ECON 7377 Microeconometrics **Spring 2026**

AI/ML Generated Regressors

Contents

Overview. Economists increasingly use variables constructed by AI/ML algorithms in downstream econometric analysis, such as poverty measured from satellite images, sentiment extracted from text, uncertainty indices from news articles, or topic weights from time-use data. Such variables are typically estimated with substantial error and unknown statistical properties. This lecture covers three recent papers that provide rigorous frameworks for using AI/ML outputs in econometric analyses.

- Rambachan et al. (2024) study program evaluation when the *outcome* is missing in the experimental sample and researchers instead observe a remotely sensed proxy, for example when crop burning is inferred from satellite imagery and true outcomes are available only from auxiliary spot checks. Their focus is the treatment effect in a randomized experiment.
- Battaglia et al. (2024) study downstream linear regressions in which a *regressor* is generated by an AI/ML algorithm, such as an imputed label, a topic weight, or a sentiment index, and plugged in as if it were data.
- Ludwig et al. (2024) provide a black-box econometric framework specific to large language models (LLMs). They distinguish two empirical uses of LLM outputs: *prediction problems*, which require "no training leakage" between the LLM's training corpus and the researcher's evaluation sample; and *estimation problems*, where LLM outputs serve as imperfect measurements of an economic concept and require a small *validation sample* for valid downstream inference.

In the estimation settings, the naive "two-step" practice of plugging AI/ML outputs in as if they were the true variables leads to biased estimates or invalid inference. For LLM prediction problems, the central risk is instead training leakage. Each paper proposes a different theoretical lens and a corresponding fix. Section 1 covers the program-evaluation setting; Section 2 covers the linear-regression setting; Section 3 covers the LLM-specific framework.

Notation. Throughout this lecture we use $D \in \{0, 1\}$ for a binary treatment, Y for an outcome of interest, and $X \in \mathcal{X}$ for pre-treatment covariates. Potential outcomes are $(Y(0), Y(1))$ with the SUTVA representation $Y = DY(1) + (1 - D)Y(0)$. The symbol $\mathbb{E}$ denotes expectation under the relevant distribution, and Pr denotes probability. Sections 1, 2, and 3 introduce additional, paper-specific notation as needed. Each paper uses the symbol θ for a different quantity, and we will be explicit about its meaning in each section. Section 3 indexes observations by a text piece r rather than a unit i and uses D_r as a *sampling* indicator (whether the researcher collected text piece r); this D_r should not be confused with the binary treatment D of Section 1.

Zhan Gao, April 27, 2026

1 Program Evaluation with Remotely Sensed Outcomes

This section is based on Rambachan et al. (2024).

1.1 Motivation

- Traditional program evaluation relies on surveys to collect economic outcomes, but high-quality outcome measurements (poverty, deforestation, fires, air pollution) are often prohibitively expensive or infeasible to obtain for every experimental unit.
- Researchers increasingly turn to *remotely sensed variables* (RSVs) as substitutes, including satellite images, nightlights, or mobile phone traces that can be collected cheaply at scale.
- Rambachan et al. (2024) survey leading general-interest economics journals from 2015–2024 and find that roughly 50% of papers using remote-sensing data follow a *common practice* that proceeds in two steps:
  1. In an auxiliary, observational sample linking the true outcome Y to the RSV R, train a predictor $\hat{Y}(R) \approx \mathbb{E}[Y \mid R]$ using off-the-shelf ML (often a deep neural network on satellite images).
  2. In the experimental sample where Y is unobserved but R is observed, plug $\hat{Y}(R)$ in lieu of Y and estimate the average treatment effect (ATE) as the difference in means of $\hat{Y}(R)$ between treated and control units.
- The paper delivers three messages. First, when the RSV is a *post-outcome* variable (the outcome causes the RSV, not vice versa), the common practice can have large positive *or* negative bias, even with a perfect ML predictor. Second, under a *stability* assumption on the conditional distribution $R \mid (Y, D)$, the ATE is nonparametrically identified by combining the two samples in a principled way. Third, the resulting estimator permits valid $\sqrt{n}$-inference *without* rate conditions on the ML-learned representation of R, justifying the use of complex deep learning algorithms with unknown statistical properties.

