_t$ :

$$\hat{\mu}_{t,\text{CALM}} = \frac{1}{n} \left\{ \sum_{i \in \mathcal{I}_2} \hat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2) + \sum_{i \in \mathcal{I}_1} \hat{\varphi}_t(i; \mathcal{I}_2, \mathcal{I}_1) \right\}.$$

Construct the asymptotic variance estimator

$$\hat{V}_{t,\text{CALM}} = \frac{1}{n} \left\{ \sum_{i \in \mathcal{I}_2} \left( \hat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2) - \hat{\mu}_{t,\text{CALM}} \right)^2 + \sum_{i \in \mathcal{I}_1} \left( \hat{\varphi}_t(i; \mathcal{I}_2, \mathcal{I}_1) - \hat{\mu}_{t,\text{CALM}} \right)^2 \right\},$$

leading to the  $1 - \alpha$  confidence interval  $[\hat{\mu}_{t,\text{CALM}} \pm z_{\alpha/2} \hat{V}_{t,\text{CALM}}^{1/2} / \sqrt{n}]$  where  $z_{\alpha/2}$  denotes the upper  $\alpha/2$ -quantile of the standard normal distribution.

Removing one data fold, however, comes at a cost: its validity typically requires that nuisance function estimation is conducted in function classes that are simple or parsimonious (hence Donsker-type), precisely the main motivating scenario of this subsection. We adopt this alternative asymptotic reasoning in Section 4 when providing sufficient conditions for CALM under misspecification with constrained calibration. When such a restriction on the nuisance function classes is not plausible, the three-way sample split of Section 2.2 remains the more robust choice: our CALM estimator then admits an asymptotically linear representation (and hence is asymptotically normally distributed) under very mild conditions on the nuisance function estimates.

We now consider the issue of estimating the calibration weight. The appropriate approach depends on the class  $\mathcal{W}$ . In the case of unconstrained  $\mathcal{W}$  consisting of all measurable functions, the optimal calibration weight is the ratio of  $\mathbb{E}[(Y_i(t) - \tilde{\mu}_t(X_i))(Y_i^\dagger(t) - \tilde{\mu}_t^\dagger(X_i)) | X_i]$  and  $\mathbb{E}[(Y_i^\dagger(t) - \tilde{\mu}_t^\dagger(X_i))^2 | X_i]$ . On the other hand, one may specify a flexible, parametric function class for the calibration weights through a parameter vector, which may exhibit more stable finite-sample performance in data-scarce environments; that is,  $\omega_t(\cdot, \beta)$ . A convenient estimator for  $\beta$  can be found by optimizing the estimated asymptotic variance: using the same notation developed above and in Section 2.2, define

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \frac{1}{|\mathcal{I}_2|} \sum_{i \in \mathcal{I}_2} \left( \hat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2; \beta) - \bar{\varphi}_t(\mathcal{I}_1, \mathcal{I}_2; \beta) \right)^2,$$

where, with a slight abuse of notation,

$$\begin{aligned}\widehat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2; \boldsymbol{\beta}) &= \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} \left( Y_i - \widehat{\mu}_t(X_i; \mathcal{I}_2) \right) + \widehat{\mu}_t(X_i; \mathcal{I}_2) \\ &\quad + \left( 1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} \right) \left( Y_i^\dagger(t; \mathcal{I}_1) - \widehat{\mu}_t^\dagger(X_i; \mathcal{I}_2) \right) \omega_t(X_i; \boldsymbol{\beta}),\end{aligned}$$

and  $\overline{\varphi}_t(\mathcal{I}_1, \mathcal{I}_2; \boldsymbol{\beta})$  denotes the sample average of  $\widehat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2; \boldsymbol{\beta})$  in data fold  $\mathcal{I}_2$ .

In the remainder of this section, we focus on the special case of covariate discretization (a.k.a. coarsening or binning). This approach is particularly attractive due to its simplicity: nuisance functions are estimated as block averages. It also leads to an estimator that is more aligned with the sampling scheme when the underlying trial is conducted in a stratified manner. Building on this insight, we assume there is a pre-specified discretization that partitions the covariate space into  $J$  blocks  $\mathcal{X}_1, \dots, \mathcal{X}_J$ , and that the propensity score is piecewise constant:  $e_t(X_i) = \sum_{j=1}^J \mathbb{1}(X_i \in \mathcal{X}_j) e_{t,j}$ . Then the two conditional expectations can be estimated via simple block averages

$$\begin{aligned}\widehat{\mu}_t(X_i; \mathcal{I}_2) &= \sum_{j=1}^J \mathbb{1}(X_i \in \mathcal{X}_j) \widehat{\mu}_{t,j}(\mathcal{I}_2), & \widehat{\mu}_{t,j}(\mathcal{I}_2) &= \frac{1}{n_{t,j}(\mathcal{I}_2)} \sum_{i \in \mathcal{I}_2} \mathbb{1}(T_i = t, X_i \in \mathcal{X}_j) Y_i, \\ \widehat{\mu}_t^\dagger(X_i; \mathcal{I}_2) &= \sum_{j=1}^J \mathbb{1}(X_i \in \mathcal{X}_j) \widehat{\mu}_{t,j}^\dagger(\mathcal{I}_2), & \widehat{\mu}_{t,j}^\dagger(\mathcal{I}_2) &= \frac{1}{n_j(\mathcal{I}_2)} \sum_{i \in \mathcal{I}_2} \mathbb{1}(X_i \in \mathcal{X}_j) Y_i^\dagger(t; \mathcal{I}_1).\end{aligned}$$

In the above,  $n_{t,j}(\mathcal{I}_2) = \sum_{i \in \mathcal{I}_2} \mathbb{1}(T_i = t, X_i \in \mathcal{X}_j)$  counts the units in data fold  $\mathcal{I}_2$  that belong to block  $j$  and are assigned to treatment  $t$ . Analogously,  $n_j(\mathcal{I}_2) = \sum_{i \in \mathcal{I}_2} \mathbb{1}(X_i \in \mathcal{X}_j)$  is the number of units in  $\mathcal{I}_2$  that fall into block  $j$ .

Due to discretization, large-sample limits of the estimated nuisance functions are also of piecewise-constant form. Building on the general result established earlier for CALM with misspecified outcomes, it is routine to show that the optimal calibration also takes a piecewise constant structure, and can be estimated via the simple formula:

$$\begin{aligned}\widehat{\omega}_t(X_i; \mathcal{I}_2) &= \sum_{j=1}^J \mathbb{1}(X_i \in \mathcal{X}_j) \widehat{\omega}_{t,j}(\mathcal{I}_2), \\ \text{with } \widehat{\omega}_{t,j}(\mathcal{I}_2) &= \frac{n_{t,j}^{-1}(\mathcal{I}_2) \sum_{i \in \mathcal{I}_2} \mathbb{1}(T_i = t, X_i \in \mathcal{X}_j) (Y_i - \widehat{\mu}_{t,j}(\mathcal{I}_2)) (Y_i^\dagger(t; \mathcal{I}_1) - \widehat{\mu}_{t,j}^\dagger(\mathcal{I}_2))}{n_j^{-1}(\mathcal{I}_2) \sum_{i \in \mathcal{I}_2} \mathbb{1}(X_i \in \mathcal{X}_j) (Y_i^\dagger(t; \mathcal{I}_1) - \widehat{\mu}_{t,j}^\dagger(\mathcal{I}_2))^2}.\end{aligned}$$

Notably, the weight  $\lambda_t(X_i)$  from the general result above does not appear in  $\hat{\omega}_{t,j}(\mathcal{I}_2)$ : since the propensity score is piecewise constant,  $\lambda_t(X_i) = \lambda_{t,j} = (1 - e_{t,j})/e_{t,j}$  for all  $i$  with  $X_i \in \mathcal{X}_j$ .

As a final remark, the granularity of  $\mathcal{X}_1, \dots, \mathcal{X}_J$  trades off robustness against the size of the efficiency gain: coarser discretizations average over larger cells, which may stabilize the estimates in finite samples but can dilute genuine heterogeneity in the true calibration weight, weakening the correlation between the true and LLM-generated potential outcomes. On the other hand, finer discretizations do the opposite, at the cost of noisier within-cell estimates when cells contain few observations.

### 3 Extensions

We now discuss special cases and extensions of the CALM framework introduced in Section 2. These include counterfactual predictions from zero-shot learning and the estimation of both unconditional and conditional average treatment effects.

#### 3.1 CALM with Zero-Shot Predictions

A special case of CALM is  $m = 0$ , where the LLM-generated counterfactual predictions are produced without any demonstrative examples. This is commonly referred to as zero-shot learning [5]. Throughout, we use the subscript **ZS** to denote objects specific to this zero-shot special case, in parallel to the subscript-free notation used for the few-shot CALM estimator of Section 2.2. The zero-shot prediction is obtained via

$$X_i, Z_i \xrightarrow[\text{LLM zero-shot}]{\text{Serialization}} Y_{i,\text{ZS}}^\dagger(t), \quad \text{for } 1 \leq i \leq n,$$

using a prompt with three components: an instruction describing the counterfactual prediction task and the treatment arms, the serialized input  $(X_i, Z_i)$ , and the desired output format. No demonstrative examples  $\mathcal{S}^*$  are drawn, and consequently no resampling or aggregation step is needed: a single LLM query per unit and treatment arm suffices to obtain  $Y_{i,\text{ZS}}^\dagger(t)$ .

Because zero-shot predictions no longer depend on a shared pool of demonstrative examples, the reservoir fold  $\mathcal{I}_1$  in Step 1 becomes unnecessary. The sample splitting therefore simplifies to a

two-way split into folds  $\mathcal{I}_1$  and  $\mathcal{I}_2$ , and Steps 3–4 carry over with  $Y_i^\dagger(t; \mathcal{I}_1)$  replaced by  $Y_{i,\text{zs}}^\dagger(t)$  and the corresponding nuisance functions

$$\mu_{t,\text{zs}}^\dagger(X_i) = \mathbb{E}[Y_{i,\text{zs}}^\dagger(t)|X_i], \quad \omega_{t,\text{zs}}(X_i) = \frac{\text{Cov}[Y_i(t), Y_{i,\text{zs}}^\dagger(t)|X_i]}{\mathbb{V}[Y_{i,\text{zs}}^\dagger(t)|X_i]}.$$

Estimating  $\mu_t(\cdot)$ ,  $\mu_{t,\text{zs}}^\dagger(\cdot)$ , and  $\omega_{t,\text{zs}}(\cdot)$  on one fold and forming the estimating equation on the other, then rotating the two folds, yields the zero-shot CALM estimator

$$\hat{\mu}_{t,\text{zs}} = \frac{1}{n} \left\{ \sum_{i \in \mathcal{I}_2} \hat{\varphi}_{t,\text{zs}}(i; \mathcal{I}_1) + \sum_{i \in \mathcal{I}_1} \hat{\varphi}_{t,\text{zs}}(i; \mathcal{I}_2) \right\},$$

with  $\hat{\varphi}_{t,\text{zs}}(i; \mathcal{I}_1)$  defined exactly as  $\hat{\varphi}_t(i; \mathcal{I}_1, \mathcal{I}_2)$  in Step 3, but with  $Y_{i,\text{zs}}^\dagger(t)$ ,  $\hat{\mu}_t(X_i; \mathcal{I}_1)$ ,  $\hat{\mu}_{t,\text{zs}}^\dagger(X_i; \mathcal{I}_1)$ , and  $\hat{\omega}_{t,\text{zs}}(X_i; \mathcal{I}_1)$  in place of  $Y_i^\dagger(t; \mathcal{I}_1)$ ,  $\hat{\mu}_t(X_i; \mathcal{I}_2)$ ,  $\hat{\mu}_t^\dagger(X_i; \mathcal{I}_2)$ , and  $\hat{\omega}_t(X_i; \mathcal{I}_2)$ . Asymptotic variance estimation and confidence interval construction proceed analogously.

The appeal of the zero-shot special case is largely computational. Because no demonstrative examples are resampled, obtaining  $\{Y_{i,\text{zs}}^\dagger(t)\}_{i=1}^n$  requires only  $n$  LLM queries, as opposed to the  $nB$  queries needed to aggregate few-shot predictions over  $B$  resamples in Step 2. This reduction in queries translates directly into lower latency and monetary cost, both of which scale with the number of API calls to the LLM in practice. Dropping the demonstrative-example block from the prompt also shortens each query, easing the token-budget constraints imposed by the LLM’s context window. Sample splitting is likewise simpler: with no fold reserved purely as a donor pool of examples, the two remaining folds are larger than any two folds under the three-way split, and the resulting estimating equations no longer require the U-statistic argument used in Section 4 to establish asymptotic independence of the predictions across units, as  $Y_{i,\text{zs}}^\dagger(t)$  depends only on unit  $i$ ’s own covariates.

This computational simplicity, however, typically comes at the cost of a smaller efficiency gain over AIPW. Recall from Section 2.1 that the variance reduction delivered by CALM is governed by the strength of the conditional correlation between the true and predicted potential outcomes,  $\rho_t(X_i) = \text{Corr}[Y_i(t), Y_i^\dagger(t)|X_i]$ : the corresponding zero-shot analogue  $\rho_{t,\text{zs}}(X_i) =$

$\text{Corr}[Y_i(t), Y_{i,\text{ZS}}^\dagger(t)|X_i]$  enters the variance formula in exactly the same way,

$$V_{t,\text{ZS}} = V_{t,\text{AIPW}} - \mathbb{E}\left[\frac{\mathbb{V}[Y_i(t)|X_i]}{e_t(X_i)}(1 - e_t(X_i))\rho_{t,\text{ZS}}^2(X_i)\right].$$

Because a zero-shot prediction is generated from the LLM’s pretrained knowledge alone, without ever observing a single realized  $(X, Z, Y)$  triple from the trial, it has no opportunity to adapt to the specific empirical relationship between covariates and outcomes in the study population: idiosyncrasies of the outcome scale, site-specific practices, and subtle features of the intervention protocol are all invisible to a zero-shot prompt. Few-shot examples, by contrast, implicitly reveal this relationship to the LLM, which can be viewed as a form of in-context calibration to the trial at hand. We therefore expect  $\rho_{t,\text{ZS}}(x)$  to be weaker than its few-shot counterpart in most applications, so that zero-shot CALM delivers a smaller (though still nontrivial) variance reduction relative to AIPW. Crucially, this loss is one of efficiency, not validity: because the correction term remains residualized and optimally weighted,  $\hat{\mu}_{t,\text{ZS}}$  is consistent and no less efficient than AIPW regardless of how weak  $\rho_{t,\text{ZS}}(x)$  turns out to be. Zero-shot CALM is thus a computationally attractive default that is always safe to use, while the few-shot approach of Section 2.2 is preferable whenever the added query cost is acceptable and reliable in-sample examples are available.

### 3.2 Average Treatment Effect Estimation

Fixing two treatments  $t$  and  $t'$ , we now turn to the average treatment effect (ATE)  $\tau_{t,t'} = \mu_t - \mu_{t'} = \mathbb{E}[Y_i(t) - Y_i(t')]$ . A natural candidate combines the two CALM mean estimators of Section 2.2, applied separately to arms  $t$  and  $t'$ :  $\hat{\mu}_{t,\text{CALM}} - \hat{\mu}_{t',\text{CALM}}$ . This difference inherits consistency, asymptotic normality, and the arm-by-arm efficiency gain over AIPW established in Section 2. The only ingredient that is not already efficient for the difference is the calibration weight:  $\omega_t(X_i)$  and  $\omega_{t'}(X_i)$  are each optimal for predicting their own arm’s missing outcome, but treating them separately ignores that  $Y_i^\dagger(t)$  and  $Y_i^\dagger(t')$  are themselves correlated given  $X_i$ , as both counterfactual predictions are generated by employing the same unit-level pre-intervention information  $(X_i, Z_i)$ . In addition, it is possible that the counterfactual prediction for arm  $t$  is predictive for the  $t'$ -potential outcome. We now derive the calibration weight that is optimal for the difference.

Substituting the asymptotic linear representation (1) for each arm, but leaving the calibration

weights generic rather than fixing them at their Section 2.1 values, gives

$$\begin{aligned} \bar{\tau}_{t,t',\text{CALM}} = & \frac{1}{n} \sum_{i=1}^n \left[ \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} (Y_i - \mu_t(X_i)) + \mu_t(X_i) \right] - \left[ \frac{\mathbb{1}(T_i = t')}{e_{t'}(X_i)} (Y_i - \mu_{t'}(X_i)) + \mu_{t'}(X_i) \right] \\ & + \left( 1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} \right) (Y_i^\dagger(t) - \mu_t^\dagger(X_i)) \omega_t(X_i) - \left( 1 - \frac{\mathbb{1}(T_i = t')}{e_{t'}(X_i)} \right) (Y_i^\dagger(t') - \mu_{t'}^\dagger(X_i)) \omega_{t'}(X_i). \end{aligned} \quad (5)$$

Importantly, the variance of  $\bar{\tau}_{t,t',\text{CALM}}$  provides insights on how the asymptotic variance of the ATE estimator depends on the calibration weights. To be specific, define

$$V_{t,t',\text{CALM}} = n \cdot \mathbb{V}[\bar{\tau}_{t,t',\text{CALM}}] = n \cdot \mathbb{V}[\bar{\tau}_{t,t',\text{AIPW}}] + \mathbb{E} \left[ \boldsymbol{\omega}(X_i)^\top \boldsymbol{\Sigma}^\dagger(X_i) \boldsymbol{\omega}(X_i) - 2 \boldsymbol{\omega}(X_i)^\top \boldsymbol{\sigma}(X_i) \right], \quad (6)$$

where  $\boldsymbol{\omega}(X_i) = (\omega_t(X_i), \omega_{t'}(X_i))$ , and  $\boldsymbol{\Sigma}^\dagger(X_i)$  and  $\boldsymbol{\sigma}(X_i)$  are weighted conditional covariance matrices:

$$\begin{aligned} \boldsymbol{\Sigma}^\dagger(X_i) &= \begin{bmatrix} \lambda_t(X_i) \mathbb{V}[Y_i^\dagger(t)|X_i] & \text{Cov}[Y_i^\dagger(t), Y_i^\dagger(t')|X_i] \\ \text{Cov}[Y_i^\dagger(t), Y_i^\dagger(t')|X_i] & \lambda_{t'}(X_i) \mathbb{V}[Y_i^\dagger(t')|X_i] \end{bmatrix} \\ \boldsymbol{\sigma}(X_i) &= \begin{bmatrix} \lambda_t(X_i) \text{Cov}[Y_i(t), Y_i^\dagger(t)|X_i] + \text{Cov}[Y_i(t'), Y_i^\dagger(t)|X_i] \\ \text{Cov}[Y_i(t), Y_i^\dagger(t')|X_i] + \lambda_{t'}(X_i) \text{Cov}[Y_i(t'), Y_i^\dagger(t')|X_i] \end{bmatrix}, \end{aligned}$$

with  $\lambda_t(X_i) = (1 - e_t(X_i))/e_t(X_i)$ . As such, the optimal calibration weights are

$$\boldsymbol{\omega}_{\text{ATE}}(X_i) = \left( \omega_{t,\text{ATE}}(X_i), \omega_{t',\text{ATE}}(X_i) \right) = \boldsymbol{\Sigma}^\dagger(X_i)^{-1} \boldsymbol{\sigma}(X_i). \quad (7)$$

The weights above clearly reduce to the arm-specific weights  $(\omega_t(X_i), \omega_{t'}(X_i))$  of Section 2 whenever the two predictions are conditionally uncorrelated, and each prediction is informative only about its own arm's true outcome. In that case  $\boldsymbol{\Sigma}^\dagger(X_i)$  is diagonal and each component of  $\boldsymbol{\sigma}(X_i)$  retains only its own-arm term, so  $\boldsymbol{\omega}_{\text{ATE}}(X_i)$  solves two decoupled univariate problems: joint calibration is a strict generalization of, rather than a departure from, the mean-estimation case.

Estimation replaces the population conditional moments in (7) by their nonparametric estimates, exactly as in Step 3, using  $Y_i^\dagger(t; \mathcal{I}_1)$  and  $Y_i^\dagger(t'; \mathcal{I}_1)$  in place of  $Y_i^\dagger(t)$  and  $Y_i^\dagger(t')$ ; call the resulting estimates  $\hat{\omega}_{t,\text{ATE}}(\cdot)$  and  $\hat{\omega}_{t',\text{ATE}}(\cdot)$ . Substituting them, together with  $\hat{\mu}_t(\cdot)$ ,  $\hat{\mu}_{t'}(\cdot)$ ,  $\hat{\mu}_t^\dagger(\cdot)$ ,  $\hat{\mu}_{t'}^\dagger(\cdot)$ , into the estimating equation of Step 3, with the arm-specific calibration weight there re-

placed by its ATE-specific counterpart, defines  $\hat{\varphi}_{t,\text{ATE}}(i; \mathcal{I}_1, \mathcal{I}_2)$  and  $\hat{\varphi}_{t',\text{ATE}}(i; \mathcal{I}_1, \mathcal{I}_2)$ ; the notation again emphasizes that  $\mathcal{I}_1$  provides demonstrative examples for counterfactual prediction using few-shot learning, while the nuisance functions are estimated with units from data fold  $\mathcal{I}_2$ . The final ATE estimator,  $\hat{\tau}_{t,t',\text{CALM}}$ , together with standard error and confidence intervals, can be constructed employing the same cross-fitting technique detailed in Step 4.