1.2 Setup

- Each unit is characterized by the random vector $(S, X, D, Y(0), Y(1), R)$, assumed i.i.d. The sample indicator $S \in \{e, o\}$ distinguishes the *experimental sample* $(S = e)$ from an *observational sample* $(S = o)$.

- Observability of the variables differs across samples (see Table 1):
  - **Experimental sample** ($S = e$): X, D, and R are observed; the outcome Y is missing.
  - **Observational sample, complete case** ($S = o$): X, D, Y, and R are observed. Here D may be subject to unobserved confounding.
  - **Observational sample, incomplete case**: X, Y, and R are observed but D is missing or deterministic (we then encode $D = 0$).
- The causal parameter of interest is the ATE in the experimental sample:

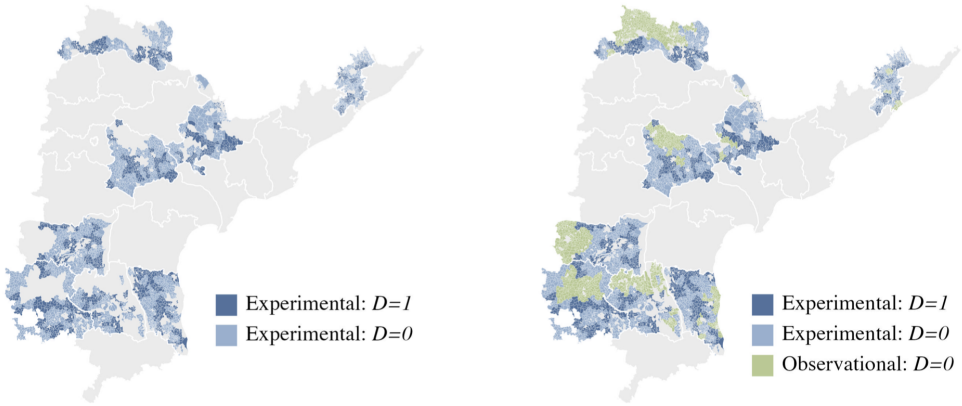
$$\theta := \mu(1) - \mu(0), \qquad \mu(d) := \mathbb{E}[Y(d) \mid S = e].$$

Table 1: Data environment (Table 1 of Rambachan et al., 2024). A check mark indicates the variable is observed in that sample.

Sample S	Covariate X	Treatment D	Outcome Y	RSV R
Experimental ($S = e$)	✓	✓	Missing	✓
Observational, complete ($S = o$)	✓	✓	✓	✓
Observational, incomplete ($S = o$)	✓	Missing/deterministic	✓	✓

Example 1 (Environmental impacts)**.** A randomized experiment offers payments for ecosystem services (PES) to villages. The true outcome $Y \in \{0, 1\}$ indicates whether a plot has *not* been burned (so $Y = 1$ is the environmentally desirable event). Access to PES contracts D is randomized at the village level. The RSV R is a satellite-based classifier constructed from spectral indices; it is not the true outcome itself. Burning causes changes in satellite images, but not vice versa, so R is post-outcome. An observational sample is formed from random spot checks where surveyors record Y for a subset of fields (Jack et al., 2025).

Example 2 (Household poverty)**.** An anti-poverty cash-transfer experiment is evaluated at the village level (Muralidharan et al., 2023). The outcome Y is village-level poverty, which is expensive to survey in person. The RSV R is luminosity or a 4000-dimensional embedding of a satellite image. An observational sample links census-based poverty measures to satellite images at the same geographic coordinates.



(a) Experimental units only.

(b) Auxiliary sample: Observational units.

Figure 1: We illustrate the two samples that we will use to evaluate an anti-poverty program in Andhra Pradesh, India (Muralidharan et al., 2023). With experimental units alone and completely missing outcomes, point identification is impossible. Therefore we introduce an auxiliary sample of observational units. See Section 5 for further details.

Figure 1: Experimental and observational samples in the India anti-poverty application. Source: Rambachan et al. (2024, Figure 1).

1.3 Assumptions

We formalize the causal setting via three assumptions. The first is standard.

Assumption 1 (Experimental unconfoundedness). *(i)* $SUTVA: Y = DY(1)+(1-D)Y(0)$ *a.s.*

(ii) *Randomization:* $D \perp (Y(0), Y(1)) \mid X, S = e$.

(iii) *Overlap:* $\Pr(D = 1 \mid X, S = e)$ *is bounded away from* 0 *and* 1 *a.s.*

The second assumption is the substantive novelty of Rambachan et al. (2024). It formalizes what empirical researchers already implicitly use when they transport an ML prediction from the observational to the experimental sample: the relationship between the RSV R and the outcome Y, conditional on treatment and covariates, is the same in both samples.

Assumption 2 (Stability of the RSV). *(i)* ***Stability****:* $S \perp R \mid X, D, Y$.

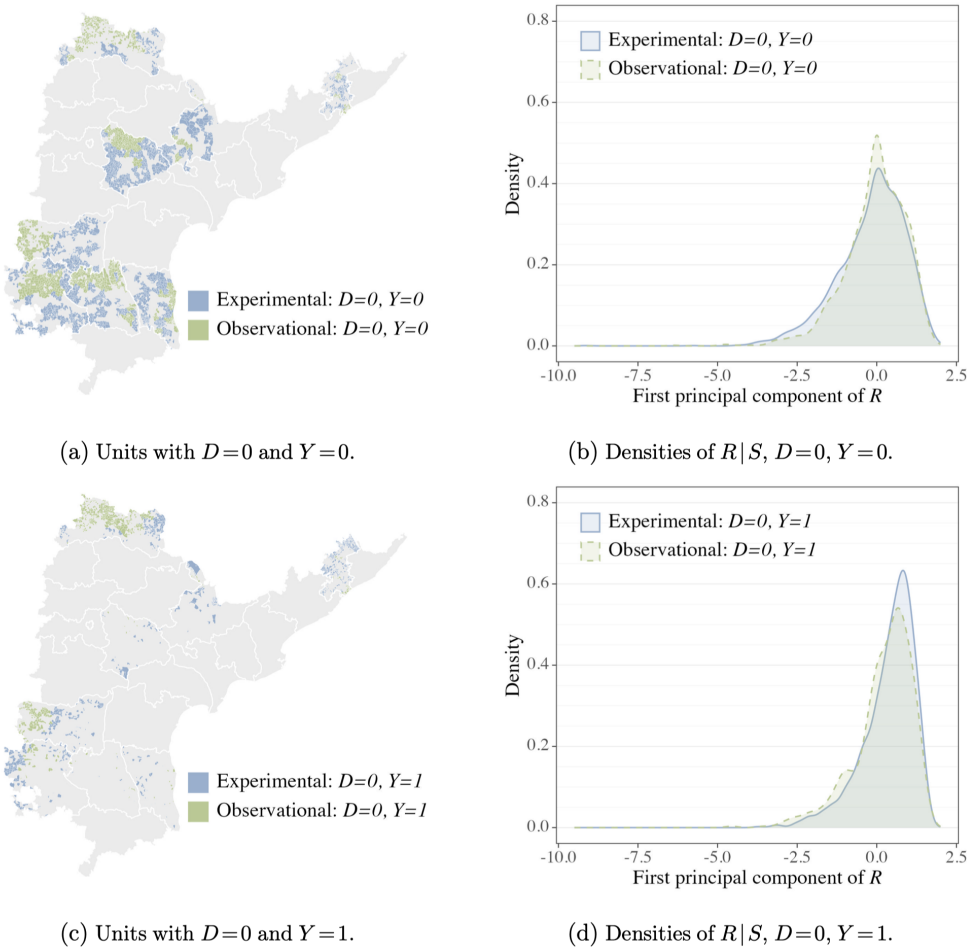
(ii) *Common support for the outcome and nondegenerate coverage of the RSV conditional on* (S, X, D) *(technical regularity).*