Before closing this subsection, we remark on the special case of discretized covariates discussed in Section 2.3, a popular choice both for experimental design and for post hoc analysis. As argued there, discretizing  $X_i$  into  $J$  blocks  $\mathcal{X}_1, \dots, \mathcal{X}_J$  lets the conditional expectations  $\hat{\mu}_t(\cdot)$ ,  $\hat{\mu}_{t'}(\cdot)$ ,  $\hat{\mu}_t^\dagger(\cdot)$ ,  $\hat{\mu}_{t'}^\dagger(\cdot)$  be estimated as simple block averages. Building on this, one can drop the data fold  $\mathcal{I}_3$  (or, more precisely, combine  $\mathcal{I}_2$  and  $\mathcal{I}_3$ ) and avoid the extra round of cross-fitting, since the relevant Donsker-type (stochastic equicontinuity) condition is easily verified in this setting; see Step 4 (reduced cross-fitting), Assumption 5, and Corollary 1 for details. Under covariate discretization based on a stratified randomization, the optimal calibration weights inherit the same simplification: like the nuisance functions, they are piecewise constant, and within each block their estimation reduces to an elementary two-variable regression rather than separate estimation of the underlying variances and covariances. To be specific, fix a block  $j$ . Using observations from this block in data fold  $\mathcal{I}_2$ ,  $\hat{\omega}_{t,j,\text{ATE}}(\mathcal{I}_2)$  and  $\hat{\omega}_{t',j,\text{ATE}}(\mathcal{I}_2)$  are simply the coefficients from

$$\begin{aligned} \text{regressing} \quad & - \frac{\mathbb{1}(T_i = t)}{e_{t,j}} (Y_i - \hat{\mu}_{t,j}(\mathcal{I}_2)) + \frac{\mathbb{1}(T_i = t')}{e_{t',j}} (Y_i - \hat{\mu}_{t',j}(\mathcal{I}_2)) \\ \text{on} \quad & \left(1 - \frac{\mathbb{1}(T_i = t)}{e_{t,j}}\right) (Y_i^\dagger(t; \mathcal{I}_1) - \hat{\mu}_{t,j}^\dagger(\mathcal{I}_2)) \quad \text{and} \quad - \left(1 - \frac{\mathbb{1}(T_i = t')}{e_{t',j}}\right) (Y_i^\dagger(t'; \mathcal{I}_1) - \hat{\mu}_{t',j}^\dagger(\mathcal{I}_2)). \end{aligned}$$

### 3.3 Conditional Average Treatment Effect

Building on the average treatment effect estimator of Section 3.2, we now turn to the conditional average treatment effect (CATE) estimand,  $\tau_{t,t'}(x) = \mathbb{E}[Y_i(t) - Y_i(t') | X_i = x]$ . Recall  $\hat{\varphi}_{t,\text{ATE}}(i; \mathcal{I}_1, \mathcal{I}_2)$  and  $\hat{\varphi}_{t',\text{ATE}}(i; \mathcal{I}_1, \mathcal{I}_2)$  from the previous subsection, and define

$$\hat{\varphi}_{t,t'}(i; \mathcal{I}_1, \mathcal{I}_2) = \hat{\varphi}_{t,\text{ATE}}(i; \mathcal{I}_1, \mathcal{I}_2) - \hat{\varphi}_{t',\text{ATE}}(i; \mathcal{I}_1, \mathcal{I}_2),$$

as well as the kernel-based smoothing weights:

$$\kappa_i(x; \mathcal{I}_3) = \frac{k_h(X_i - x)}{\sum_{i \in \mathcal{I}_3} k_h(X_i - x)}, \quad k_h(X_i - x) = \frac{1}{h^p} k\left(\frac{X_i - x}{h}\right),$$

for some kernel function  $k$ , bandwidth sequence  $h$ , and covariate dimension  $p$ . A CALM-based CATE estimator can then be constructed as a cross-fitted local average:

$$\begin{aligned} \hat{\tau}_{t,t',\text{CALM}}(x) = \frac{1}{3} & \left\{ \sum_{i \in \mathcal{I}_3} \kappa_i(x; \mathcal{I}_3) \hat{\varphi}_{t,t'}(i; \mathcal{I}_1, \mathcal{I}_2) \right. \\ & \left. + \sum_{i \in \mathcal{I}_2} \kappa_i(x; \mathcal{I}_2) \hat{\varphi}_{t,t'}(i; \mathcal{I}_3, \mathcal{I}_1) + \sum_{i \in \mathcal{I}_1} \kappa_i(x; \mathcal{I}_1) \hat{\varphi}_{t,t'}(i; \mathcal{I}_2, \mathcal{I}_3) \right\}. \end{aligned}$$

This concrete structure allows us to explicitly compute the asymptotic variance of the CATE estimator; in particular, we show in Section 4 that CALM again delivers an efficiency gain over the AIPW approach, provided that the language-model-based predictions are correlated with the true potential outcomes.

To gain some intuition, again consider the asymptotic linear representation. Let  $\kappa_i(x) = k_h(X_i - x) / \sum_{i=1}^n k_h(X_i - x)$  denote the local weights constructed using the full sample; then

$$\begin{aligned} \bar{\tau}_{t,t',\text{CALM}}(x) = \sum_{i=1}^n \kappa_i(x) & \left\{ \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} (Y_i - \mu_t(X_i)) + \mu_t(X_i) - \frac{\mathbb{1}(T_i = t')}{e_{t'}(X_i)} (Y_i - \mu_{t'}(X_i)) - \mu_{t'}(X_i) \right. \\ & + \left( 1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)} \right) (Y_i^\dagger(t) - \mu_t^\dagger(X_i)) \omega_{t,\text{ATE}}(X_i) \\ & \left. - \left( 1 - \frac{\mathbb{1}(T_i = t')}{e_{t'}(X_i)} \right) (Y_i^\dagger(t') - \mu_{t'}^\dagger(X_i)) \omega_{t',\text{ATE}}(X_i) \right\}. \end{aligned} \quad (8)$$

Then its conditional variance serves as a natural benchmark for efficiency comparison: define

$$\begin{aligned} \mathbf{V}_{t,t',\text{CALM}}(x) &= nh^p \cdot \mathbb{V}[\bar{\tau}_{t,t',\text{CALM}}(x) | \mathbf{X}] \\ &= nh^p \cdot \mathbb{V}[\bar{\tau}_{t,t',\text{AIPW}}(x) | \mathbf{X}] + \frac{\kappa_2}{f(x)} \left( \boldsymbol{\omega}_{\text{ATE}}(x)^\top \boldsymbol{\Sigma}^\dagger(x) \boldsymbol{\omega}_{\text{ATE}}(x) - 2 \boldsymbol{\omega}_{\text{ATE}}(x)^\top \boldsymbol{\sigma}(x) \right) + o_p(1). \end{aligned} \quad (9)$$

In the above,  $\kappa_2 = \int k(u)^2 du$  depends only on the kernel function,  $f(x)$  is the density of  $X_i$  at the evaluation point  $x$ , and  $\mathbb{E}[\cdot | \mathbf{X}]$  represents expectation conditional on all structured covariates  $X_1, X_2, \dots, X_n$ . Importantly, because the oracle calibration weights we derived in Section 3.2

are already specific to each value of the structured covariates, they also minimize the asymptotic variance of the conditional average treatment effect estimator, meaning that

$$\frac{\kappa_2}{f(x)} \left( \boldsymbol{\omega}_{\text{ATE}}(x)^\top \boldsymbol{\Sigma}^\dagger(x) \boldsymbol{\omega}_{\text{ATE}}(x) - 2\boldsymbol{\omega}_{\text{ATE}}(x)^\top \boldsymbol{\sigma}(x) \right) \leq 0,$$

with equality holding only if  $\boldsymbol{\sigma}(x) = 0$ .

## 4 Theoretical Investigations

We now investigate the theoretical properties of the CALM framework introduced in Sections 2 and 3, which integrates few-shot LLM predictions into treatment effect estimation. We first formalize the core assumptions underpinning our approach. We then establish the asymptotic properties of the CALM estimator for various causal parameters, demonstrating its consistency, asymptotic normality, and the validity of the resulting statistical inference.

Our first assumption concerns unconfounded treatment assignment and strong overlap [29, 30], a condition inherently satisfied in randomized controlled trials.

**Assumption 1 (Treatment assignment)** *(i) Treatment assignment  $T \in \{1, \dots, k\}$  is independent of the potential outcomes and the unstructured data, conditional on the structured covariates:  $T \perp\!\!\!\perp \{Y(1), \dots, Y(k), Z\} \mid X$ . (ii) The propensity score  $e_t(x) = \mathbb{P}[T = t \mid X = x]$  is bounded away from 0 for all  $t \in \{1, \dots, k\}$ .*

Next, we assume that the data consist of independent and identically distributed (iid) observations.

**Assumption 2 (Random sampling)** *The observed data  $\{(Y_i, T_i, X_i, Z_i)\}_{i=1}^n$  are independent and identically distributed.*

Recall that a unique feature of our method is that it averages over LLM-based predictions generated using small and randomly sampled demonstrative examples. While this averaging helps reduce the randomness of the LLM-predicted potential outcomes and eliminates dependence on the ordering of the examples, it also complicates the theoretical analysis. To facilitate the discussion, we first introduce a notation that can be viewed as a large-sample analogue of the few-shot predicted

outcome  $Y_i^\dagger(t; \mathcal{I})$ . Let  $\mathcal{S}^* = (O_1^*, O_2^*, \dots, O_m^*)$  denote a random sample drawn from the arm  $T = t$ , independent of  $(X_i, Z_i)$ . In line with Step 2, let  $Y_i^\dagger(t; \mathcal{S}^*)$  denote the few-shot prediction. We then define

$$Y_i^\dagger(t) = \mathbb{E}[Y_i^\dagger(t; \mathcal{S}^*) \mid X_i, Z_i],$$

where the expectation is taken with respect to the random demonstrative sample  $\mathcal{S}^*$ . To be fully rigorous, a more precise notation would be  $Y_{i,m,\theta}^\dagger(t)$  to explicitly denote the dependence on both the number of demonstrative examples,  $m$ , and the parameters of the underlying language model,  $\theta$ . However, we suppress these subscripts for notational simplicity.

Our next assumption requires finite moments and nonvanishing variance for the outcome variables, along with bounded counterfactual predictions. In applications, boundedness is standard since prompts naturally constrain outcome ranges. From a technical standpoint, this condition helps establish the convergence of aggregated predictions, though it can be relaxed to finite higher-order moments.

**Assumption 3 (Finite moments)** *There exists some constant  $c > 0$  such that for all  $t = 1, 2, \dots, k$ ,  $|\mu_t(X_i)|, \mathbb{V}[Y_i(t)|X_i], |Y_i^\dagger(t; \mathcal{S}^*)| \leq c$ . In addition,  $\mathbb{V}[Y_i(t)|X_i], \mathbb{V}[Y_i^\dagger(t)|X_i] \geq c^{-1}$ .*

We now introduce a high-level assumption stating that the estimated nuisance functions,  $\hat{\mu}_t(\cdot)$ ,  $\hat{\mu}_t^\dagger(\cdot)$ , and  $\hat{\omega}_t(\cdot)$ , are consistent. Depending on the specific form of the estimators, such as  $k$ -nearest neighbors, kernel- or series-based methods, regression trees, or random forests, more primitive conditions are available in the literature. Such high-level consistency assumptions are commonly employed in the analysis of semiparametric estimators with Neyman-orthogonal estimating equations. Together with sample splitting in Step 1 and cross-fitting in Step 4, they help facilitate the establishment of consistency and asymptotic normality. Let  $\|\cdot\|_{L_2}$  denote the  $L_2$ -norm with respect to the distribution of  $X$ .