(iii) *Two samples:* $\Pr(S = e \mid X)$ *bounded away from* 0 *and* 1.

- Part (i) says the conditional distribution of R given (X, D, Y) is the same in both samples. It permits the "measurement error" distribution of R learned on the observational sample to be transported to the experimental sample. Importantly, the assumption does *not* require the underlying treatment effect distribution to be stable across samples.

- Stability contrasts with the surrogacy framework (Prentice, 1989; Athey et al., 2024), which requires R to fully mediate the effect of D on Y. In that framework, R is *pre-outcome*; here R is *post-outcome*.

- Stability is partially assessable when outcomes are available on both samples: one can compare the empirical distribution of summaries of R within (D, Y) strata across samples, or check whether different representations lead to materially different estimates. Rambachan et al. (2024) provide diagnostic plots for selected strata in their India application.



(a) Units with $D=0$ and $Y=0$.

(b) Densities of $R \mid S$, $D=0$, $Y=0$.

(c) Units with $D=0$ and $Y=1$.

(d) Densities of $R \mid S$, $D=0$, $Y=1$.

Figure 2: Our main assumption (Assumption 2(i)) is plausible in real data. We compare $\Pr(R \mid S=e, D=0, Y=0)$ with $\Pr(R \mid S=o, D=0, Y=0)$ in Figure 2b, and $\Pr(R \mid S=e, D=0, Y=1)$ with $\Pr(R \mid S=o, D=0, Y=1)$ in Figure 2d, using data from Muralidharan et al. (2023) that we analyze in Section 5. Because the satellite image $R \in \mathbb{R}^{4000}$ is high dimensional, we visualize the density of its standardized first principal component on the right hand side, for units highlighted on the left hand side.

Figure 2: Diagnostic for the stability assumption in the India application: densities of the first principal component of the RSV are compared across samples within selected (D, Y) strata. Source: Rambachan et al. (2024, Figure 2).

The paper then distinguishes two routes. With *complete cases*, the observational sample contains treatment variation and direct effects of D on R can be accommodated. With *incomplete* cases, where the observational sample has no useful variation in D, a no-direct-effect restriction is needed:

Assumption 3 (Observational completeness)**.** Either *(i)* complete cases*:* $\Pr(D = 1 \mid S = o, X)$ *is bounded away from* 0 *and* 1*;* or *(ii)* no direct effect*:* $D \perp R \mid X, Y$*.*

Assumption 3(ii) says the treatment D influences the RSV R only through its effect on the outcome Y. In the crop-burning example, this means that PES contracts affect the satellite classifier only because they change the probability of burning, not because they directly alter satellite images via, say, visible investments in farm equipment.

1.4 The Common Practice and Its Bias

- Write $\widetilde{\mu}(d) := \mathbb{E}\{\mathbb{E}[Y \mid R, S = o] \mid D = d, S = e\}$. The common practice estimates this object: the first step learns $\mathbb{E}[Y \mid R, S = o]$ on the observational sample, and the second step averages the predictions on the treated and control arms of the experimental sample. The resulting estimand is
$$\widetilde{\theta} := \widetilde{\mu}(1) - \widetilde{\mu}(0). \tag{1}$$

- The following result shows that $\widetilde{\theta}$ can differ arbitrarily from the true ATE θ, even with a perfect ML predictor.

Proposition 1 (Bias of common practice; Rambachan et al., 2024, Proposition 1)**.** Suppose Assumptions 1, 2, and 3(ii) hold with $X = \emptyset$. Assume further that the observational sample contains no treated units, $\Pr(D = 1 \mid S = o) = 0$, and $S \perp (Y, R) \mid D$. Then

$$\widetilde{\theta} - \theta = \mu(1) \int \{w(r) - 1\} \Pr(R = r \mid Y = 1, S = e)\, dr,$$

with $w(r) = \dfrac{\Pr\{Y(0) = 1 \mid S = e\} \Pr(R = r \mid D = 1)}{\Pr\{Y(1) = 1 \mid S = e\} \Pr(R = r \mid D = 0)}$. Moreover, data-generating processes exist with $\widetilde{\theta} - \theta > 0$ and with $\widetilde{\theta} - \theta < 0$.

- Intuitively, the common practice implicitly uses the RSV as a surrogate *between* treatment and outcome. But when R is post-outcome, the direction of causality is reversed: the outcome affects the RSV, and the common practice's transported predictions $\Pr(Y \mid R, S = o)$ do not correspond to potential outcomes in the experimental sample.

- In the degenerate case where R fails to predict Y (i.e., $\mathbb{E}[Y \mid R, S = o] = \mathbb{E}[Y \mid S = o]$), $\widetilde{\theta} = 0$ regardless of the true ATE: an ideal method would return infinite standard errors, but common practice instead returns a *precise* estimate of zero.

- The bias is present even without ML: it arises from the causal structure of the problem, not from any statistical properties of the prediction algorithm.

1.5 Identification

Under the assumptions above, the ATE is identified by a data-combination formula that contrasts treatment-induced variation of R in the experiment against outcome-induced variation of R in the observational sample.

Theorem 1 (Identification as a conditional moment; Rambachan et al., 2024, Theorem 1)**.** *Suppose the outcome is binary and Assumptions 1, 2, and 3(ii) hold. For any $x \in \mathcal{X}$, the conditional average treatment effect $\theta(x) := \mathbb{E}[Y(1) - Y(0) \mid S = e, X = x]$ satisfies the conditional moment*

$$\mathbb{E}\{\Delta^e(x) - \Delta^o(x)\,\theta(x) \mid X = x, R\} = 0 \quad a.s., \tag{2}$$

where

$$\Delta^e(x) = \frac{\mathbb{1}\{D = 1, S = e\}}{\Pr(D = 1, S = e \mid X = x)} - \frac{\mathbb{1}\{D = 0, S = e\}}{\Pr(D = 0, S = e \mid X = x)},$$
$$\Delta^o(x) = \frac{\mathbb{1}\{Y = 1, S = o\}}{\Pr(Y = 1, S = o \mid X = x)} - \frac{\mathbb{1}\{Y = 0, S = o\}}{\Pr(Y = 0, S = o \mid X = x)}.$$

- The term $\Delta^e(x)$ captures treatment-induced variation from the experimental sample, and $\Delta^o(x)$ captures outcome-induced variation from the observational sample. The conditional moment (2) says that these two variations must be consistent: their projections onto R must agree, weighted by $\theta(x)$.

- Equation (2) fits the classical conditional moment framework of Chamberlain (1987) and Newey and McFadden (1994), which enables standard estimation and inference techniques.

- **Corollary (Identification via representation).** For any representation $H(X, R)$ with $\mathbb{E}\{H \cdot \Delta^o \mid X\} \neq 0$,
$$\theta(x) = \frac{\mathbb{E}\{H(X, R)\,\Delta^e(x) \mid X = x\}}{\mathbb{E}\{H(X, R)\,\Delta^o(x) \mid X = x\}}. \tag{3}$$
Any *relevant* predictive representation identifies $\theta(x)$. The user has substantial freedom in choosing H, but if the denominator is close to zero the RSV is weakly related to outcome variation and standard errors should be large.