**Assumption 4 (Consistent nuisance function estimation)** *The cross-fitted function estimates obtained are consistent:  $\|\hat{\mu}_t(\cdot; \mathcal{I}) - \mu_t(\cdot)\|_{L_2}, \|\hat{\mu}_t^\dagger(\cdot; \mathcal{I}) - \mu_t^\dagger(\cdot)\|_{L_2}$  and  $\|\hat{\omega}_t(\cdot; \mathcal{I}) - \omega_t(\cdot)\|_{L_2} \xrightarrow{P} 0$  for each data fold  $\mathcal{I}$  and treatment arm  $t \in \{1, \dots, k\}$ .*

We are now ready to state the main theoretical result: the CALM estimator  $\hat{\mu}_{t,\text{CALM}}$ , which incorporates few-shot predicted potential outcomes using an LLM, is consistent, asymptotically

normally distributed, and achieves a smaller asymptotic variance than the standard AIPW estimator whenever the LLM-based predictions are informative about the true potential outcomes.

**Theorem 1 (Asymptotic properties of  $\hat{\mu}_{t,\text{CALM}}$ )** *Let Assumptions 1–4 hold, and  $B \rightarrow \infty$ . Then*

$$\sqrt{n}(\hat{\mu}_{t,\text{CALM}} - \bar{\mu}_{t,\text{CALM}}) \xrightarrow{p} 0, \quad \text{which implies} \quad \sqrt{n}(\hat{\mu}_{t,\text{CALM}} - \mu_t) \xrightarrow{d} \mathcal{N}(0, \mathbf{V}_{t,\text{CALM}}),$$

where  $\bar{\mu}_{t,\text{CALM}}$  and  $\mathbf{V}_{t,\text{CALM}}$  are defined in (1) and (2), respectively.

The above theorem establishes that the CALM estimator remains consistent and asymptotically normal when coupled with the resampling-based few-shot strategy introduced in Section 2. It is also clear that CALM yields strictly smaller asymptotic variance, and thus significant efficiency gains, whenever the few-shot LLM-based prediction is informative about the true potential outcome  $Y_i(t)$ , i.e., when  $\rho_t(x)$ , or equivalently  $\omega_t(x)$ , is nonzero. This is also borne out in our empirical analysis.

Next, we examine to what extent the conclusions of Theorem 1 continue to hold without Assumption 4, i.e., under outcome model misspecification and constrained calibration, a scenario we discuss in Section 2.3. We make the following “stochastic equicontinuity” assumption on the estimated nuisance functions and calibration weights, which helps establish an asymptotically linear representation and hence, the asymptotic normality of the CALM estimator. Such conditions allow removing one data fold, reducing the algorithm to a two-way sample split, thus significantly reducing computation. They are also relatively straightforward to verify whenever nuisance function estimation is conducted in classes that are simple or parsimonious (i.e., Donsker), a main motivational scenario behind Section 2.3.

**Assumption 5 (Misspecified outcome models with constrained calibration)** *Let  $\tilde{\mu}_t(\cdot)$  and  $\tilde{\mu}_t^\dagger(\cdot)$  be two functions with finite variances,  $\mathbb{V}[\tilde{\mu}_t(X_i)], \mathbb{V}[\tilde{\mu}_t^\dagger(X_i)] < \infty$ , and  $\omega_t(\cdot) \in \mathcal{W}$  be some calibration weight. Then*

$$\begin{aligned} \frac{1}{\sqrt{|\mathcal{I}'|}} \sum_{i \in \mathcal{I}'} \left(1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)}\right) (\hat{\mu}_t(X_i; \mathcal{I}') - \tilde{\mu}_t(X_i)) &\xrightarrow{p} 0, \\ \frac{1}{\sqrt{|\mathcal{I}'|}} \sum_{i \in \mathcal{I}'} \left(1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)}\right) Y_i^\dagger(t; \mathcal{I}') (\hat{\omega}_t(X_i; \mathcal{I}') - \omega_t(X_i)) &\xrightarrow{p} 0, \\ \frac{1}{\sqrt{|\mathcal{I}'|}} \sum_{i \in \mathcal{I}'} \left(1 - \frac{\mathbb{1}(T_i = t)}{e_t(X_i)}\right) (\hat{\mu}_t^\dagger(X_i; \mathcal{I}') \hat{\omega}_t(X_i; \mathcal{I}') - \tilde{\mu}_t^\dagger(X_i) \omega_t(X_i)) &\xrightarrow{p} 0, \end{aligned}$$

for data folds  $\mathcal{I} \neq \mathcal{I}'$ .

We are now ready to present the following theorem.

**Theorem 2 ( $\hat{\mu}_{t,\text{CALM}}$  under misspecification and constrained calibration)** *Let Assumptions 1–3 and 5 hold, and  $B \rightarrow \infty$ . Also assume that  $\hat{\mu}_{t,\text{CALM}}$  is constructed via Step 4 (reduced cross-fitting) in Section 2.3. Then*

$$\sqrt{n}(\hat{\mu}_{t,\text{CALM}} - \bar{\mu}_{t,\text{CALM}}) \xrightarrow{\text{P}} 0, \quad \text{which implies} \quad \sqrt{n}(\hat{\mu}_{t,\text{CALM}} - \mu_t) \xrightarrow{\text{d}} \mathcal{N}(0, \mathbf{V}_{t,\text{CALM}}),$$

where  $\bar{\mu}_{t,\text{CALM}}$  and  $\mathbf{V}_{t,\text{CALM}}$  are defined in (3) and (4), respectively. If, in addition,  $\omega_t(\cdot)$  is the optimal constrained calibration weight and  $0 \in \mathcal{W}$ , then  $\mathbf{V}_{t,\text{CALM}} \leq \mathbf{V}_{t,\text{AIPW}}$ .

As we discussed above and in Section 2.3, the Donsker or stochastic equicontinuity type condition in Assumption 5 is most appropriate when the function classes used to estimate the nuisance functions are not “too large.” This encompasses both parametric models and classical nonparametric techniques, such as kernel- or sieve-based methods under additional regularity conditions on the bandwidth or sieve dimension. Given space constraints, however, we present below a special but important case of covariate discretization.

**Corollary 1 (Covariate discretization)** *Consider the discretized covariates approach introduced in Section 2.3, and assume that  $\mathbb{P}[X_i \in \mathcal{X}_j] > 0$  for all  $j = 1, 2, \dots, J$ . If Assumptions 1–3 hold, then Assumption 5 is satisfied, and the conclusions of Theorem 2 follow.*

We now turn to the average treatment effect estimator detailed in Section 3.2. Given the different structure of the calibration weights, we introduce the following assumption.

**Assumption 6 (Consistent nuisance function estimation)** *The cross-fitted function estimates are consistent:  $\|\hat{\mu}_t(\cdot; \mathcal{I}) - \mu_t(\cdot)\|_{L_2}$ ,  $\|\hat{\mu}_t^\dagger(\cdot; \mathcal{I}) - \mu_t^\dagger(\cdot)\|_{L_2}$  and  $\|\hat{\omega}_{\text{ATE}}(\cdot; \mathcal{I}) - \omega_{\text{ATE}}(\cdot)\|_{L_2} \xrightarrow{\text{P}} 0$  for each data fold  $\mathcal{I}$  and treatment arm  $t, t' \in \{1, \dots, k\}$ .*

Building on the insights of Section 3.2, the estimated ATE using CALM is consistent, asymptotically normally distributed, and achieves higher efficiency compared to the standard augmented inverse probability weighting approach.

**Theorem 3 (Asymptotic properties of  $\hat{\tau}_{t,t',\text{CALM}}$ )** *Let Assumptions 1–3 and 6 hold. In addition, assume the minimum eigenvalue of  $\Sigma^\dagger(X_i)$  is uniformly bounded away from 0, and  $B \rightarrow \infty$ . Then*

$$\sqrt{n}(\hat{\tau}_{t,t',\text{CALM}} - \bar{\tau}_{t,t',\text{CALM}}) \xrightarrow{\text{P}} 0, \quad \text{which implies} \quad \sqrt{n}(\hat{\tau}_{t,t',\text{CALM}} - \tau_{t,t'}) \xrightarrow{\text{d}} \mathcal{N}(0, \mathbf{V}_{t,t',\text{CALM}}),$$

where  $\bar{\tau}_{t,t',\text{CALM}}$  and  $\mathbf{V}_{t,t',\text{CALM}}$  are defined in (5) and (6), respectively.

Our final theoretical result establishes an asymptotic efficiency gain of the CALM-based conditional average treatment effect estimator, which we discussed in Section 3.3. To conserve space, additional regularity conditions are postponed to the Supplementary Materials.

**Theorem 4 (Asymptotic properties of  $\hat{\tau}_{t,t',\text{CALM}}(x)$ )** *Let Assumptions 1–3 hold,  $B \rightarrow \infty$ ,  $nh^p \rightarrow \infty$ , and  $nh^{p+2s} \lesssim 1$ , where  $s \geq 1$  denotes the smoothness of  $\mu_t(\cdot)$  and  $\mu_{t'}(\cdot)$ . Then under the additional regularity conditions laid out in the Supplementary Materials,*

$$\sqrt{nh^p}(\hat{\tau}_{t,t',\text{CALM}}(x) - \bar{\tau}_{t,t',\text{CALM}}(x)) \xrightarrow{\text{P}} 0,$$

where  $\bar{\tau}_{t,t',\text{CALM}}(x)$  is defined in (8), and therefore

$$\sqrt{nh^p}(\hat{\tau}_{t,t',\text{CALM}}(x) - \tau_{t,t'}(x)) \xrightarrow{\text{d}} \mathcal{N}(0, \mathbf{V}_{t,t',\text{CALM}}(x)),$$

and  $\mathbf{V}_{t,t',\text{CALM}}(x)$  is defined in (9).

## 5 Simulation Studies

We now evaluate the performance of CALM using synthetic data calibrated from the BRIGHTEN study [35]. Before detailing our complete simulation design and full results, we highlight four key findings. First, CALM consistently delivers estimates with higher efficiency while maintaining valid statistical inference (Figure 3). Second, augmented inverse probability weighting may exhibit substantial bias when few-shot predictions are naively incorporated as additional covariates, leading to systematic under-coverage (Figure 3, panels D–F). Third, surprisingly, we observe efficiency gains even in the absence of unstructured covariates, suggesting that CALM also benefits from intrinsic information encoded in pre-trained language models (Figure 4). Finally, CALM’s heterogeneous

calibration weighting mechanism selectively leverages counterfactual predictions when they are most informative, leading to substantial variance reduction (Figure 5).

## 5.1 Overview of BRIGHTEN Study

The BRIGHTEN study [35] comprises fully remote, smartphone-based randomized controlled trials conducted between 2016 and 2018 in the U.S., with the goal of evaluating the feasibility and effectiveness of delivering mental health care at scale through mobile applications. Recruitment targeted adults ( $\geq 18$  years old) exhibiting clinically significant depressive symptoms. In total, 2,193 subjects consented and enrolled. They were then randomized to one of the app-delivered interventions and followed for 12 weeks. To match our earlier notation,  $T_i \in \{0, 1\}$  indicates treatment assignment, to either the intervention group receiving internet-based Problem-Solving Therapy (iPST), or to a control group with alternative interventions. Structured covariates  $X_i$  capture demographic and clinical characteristics: sex, education, employment status, marital status, race, age, etc. The unstructured textual covariates  $Z_i$  include participants’ self-reported reasons for enrollment and their initial satisfaction with the mobile app. The outcome of interest,  $Y_i$ , is the PHQ-9 score measured after the trial concludes.

The BRIGHTEN study provides a unique data structure that integrates structured covariates with rich, unstructured baseline textual surveys, making it particularly well-suited for evaluating CALM (Figure 1, Panel A). Furthermore, baseline free-text responses regarding app-use motivation and pre-treatment app satisfaction exhibit the highest cosine similarity with discretized PHQ-9 scores, the primary outcome of the trial, demonstrating the potential of unstructured textual data as valuable prognostic factors (Figure 1, Panel B).

## 5.2 Simulating BRIGHTEN Study with Structured/Unstructured Data

In this subsection, we outline our approach to generating a synthetic superpopulation that closely mirrors the BRIGHTEN study (Figure 2). This endeavor presents three main methodological challenges. To begin with, modeling outcome-covariate relationships is complex when the feature set spans both structured variables (numerical and categorical) and unstructured text (free-text survey responses). Second, preserving dependencies across modalities requires embedding textual responses into continuous vector spaces, a step that introduces high-dimensionality concerns given

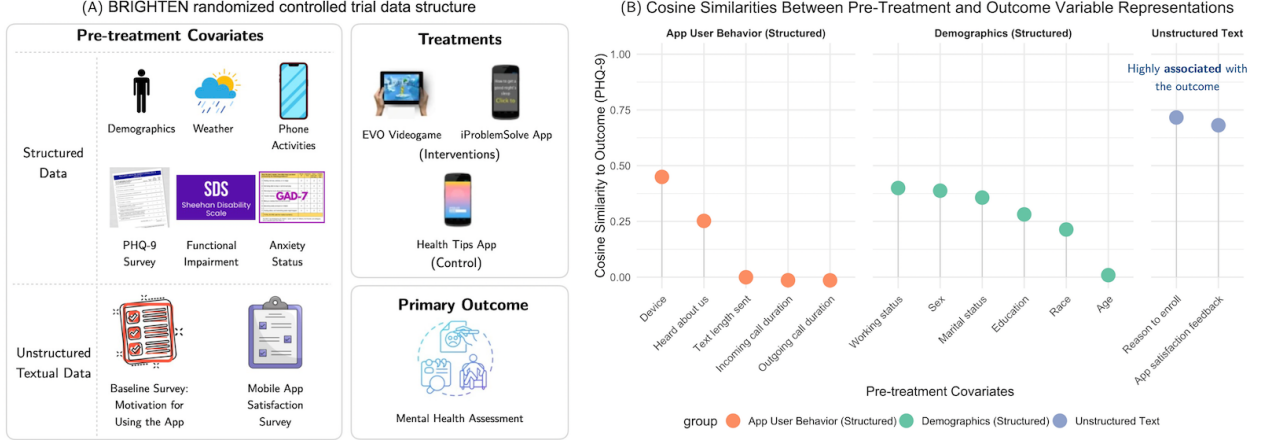


Figure 1: (A) BRIGHTEN study data structure. (B) Cosine similarity with the primary outcome.

the limited sample size of the original trial data. Third, as downstream tasks require natural language inputs, reconstructing text from these vector representations demands faithfully capturing the style, diversity, and semantic nuance of the original survey responses. Together, these hurdles make constructing a large-scale yet realistic synthetic population a non-trivial task.

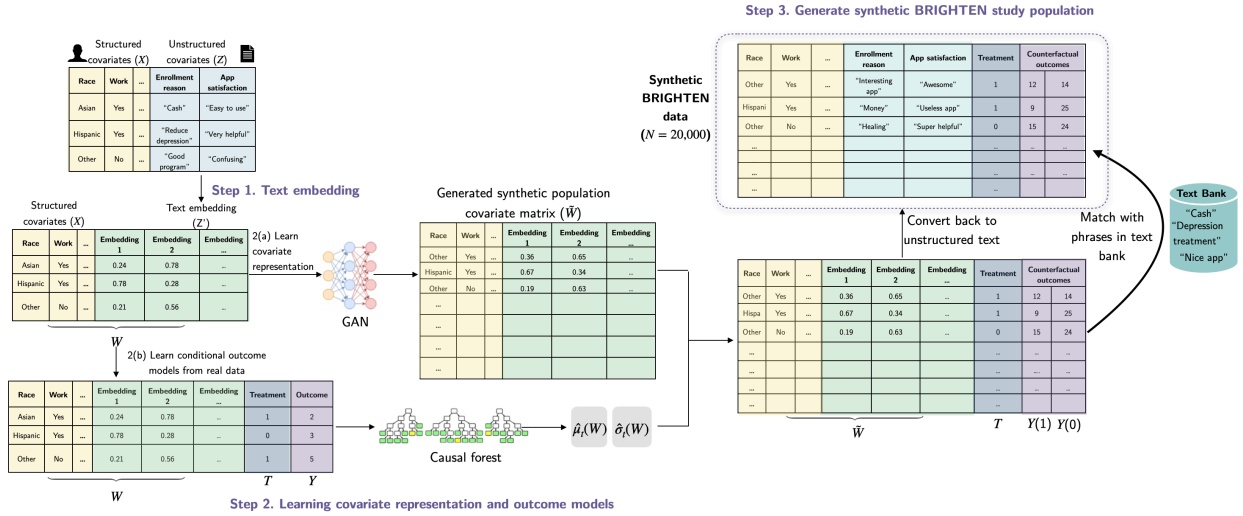


Figure 2: Illustration of the synthetic BRIGHTEN data generation.

The initial step involves preprocessing the covariates in the original BRIGHTEN study dataset. Structured numeric covariates are median-imputed and standardized, while categorical covariates are one-hot encoded. Unstructured covariates, such as survey responses, are first normalized so that missing entries are mapped to the string "No response." We then embed each unstructured covariate separately using the pretrained sentence transformer all-MiniLM-L6-v2 [45], applying padding,

truncation, and  $L_2$ -normalization to obtain dense text embeddings. We adopt **all-MiniLM-L6-v2** because it is a compact, contrastively trained transformer model that produces high-quality sentence embeddings with well-preserved semantic structure. Finally, embeddings from the two unstructured covariate columns are combined with the processed structured covariates to form the final covariate representation, denoted as  $W_i = (X_i, Z'_i)$ , where  $Z'_i$  represents the embedding vectors.

Second, we learn both a generative model for the covariates and a predictive model for the outcomes. Specifically, the joint covariate distribution  $p(W)$  is modeled using a conditional Generative Adversarial Network (CTGAN) [46]. To improve sample efficiency and training stability, each text-embedding block is compressed via principal component analysis (PCA) to 32 components. The structured covariates and PCA-compressed embeddings are then concatenated to form a unified training table, whose schema is validated with **Metadata** before fitting a **CTGANSynthesizer** [33]. This yields a generator  $G$  that models  $\hat{p}(W)$ , which we use to draw synthetic covariates for downstream simulations. Next, using the embedded BRIGHTEN dataset  $\{W_i, T_i, Y_i\}_{i=1}^n$ , we fit causal forests to learn the conditional mean and conditional variance,  $m_t(W) = \mathbb{E}[Y|W, T = t]$  and  $s_t^2(W) = \mathbb{V}[Y|W, T = t]$  for  $t \in \{0, 1\}$ , using the R package **grf** [43]. We denote the fitted models as  $\hat{m}_t(W)$  and  $\hat{s}_t^2(W)$ . Finally, we categorize the outcome variable into five clinically standard levels of depression severity: minimal, mild, moderate, high, and severe, to simplify prediction tasks for LLMs, as they are more effective when predicting discrete, semantically meaningful categories rather than continuous numeric scores.

The next step is to generate a synthetic super-population of size  $N = 20,000$ . Using the generator learned in the second step, we first form the synthetic covariate matrix, denoted as  $\widetilde{W}_j = (\widetilde{X}_j, \widetilde{Z}'_j)$  for  $j = 1, \dots, N$ . We then generate the treatment variable as  $\widetilde{T}_j \sim \text{Bernoulli}(0.5)$ ,  $j = 1, \dots, N$ , and the potential outcomes as  $\widetilde{Y}_j(t) = \hat{m}_t(\widetilde{W}_j) + \hat{s}_t(\widetilde{W}_j)\varepsilon_j$ ,  $\varepsilon_j \sim N(0, 1)$ ,  $t \in \{0, 1\}$ . The observed synthetic outcomes are therefore  $\widetilde{Y}_j = \widetilde{T}_j \cdot \widetilde{Y}_j(1) + (1 - \widetilde{T}_j) \cdot \widetilde{Y}_j(0)$ ,  $j = 1, \dots, N$ . The true ATE based on the synthetic population is computed as  $\tau \equiv \tau_{1,0} = \frac{1}{N} \sum_{j=1}^N \{\widetilde{Y}_j(1) - \widetilde{Y}_j(0)\}$ .

Lastly, because the generated covariates  $\widetilde{W}_j$  consist of  $(\widetilde{X}_j, \widetilde{Z}'_j)$ , where  $\widetilde{Z}'_j$  is represented in the numerical embedding space, we recover  $\widetilde{Z}'_j$  back into unstructured text. To this end, we first establish a phrase bank for each free-text field (impression of the mobile app, reasons for enrolling) using responses from the original BRIGHTEN dataset after normalizing the raw texts. The surviving unique strings form the seed set  $\mathcal{S}_k$  for unstructured covariate column  $k$ , where

$k \in \{1, 2\}$ . We then paraphrase each seed using pretrained LLMs, Pre-training with Extracted Gap-sentences for Abstractive Summarization (PEGASUS) and Text-to-Text Transfer Transformer (T5), to increase lexical and semantic diversity. PEGASUS is a transformer model specialized for text summarization and paraphrase generation, trained with a gap-sentence generation objective that makes it highly effective for producing semantically faithful variations of input text [48]. T5 (Text-to-Text Transfer Transformer) is a general-purpose encoder-decoder transformer that frames all NLP tasks in a unified text-to-text format, enabling effective paraphrase generation among other tasks [37]. Candidate paraphrases are filtered by enforcing length constraints and requiring cosine similarity in the PCA-compressed embedding space to fall within a semantic band-pass window  $[0.65, 0.98]$ , relative to the original seed. This procedure yields the final phrase bank  $\mathcal{B}_k$  for each unstructured covariate column  $k \in \{1, 2\}$ , with their size fixed at 50,000. All phrases in  $\mathcal{B}_k$  are then re-embedded using the same encoder applied in the first step and projected through the saved PCA transformation. For each row of  $\tilde{Z}'_i$ , we perform top-1 nearest neighbor retrieval to identify the closest matching string, thereby recovering natural-language representations  $\tilde{Z}_i$  corresponding to the embedding vectors  $\tilde{Z}'_i$ .

### 5.3 Benchmarking CALM with Other Methods

Using the synthetic BRIGHTEN super population generated in the previous section, we conduct simulation studies using 300 Monte Carlo iterations. Each Monte Carlo sample is randomly drawn from this synthetic population with sample sizes ranging from  $n = 400$  to 2,000. This subsection focuses on comparing CALM to alternative methods. The next two subsections offer additional insights into CALM’s performance.