1.6 Optimal Representation and Three Predictions

For clarity we focus on the case $X = \emptyset$ with a binary outcome and Assumption 3(ii) in force, following Rambachan et al. (2024, Section 4). Under these restrictions $\theta(x)$ collapses to the scalar ATE θ, and $\Delta^e(x), \Delta^o(x)$ simplify to Δ^e, Δ^o.

- Because (3) holds for any valid representation, there exists an *optimal* representation that minimizes asymptotic variance. Writing the conditional moment as a regression $\Delta^e = \Delta^o \theta + \epsilon$ with $\mathbb{E}(\epsilon \mid R) = 0$, the classical result of Chamberlain (1987) gives

$$H^*(R) = \frac{\mathbb{E}[\Delta^o \mid R]}{\sigma^2(\theta, R)}, \quad \sigma^2(\theta, R) = \mathbb{E}[(\Delta^e - \Delta^o\theta)^2 \mid R]. \tag{4}$$

- Unpacking the numerator and denominator: $\mathbb{E}[\Delta^o \mid R]$ is built from $\Pr(Y = 1 \mid S = o, R)$ and $\Pr(S = o \mid R)$; the denominator involves $\Pr(D = 1 \mid S = e, R)$. Therefore, the semiparametrically efficient use of the RSV requires *three* machine-learning predictions, each fed into H^*:
  - $\text{PRED}_Y(R) \approx \Pr(Y = 1 \mid S = o, R)$: the outcome predictor (this is what common practice already uses).
  - $\text{PRED}_D(R) \approx \Pr(D = 1 \mid S = e, R)$: the treatment predictor.
  - $\text{PRED}_S(R) \approx \Pr(S = e \mid R)$: the sample-indicator predictor.
- Common practice only uses the first prediction. The paper's observation is that the second and third matter for *efficient* use of the RSV, even though identification and valid inference only require a relevant limiting representation. Intuitively, different causal estimands call for different optimal compressions of a satellite image; no single "poverty prediction" is uniformly best.

1.7 Inference Without Rate Conditions

A distinctive feature of the framework is that valid $n^{-1/2}$-inference on θ requires *only* that the learned representation has some probability limit, not that it converges at any particular rate to the optimal H^*. This justifies the use of complex deep learning algorithms with unknown statistical properties.

Assumption 4 (Limit)**.** *The learned representation has some mean-square limit:* $\mathbb{E}\{\widehat{H}(R) - \widetilde{H}(R)\}^2 = o_p(1)$ *for some* $\widetilde{H}$ *with* $\mathbb{E}\{\widetilde{H}(R)^2\} < \infty$*. In addition,* $\widetilde{H}$ *is correlated with outcome variation:* $\mathbb{E}\{\widetilde{H}(R)\Delta^o\}$ *is bounded away from zero.*

- Assumption 4 imposes *no* rate condition and *no* complexity restriction on the ML algorithm used to form $\widehat{H}$. It only requires that the algorithm converges to some limit, which need not equal the optimal H^*.

- The relevance condition $\mathbb{E}\{\widetilde{H}(R)\Delta^o\} \neq 0$ is testable in finite samples: if the empirical analog is statistically indistinguishable from zero, the RSV is effectively a "weak instrument" for the outcome, and standard errors explode. This is a feature, not a bug.

Proposition 2 (Asymptotic normality; Rambachan et al., 2024, Propositions 2–3)**.** Suppose the assumptions of Theorem 1 and Assumption 4 hold, and the marginal probabilities $\Pr(D = d, S = e)$ and $\Pr(Y = y, S = o)$ are known and bounded away from zero. Then the sample-split estimator $\hat{\theta}$ defined below satisfies

$$\sqrt{n}(\hat{\theta} - \theta) \xrightarrow{d} \mathcal{N}\left(0, \frac{\mathbb{E}[(\Delta^e - \theta\Delta^o)^2 \widetilde{H}(R)^2]}{[\mathbb{E}\{\Delta^o \widetilde{H}(R)\}]^2}\right).$$

When $\widetilde{H} = H^*$, the asymptotic variance attains the semiparametric efficiency bound. When the marginal probabilities are estimated by sample counts, the limiting variance includes the additional count-estimation terms characterized in Rambachan et al. (2024, Proposition 3); the paper therefore recommends the bootstrap implementation in Algorithm 1.