We consider several variants of CALM: zero-shot learning ( $m = 0$ , as discussed in Section 3.1) and few-shot learning with  $m \in \{6, 10, 14\}$  demonstrative examples and  $B = 200$  resampling aggregation (Section 2.2). GPT-4o-mini is employed to generate counterfactual outcome predictions. For further context and robustness, the following case study in Section 6 includes additional results using gemini-2.5-flash and GPT-3.5-turbo as alternative LLMs for generating counterfactual outcomes. An illustrative example of zero- and few-shot learning prompts is provided in the Supplementary Materials.

We compare CALM to several variants of AIPW: (1) AIPW: the standard AIPW estimator using

only structured covariates; (2) AIPW (zero-shot covariate): the AIPW estimator with zero-shot predictions as additional covariates; (3) AIPW (few-shot covariate): the AIPW estimator with few-shot predictions as additional covariates. For both the CALM and AIPW methods, we use random forests to estimate conditional outcome models. A comparison of performance using random forests versus gradient boosting is provided in Supplementary Materials. We assess performance based on absolute bias, the  $\sqrt{n}$ -scaled standard deviation (SD) of the ATE estimates, and coverage probability of 95% confidence intervals. Results are summarized in Figure 3.

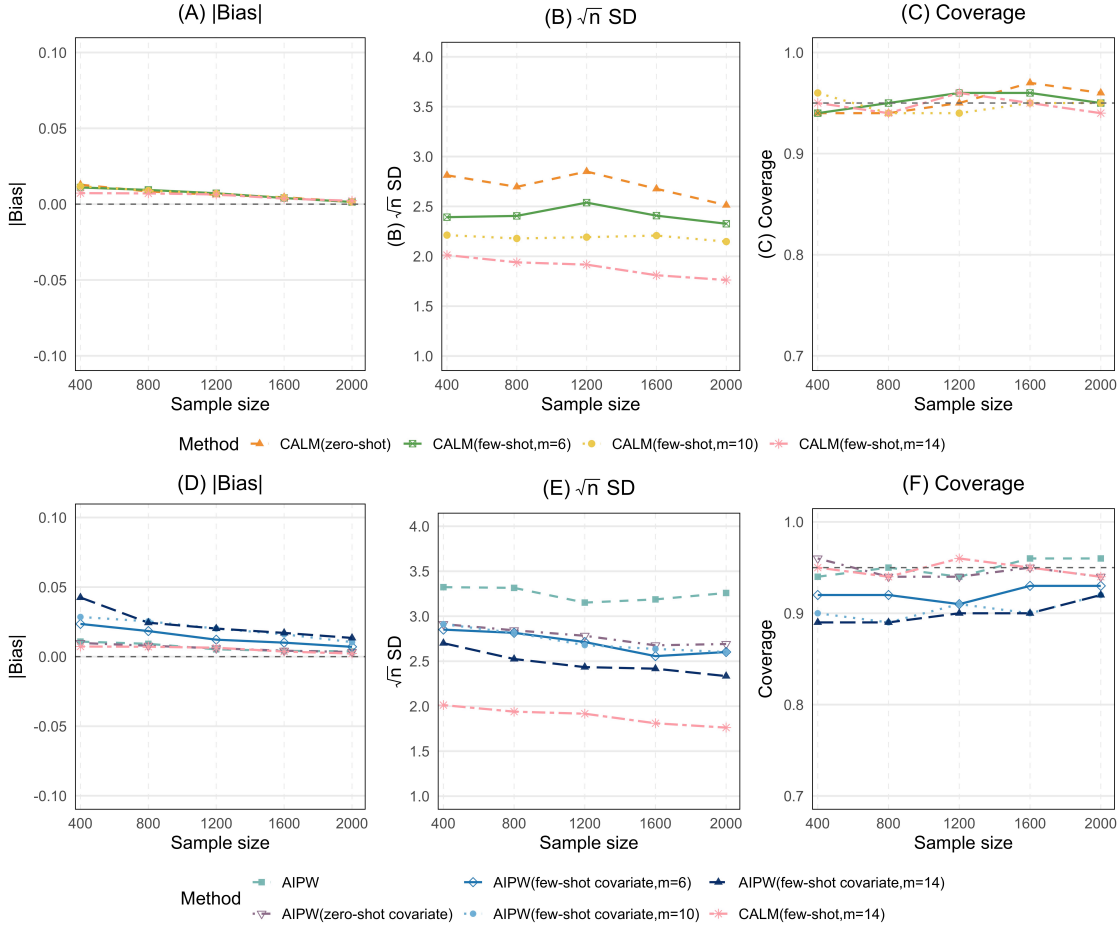


Figure 3: Panel (A)–(C): Comparison of CALM with different number of demonstrative examples:  $m = 0$  (zero-shot) and  $m \in \{6, 10, 14\}$  (few-shot). Panel (D)–(F): Comparison with AIPW-based methods.

Panel (A)–(C) of Figure 3 compares zero-shot and few-shot CALM methods. The results show that increasing the number of few-shot examples in the prompt leads to progressively greater efficiency gains, with  $m = 14$  achieving the lowest standard deviation. We conjecture that this is

because few-shot learning provides additional information and guidance to the LLM in generating outcomes, leading to stronger correlation with the observed outcomes and consequently improving efficiency in the estimated treatment effects. The validity of statistical inference of the CALM method is further supported by the coverage probabilities, where CALM with zero-shot or few-shot learning achieves a nearly nominal level of coverage.

Panels (D)–(F) of Figure 3 show that CALM achieves substantial efficiency gains compared to all evaluated AIPW-based methods. Furthermore, while standard AIPW and zero-shot AIPW yield consistent estimates of the ATE, naively incorporating few-shot predictions as covariates into AIPW introduces bias that scales with the number of prompt examples. This growing bias leads to poor coverage performance. We hypothesize that this bias stems from sample reuse when generating LLM-based counterfactual predictions via few-shot prompting. In contrast, using zero-shot LLM predictions in AIPW avoids this bias and maintains nominal coverage.

#### 5.4 Does Pre-trained LLM Contain Relevant Information?

As outlined in the motivation for our proposed method, leveraging LLMs to generate outcome predictions offers two main advantages: the ability to exploit the knowledge embedded in pre-trained language models, and the seamless integration of unstructured covariates. These considerations naturally raise two questions: How much of the gain can be attributed to the prior knowledge encoded in pre-trained LLMs themselves? And to what extent do unstructured covariates enhance the informativeness of LLM predictions, and consequently improve the efficiency of treatment effect estimation?

In this section, we consider four variations of the CALM methods with different counterfactual-outcome generation mechanisms: (i) **CALM(zero-shot,  $X + Z$ )** that incorporates zero-shot predictions formed with both  $X_i$  and  $Z_i$ ; (ii) **CALM(zero-shot,  $X$ )** where the zero-shot predictions are generated using only structured information  $X_i$ ; (iii) **CALM(RF,  $X$ )** for which counterfactual predictions are from a random forest using only structured  $X_i$ , instead of a language model; (iv) **CALM(GB,  $X$ )**, which is similar to (iii) with the only difference that gradient boosting is employed to construct counterfactual predictions. Monte Carlo standard deviations of the estimates are reported in Panel (A) of Figure 4, and we also show the calibration weights for a specific data stratum in Panel (B) of the same figure.

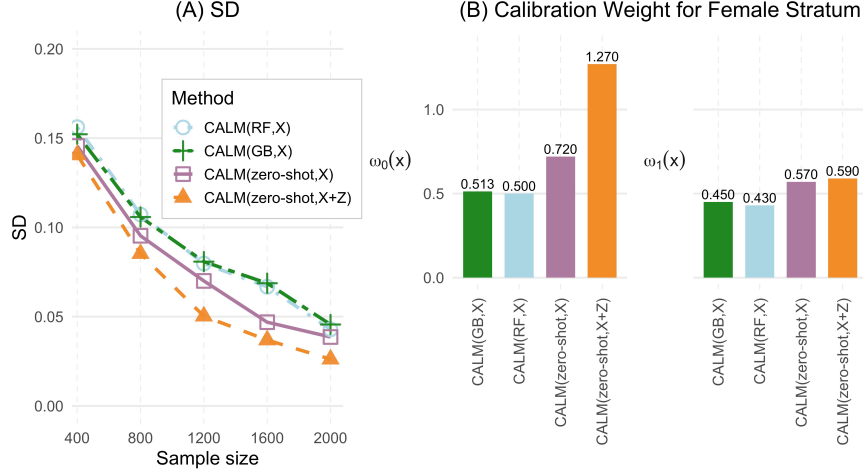


Figure 4: Panel (A): Monte Carlo standard deviation under different counterfactual outcome prediction mechanisms. **CALM(zero-shot,X)** refers to the CALM zero-shot learning method where only the structured covariates  $X_i$  are included in the prompt. **CALM(zero-shot,X + Z)** incorporates both structured  $X_i$  and unstructured  $Z_i$  information. **CALM(RF,X)** and **CALM(GB,X)** do not employ any language model for counterfactual prediction. They instead use random forest and gradient boosting methods. Panel (B): An example of the calibration weight under different counterfactual outcome prediction mechanisms in the female stratum.

Figure 4 demonstrates that our proposed method with zero-shot learning, whether or not unstructured covariates are included in the prompt, yields higher estimation efficiency compared to using standard machine learning methods alone. Moreover, incorporating unstructured covariates into the zero-shot prompt provides an additional efficiency gain. These results suggest that the pre-trained LLM (GPT-4o-mini in this setting) encodes knowledge that improves the efficiency of ATE estimation. Even without informative unstructured inputs, the LLM’s exposure to large and diverse training corpora may enable it to capture latent associations between covariates and outcomes, which CALM can then exploit to enhance efficiency. Incorporating unstructured covariates into the prompt further amplifies this benefit.

### 5.5 How Informative are LLM-based Predictions Through $\omega_t(x)$ ?

Next, we provide some insights on the informativeness of LLM-based predictions through the lens of the calibration weights  $\omega_t(x)$ . Specifically, we examine the relationship between the LLM-predicted outcomes and the observed outcomes across different covariate strata. This analysis is motivated by the inherent structure of the CALM method, which selectively borrows information from distinct regions of the covariate space, depending on where the LLM predictions are most informative.

To better demonstrate the mechanism underlying the efficiency gain, we show the calibration weight  $\omega_t(x)$  between the LLM-predicted outcomes and observed outcomes in Figure 5 (A) across strata defined by gender and race. To quantify how these calibration weights translate into improved efficiency, we report the percentage reduction in estimator variance between the standard AIPW method and the CALM method with zero-shot learning in Figure 5 (B). Specifically, we compute this percentage variance reduction by subtracting the empirical variance under CALM from that under AIPW, dividing the difference by the empirical variance under AIPW, and multiplying by 100.

Figure 5 suggests substantial heterogeneity in the strength of these correlations, across strata and between treatment arms. In some strata, LLM-predicted outcomes exhibit strong alignment with the observed outcomes (e.g., American Indian/Alaskan Native), while in others the alignment is weaker (e.g., Other race). This pattern suggests that the informativeness of the LLM-based predictions is not uniform across the covariate space. This heterogeneity plays a significant role in our proposed CALM method. Rather than uniformly leveraging the LLM predictions, CALM adaptively borrows strength in regions of the covariate space where the predictions are most reliable. To evaluate the practical implications of this selective borrowing, we compare the variance of the CALM estimator to that of the AIPW estimator across strata. Figure 5 (B) reports the percentage reduction in variance achieved by CALM within each stratum. The results show that the degree of variance reduction achieved by CALM aligns closely with the strength of the calibration weight shown in Figure 5 (A). In strata where LLM predictions are highly aligned with observed outcomes, CALM yields greater variance reductions. Together, these findings suggest that the calibration weight of selectively borrowing information from LLM-generated predictions contributes to the improved estimation efficiency of our proposed method.

## 6 Empirical Illustration: The BRIGHTEN Study

In this section, we apply our proposed method to the real BRIGHTEN study dataset. We summarize the takeaways from the case study section as follows: First, the CALM-based methods achieve higher estimation efficiency and are able to detect significant treatment effects in the female Hispanic subgroup compared to the benchmark approaches. Second, the CALM-based methods attain

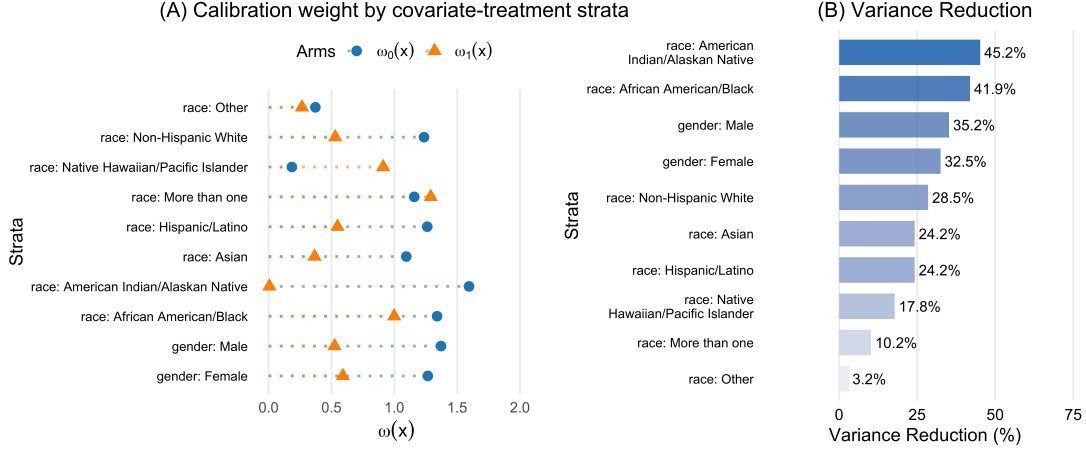


Figure 5: Panel (A): Calibration weight  $\omega_t(x)$  for the “race” and “gender” strata using the CALM zero-shot learning method ( $n = 2,000$ ). Panel (B): the variance reduction in comparison with the AIPW estimator.

slightly greater efficiency when generating the counterfactual prediction under **GPT-4o-mini** or **gemini-2.5-flash**, compared to **GPT-3.5-turbo**. Third, CALM’s performance is robust to variations in prompt design, demonstrating invariance to different prompting strategies.

We begin by estimating the ATE using CALM and the benchmark methods. The resulting point estimates and 95% confidence intervals are shown in Figure 6 (A). Consistent with the simulation findings, CALM with few-shot learning yields the narrowest confidence intervals, reflecting higher estimation efficiency. In contrast, AIPW augmented with few-shot learning produces a significantly negative treatment effect estimate, which is likely attributable to the estimation bias documented in Section 5.3. To assess the impact of different LLMs on CALM’s performance, we compare results using three models: **GPT-4o-mini**, **GPT-3.5-turbo**, and **gemini-2.5-flash**, as shown in Figure 6 (B). The figure shows that **GPT-4o-mini** and **gemini-2.5-flash** achieve comparable efficiency, whereas **GPT-3.5-turbo** yields slightly wider confidence intervals, suggesting reduced estimation efficiency due to less informative LLM-based predictions relative to the other two models.

Furthermore, we investigate strata-level treatment effects, with results presented in Figure 7. The figure shows that CALM with zero-shot or few-shot learning is able to detect a significant treatment effect among female Hispanic participants, which is not identified by the benchmark AIPW methods. This finding highlights how the efficiency gains achieved by CALM translate into improved power for subgroup analyses in the BRIGHTEN study, enabling the detection of

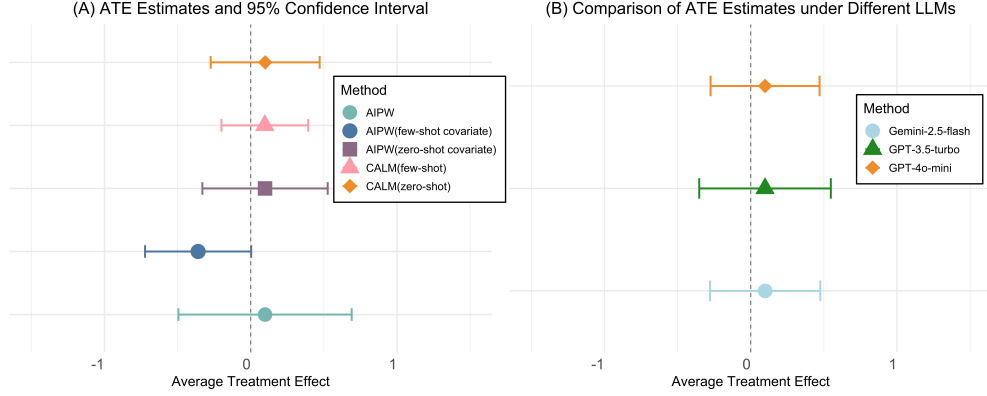


Figure 6: Panel (A): Comparison of the ATE estimates and the associated 95% confidence intervals from the BRIGHTEN study under AIPW and CALM-based methods. Panel (B): Comparison across different language models for generating counterfactual outcomes under CALM zero-shot learning method.

significant treatment effect heterogeneity of the mobile app intervention. In this case, the enhanced efficiency also facilitates scientific discovery.

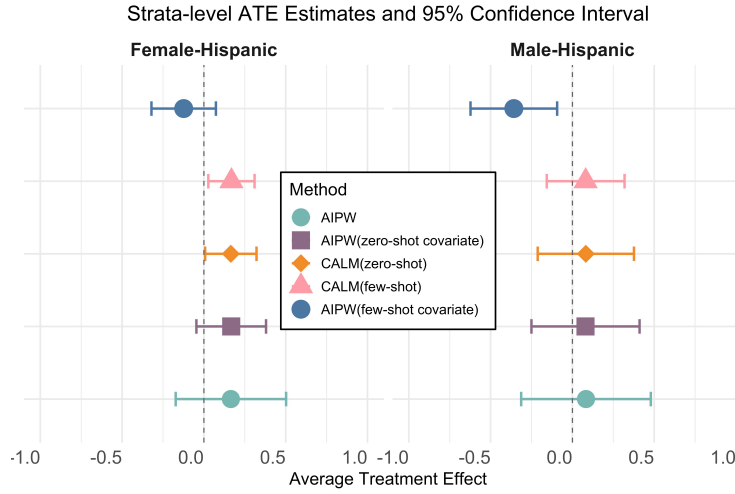


Figure 7: Comparison of the ATE estimates and the associated 95% confidence intervals from the BRIGHTEN study under AIPW and CALM-based methods in female Hispanic and male Hispanic strata.

Additionally, in the preceding simulation studies, we fix a single prompt engineering technique for generating LLM-based predictions. A natural question is whether these predictions are sensitive to prompt design, and whether alternative formulations might alter estimation performance within the CALM framework. To assess this, we conduct a robustness analysis using four prompt engineering strategies: (1) self-consistency, which generates multiple predictions under different sampling

seeds and aggregates them to reduce variance; (2) role-based prompting, which assigns the LLM a specific role (e.g., “You are an experienced clinical researcher”) to provide domain-specific context; (3) decomposition prompting, which breaks the task into sequential sub-questions to capture complex covariate dependencies; and (4) contrastive prompting, which presents systematically varied hypothetical cases to highlight differences in predicted outcomes. Examples of these prompt designs and the ATE estimates with associated 95% confidence intervals are shown in Figure 8.

Figure 8 shows that all four prompting strategies yield comparable point estimates and confidence intervals, regardless of the specific prompt design. Among them, role-based prompting produces slightly narrower intervals, indicating a modest efficiency gain, though the difference is small and not statistically significant relative to the other methods. Overall, the results suggest that CALM is robust to variations in prompt engineering, even when the structure and contextual framing differ substantially. This stability under different prompt engineering implies that CALM’s performance does not critically depend on prompt format, enhancing its practical applicability in settings where prompt design may vary.

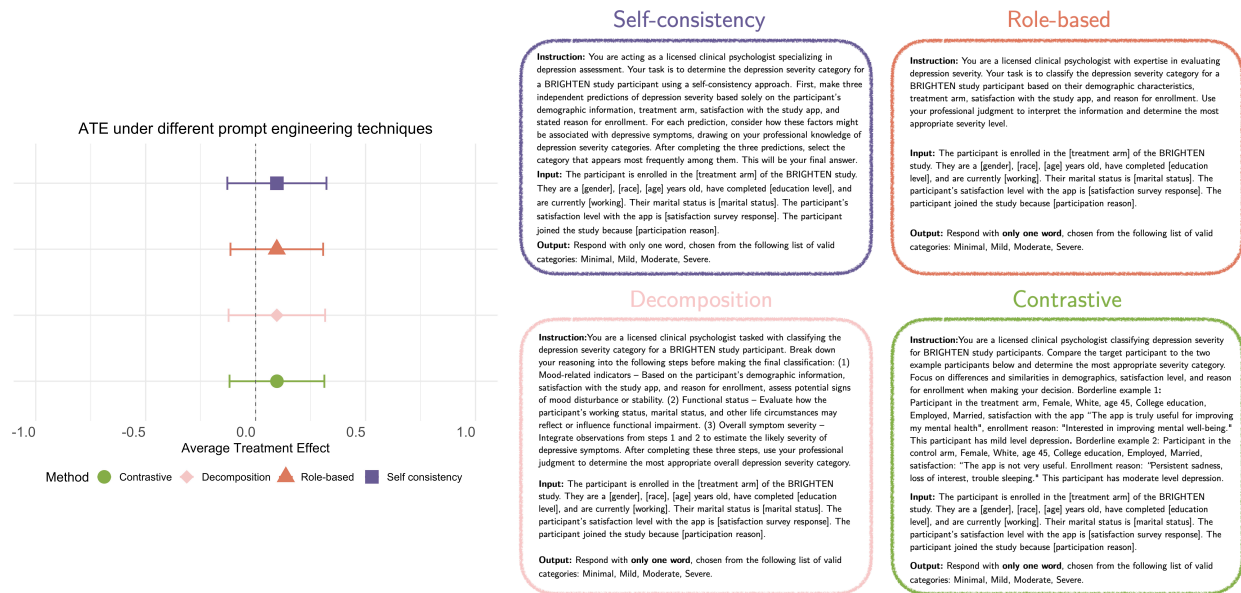


Figure 8: Comparison of ATE and the associated 95% confidence intervals under four different prompt engineering techniques: self-consistency, role-based, decomposition, and contrastive methods.