Remark 1 (Infinite-order Neyman orthogonality)**.** The moment condition of Theorem 1 enjoys a form of *infinite-order* Neyman orthogonality (Mackey et al., 2018; Chernozhukov et al., 2022). This property is what allows valid inference without rate conditions on the nuisance estimators. It differs from the first-order orthogonality exploited by debiased/double machine learning (see the lecture on debiased inference), which typically requires product rates of the form $n^{-1/4}$ for each nuisance.

1.8 Estimation Algorithm

Algorithm 1: Inference with a learned representation (Algorithm 1 of Rambachan et al., 2024).

Input: $\{(S_i, \mathbb{1}\{S_i = e\}D_i, \mathbb{1}\{S_i = o\}Y_i, R_i)\}_{i=1}^n$ and a split into TRAIN and TEST folds.

Step 1: Learn the representation on train.

(a) Count the marginal counts $\text{COUNT}_{Y=y,S=o}$, $\text{COUNT}_{D=d,S=e}$.

(b) Train the three predictors $\text{PRED}_Y(R)$, $\text{PRED}_D(R)$, $\text{PRED}_S(R)$.

(c) Compute an initial estimate $\hat{\theta}_{\text{init}}$ by regressing $\widehat{\mathbb{E}}[\Delta^e \mid R]$ on $\widehat{\mathbb{E}}[\Delta^o \mid R]$.

(d) Form $\widehat{H}(R) = \widehat{\mathbb{E}}[\Delta^o \mid R]/\widehat{\sigma}^2(\hat{\theta}_{\text{init}}, R)$.

Step 2: Efficient causal estimate on test.

(a) Compute $\widehat{\Delta}^e, \widehat{\Delta}^o$ from the marginal counts.

(b) Form the estimator

$$\hat{\theta} = \frac{\mathbb{E}_{\text{TEST}}\{\widehat{\Delta}^e \widehat{H}(R)\}}{\mathbb{E}_{\text{TEST}}\{\widehat{\Delta}^o \widehat{H}(R)\}}.$$

(c) Bootstrap the confidence interval, fixing $\widehat{H}(R)$.

Output: Estimate $\hat{\theta}$ and $(1-\alpha)$-bootstrap CI.

- The sample split ensures that the learned representation $\widehat{H}$ is independent of the sample used for inference, which is what enables valid standard errors without complexity restrictions. Cross-fitting with any fixed number of folds also works.
- Unlike double machine learning in the lecture on debiased inference, here sample splitting does not serve to weaken rate conditions, because there are none. It simply avoids overfitting of the learned representation on the inference sample.

1.9 Empirical Evidence

- Rambachan et al. (2024) revisit two applications.
  - **Crop burning in India (Jack et al., 2025).** PES contracts are offered at the village level; the true outcome is whether a field has *not* been burned, measured in an auxiliary spot-check sample, while the experimental sample contains a satellite-based classifier as the RSV. The common practice estimates the effect of offering PES at $\widetilde{\theta} = 0.079$. The

RSV is post-outcome (burning causes changes in the satellite image). Applying the correction yields $\hat{\theta} = 0.148$, *nearly double* the common-practice estimate. Common practice underestimates the treatment effect by about 47% due to the bias described in Proposition 1.