## 7 Conclusion

In this manuscript, we introduced **CALM**, a statistically principled framework for leveraging LLM-generated predictions in randomized experiments. By residualizing and heterogeneously calibrating LLM predictions, CALM yields consistent estimators with valid inference even when zero- and few-shot predictions are biased, and it empirically improves efficiency relative to covariate-adjusted AIPW for mean potential outcomes; extensions to average and heterogeneous treatment effects follow directly. In simulations calibrated to the BRIGHTEN RCT, CALM reduces variance and increases power for detecting treatment-effect heterogeneity while remaining robust to prompt design.

### Data availability

The BRIGHTEN study data are publicly accessible upon request via Synapse:

<https://www.synapse.org/Synapse:syn10848316/wiki/548727> (Synapse Project:syn10848316).

## References

- [1] Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y. *et al.* (2022). “Flamingo: A Visual Language Model for Few-Shot Learning,” In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, 23716–23730.
- [2] Angelopoulos, A. N., Bates, S., Fannjiang, C., Jordan, M. I., and Zrnic, T. (2023). “Prediction-Powered Inference,” *Science*, 382(6671), 669–674.
- [3] Angelopoulos, A. N., Duchi, J. C., and Zrnic, T. (2024). “Ppi++: Efficient Prediction-Powered Inference,” arXiv:2311.01453.
- [4] Arbour, D., Ben-Michael, E., Feller, A., Lal, A., and Yuan, L.-H. (2026). “AI-Assisted Variance Reduction in Randomized Experiments,” arXiv:2606.08853.
- [5] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P. *et al.* (2020). “Language Models Are Few-Shot Learners,” In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 1877–1901.

- [6] Bryan, E. G., Chen, H., Vilaro, M., Chu, H., Grillo, G., Te, P. *et al.* (2025). “Developing a Supportive Virtual Human to Deliver Clinical Trial Education for Older Women and Other Populations Historically Excluded From Research,” *Patient Education and Counseling*, 130, 108485.
- [7] Carlson, J. and Dell, M. (2026). “A Unifying Framework for Robust and Efficient Inference with Unstructured Data,” arXiv:2505.00282.
- [8] Cattaneo, M. D. (2010). “Efficient Semiparametric Estimation of Multi-Valued Treatment Effects Under Ignorability,” *Journal of Econometrics*, 155(2), 138–154.
- [9] Cattaneo, M. D. and Jansson, M. (2018). “Kernel-Based Semiparametric Estimators: Small Bandwidth Asymptotics and Bootstrap Consistency,” *Econometrica*, 86(3), 955–995.
- [10] Cattaneo, M. D., Jansson, M., and Ma, X. (2019). “Two-Step Estimation and Inference with Possibly Many Included Covariates,” *The Review of Economic Studies*, 86(3), 1095–1122.
- [11] Chen, X., Hong, H., and Tarozi, A. (2008). “Semiparametric Efficiency in GMM Models with Auxiliary Data,” *The Annals of Statistics*, 36(2), 808–843.
- [12] Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W. *et al.* (2018). “Double/Debiased Machine Learning for Treatment and Structural Parameters,” *The Econometrics Journal*, 21(1), C1–C68.
- [13] Coravos, A., Goldsack, J. C., Karlin, D. R., Nebeker, C., Perakslis, E., and Zimmerman, N. (2020). “Modernizing and Designing Evaluation Frameworks for Connected Sensor Technologies in Medicine,” *npj Digital Medicine*, 3(37), 1–10.
- [14] De Bartolomeis, P., Abad, J., Wang, G., Donhauser, K., Duch, R. M., Yang, F. *et al.* (2025). “Efficient Randomized Experiments Using Foundation Models,” arXiv:2502.04262.
- [15] Driess, D., Xia, F., Sajjadi, M. S. M., Lynch, C., Chowdhery, A., Ichter, B. *et al.* (2023). “PaLM-E: An Embodied Multimodal Language Model,” In *Proceedings of the 40th International Conference on Machine Learning*, 8469–8488.

- [16] Egami, N., Hinck, M., Stewart, B., and Wei, H. (2023). “Using Imperfect Surrogates for Downstream Inference: Design-Based Supervised Learning for Social Science Applications of Large Language Models,” In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 68589–68601.
- [17] Fisch, A., Maynez, J., Hofer, R., Dhingra, B., Globerson, A., and Cohen, W. W. (2024). “Stratified Prediction-Powered Inference for Effective Hybrid Evaluation of Language Models,” In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, 111489–111514.
- [18] Freedman, D. A. (2008). “On Regression Adjustments to Experimental Data,” *Advances in Applied Mathematics*, 40(2), 180–193.
- [19] Gao, T., Fisch, A., and Chen, D. (2021). “Making Pre-Trained Language Models Better Few-Shot Learners,” In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 3816–3830.
- [20] Guterman, E. L., Kiekhoefer, R. E., Wood, A. J., Allen, I. E., Kahn, J. G., Dulaney, S. *et al.* (2023). “Care Ecosystem Collaborative Model and Health Care Costs in Medicare Beneficiaries with Dementia: A Secondary Analysis of a Randomized Clinical Trial,” *JAMA Internal Medicine*, 183(11), 1222–1228.
- [21] Hahn, J. (1998). “On the Role of the Propensity Score in Efficient Semiparametric Estimation of Average Treatment Effects,” *Econometrica*, 66(2), 315–331.
- [22] Hanvey, G., Dede, D., Krieger, J., Ross, K., Amdur, R., and Pereira, D. (2024). “Investigating Inequities in Cancer Clinical Trial Participation: Understanding Demographic and Socioeconomic Determinants of Behavioral Cancer Research Participation,” *Psychosomatic Medicine*, 86(5), A50.
- [23] Hirano, K., Imbens, G. W., and Ridder, G. (2003). “Efficient Estimation of Average Treatment Effects Using the Estimated Propensity Score,” *Econometrica*, 71(4), 1161–1189.

- [24] Li, J., Li, D., Savarese, S., and Hoi, S. (2023). “BLIP-2: Bootstrapping Language-Image Pre-Training with Frozen Image Encoders and Large Language Models,” In *Proceedings of the 40th International Conference on Machine Learning*, 19730–19742.
- [25] Lin, W. (2013). “Agnostic Notes on Regression Adjustments to Experimental Data: Reexamining Freedman’s Critique,” *The Annals of Applied Statistics*, 7(1), 295–318.
- [26] Liu, H., Li, C., Wu, Q., and Lee, Y. J. (2023). “Visual Instruction Tuning,” In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, 34892–34916.
- [27] Ludwig, J., Mullainathan, S., and Rambachan, A. (2025). “Large Language Models: An Applied Econometric Framework,” arXiv:2412.07031.
- [28] Luo, R., Sun, L., Xia, Y., Qin, T., Zhang, S., Poon, H. *et al.* (2022). “BioGPT: Generative Pre-Trained Transformer for Biomedical Text Generation and Mining,” *Briefings in Bioinformatics*, 23(6), bbac409.
- [29] Ma, X., Sasaki, Y., and Wang, Y. (2025). “Testing Limited Overlap,” *Econometric Theory*, 41(5), 1129–1162.
- [30] Ma, X. and Wang, J. (2020). “Robust Inference Using Inverse Probability Weighting,” *Journal of the American Statistical Association*, 115(532), 1851–1860.
- [31] Min, S., Lyu, X., Holtzman, A., Artetxe, M., Lewis, M., Hajishirzi, H., and Zettlemoyer, L. (2022). “Rethinking the Role of Demonstrations: What Makes In-Context Learning Work?” In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 11048–11064.
- [32] OpenAI *et al.* (2023). “GPT-4 Technical Report,” arXiv preprint arXiv:2303.08774.
- [33] Patki, N., Wedge, R., and Veeramachaneni, K. (2016). “The Synthetic Data Vault,” In *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 399–410.
- [34] Poulet, P.-E., Tran, M., du Montcel, S. T., Dubois, B., Durrleman, S., and Jedynek, B. (2025). “Prediction-Powered Inference for Clinical Trials,” medRxiv.

- [35] Pratap, A., Homiar, A., Waninger, L., Herd, C., Suver, C., Volponi, J. *et al.* (2022). “Real-World Behavioral Dataset from Two Fully Remote Smartphone-Based Randomized Clinical Trials for Depression,” *Scientific Data*, 9(1), 522.
- [36] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S. *et al.* (2021). “Learning Transferable Visual Models From Natural Language Supervision,” In *Proceedings of the 38th International Conference on Machine Learning*, 8748–8763.
- [37] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M. *et al.* (2020). “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer,” *Journal of Machine Learning Research*, 21(140), 1–67.
- [38] Ramos, F. N., Bernstein, R. A., and Ezawa, I. D. (2024). “Assessing Predictive Factors of Attitudes Toward Peer-Supported Mental Health Interventions in the Metaverse: Mixed Methods Study,” *JMIR XR and Spatial Computing (JMXR)*, 1(1), e57990.
- [39] Robinson, P. (1988). “Root-N-Consistent Semiparametric Regression,” *Econometrica*, 56(4), 931–954.
- [40] Ruan, X., Zhao, Z., Wei, W., Zhang, Y., Zheng, Z., Huang, S., and Wang, J. (2026). “HERO: Improving the Reliability and Sensitivity of Generative Model Evaluation Using Historical Data,” arXiv:2606.29784.
- [41] Schick, T. and Schütze, H. (2021). “Exploiting Cloze Questions for Few Shot Text Classification and Natural Language Inference,” arXiv:2001.07676.
- [42] Taylor, R., Kardas, M., Cucurull, G., Scialom, T., Hartshorn, A., Saravia, E. *et al.* (2022). “Galactica: A Large Language Model for Science,” arXiv:2211.09085.
- [43] Tibshirani, J., Athey, S., Wager, S., Friedberg, R., Hadad, V., Miner, L. *et al.* (2024). “GRF: Generalized Random Forests,” R package version 2.3.2.
- [44] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T. *et al.* (2023). “Llama: Open and Efficient Foundation Language Models,” arXiv:2302.13971.

- [45] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., and Zhou, M. (2020). “Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers,” In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 5776–5788.
- [46] Xu, L., Skoularidou, M., Cuesta-Infante, A., and Veeramachaneni, K. (2019). “Modeling Tabular Data Using Conditional GAN,” In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 7335–7345.
- [47] Zhang, B., Li, J., Hortaçsu, A., Ye, X., Chernozhukov, V., Ni, A. *et al.* (2025). “Agentic Economic Modeling,” arXiv:2510.25743.
- [48] Zhang, J., Zhao, Y., Saleh, M., and Liu, P. (2020). “Pegasus: Pre-Training with Extracted Gap-Sentences for Abstractive Summarization,” In *Proceedings of the 37th International Conference on Machine Learning*, 11328–11339.
- [49] Zhao, Z., Wallace, E., Feng, S., Klein, D., and Singh, S. (2021). “Calibrate Before Use: Improving Few-Shot Performance of Language Models,” In *Proceedings of the 38th International Conference on Machine Learning*, 139, 12697–12706.
- [50] Zrnica, T. and Candès, E. J. (2024). “Cross-Prediction-Powered Inference,” *Proceedings of the National Academy of Sciences*, 121(15), e2322083121.