– **Smartcards in Andhra Pradesh (Muralidharan et al., 2023).** A biometrically authenticated payments infrastructure is randomized at the village level; the outcome is village-level poverty measured from luminosity and 4000-dimensional satellite embeddings. A semi-synthetic design shows the new estimator matches an unbiased benchmark (difference-in-means on a subset of villages with surveyed outcomes), with comparable confidence intervals. Using RSVs in lieu of surveys could reduce survey costs by an estimated $3 million in this application.

Table 2: Effect of PES contracts on crop burning (Table 2 of Rambachan et al., 2024).

	Common practice $\widetilde{\theta}$	RSV relevance β	Causal parameter θ
Estimate	0.079^{*}	0.530^{***}	0.148^{*}
(SE)	(0.041)	(0.072)	(0.084)

Notes: $\widetilde{\theta}$ is the two-step common-practice estimate. The parameter $\beta := \mathbb{E}[R \mid Y = 1] - \mathbb{E}[R \mid Y = 0]$ measures how strongly the RSV varies with the outcome; in the binary direct-plug-in setting of the crop-burning application, the paper's linear decomposition gives $\widetilde{\theta} = \beta\,\theta$, so $\beta < 1$ implies common practice attenuates θ toward zero. Here $0.530 \times 0.148 \approx 0.078 \approx \widetilde{\theta}$.

1.10 Practical Recommendations

- **Collect an auxiliary sample.** Ideally this sample has linked (Y, R) and observed D; otherwise, stronger assumptions (no direct effect of D on R) are required.
- **Train three predictors.** Predict Y from R using the observational sample; predict D from R using the experimental sample; predict S from R using the pooled sample. Complex ML methods with unknown statistical properties are fine.
- **Run diagnostics.** Before reporting estimates, (a) inspect the density of low-dimensional summaries of R conditional on available (D, Y) strata across the two samples, and compare estimates across alternative representations to assess stability; and (b) test whether the empirical analog of $\mathbb{E}[\widetilde{H}(R)\,\Delta^{o}]$ is significantly nonzero ("RSV relevance").

- **Use the sample-split estimator of Algorithm 1** and a bootstrap confidence interval clustered at the randomization unit. Coverage remains valid as long as the learned representation converges to *any* limit correlated with outcome variation.

2 Inference for Regression with AI/ML-Generated Variables

This section is based on Battaglia et al. (2024).

2.1 Motivation

- A growing body of empirical work estimates a latent quantity θ_i via AI/ML algorithms, then uses $\hat{\theta}_i$ as a regressor (or control variable) in a downstream regression. Examples include:
  - *Label imputation*: $\theta_i \in \{0, 1\}$ (e.g., whether a job posting offers remote work), with $\hat{\theta}_i$ coming from a fine-tuned language model applied to the posting text (e.g., Hansen et al., 2018, 2023).
  - *Dimensionality reduction*: θ_i is a topic weight or embedding from an unsupervised model (e.g., LDA, embedding models). In Bandiera et al. (2020), θ_i is a CEO "behavior index" constructed from time-use surveys.
  - *Index construction*: θ_i is a latent score (e.g., policy uncertainty, hawkish sentiment) constructed by classifying and aggregating text (Baker et al., 2016; Gorodnichenko et al., 2023).
- In each case the researcher runs a regression like $Y_i = \gamma\theta_i + \mathbf{q}_i^\top \boldsymbol{\alpha} + \varepsilon_i$ using $\hat{\theta}_i$ in place of θ_i (the *two-step strategy*), treating $\hat{\theta}_i$ as if it were raw data. Two concerns arise:
  1. **Bias from measurement error.** $\hat{\theta}_i \neq \theta_i$ in general.
  2. **Invalid standard errors** due to the generated-regressor problem (Pagan, 1984).
- Battaglia et al. (2024) show that, under a novel asymptotic framework tailored to the modern setting (large n, high-quality but imperfect ML), the first concern is the binding one: the two-step OLS estimator remains consistent, and OLS standard errors are *consistent*, but the $\sqrt{n}$ distribution has a nonzero location shift. The result differs from the classical Pagan (1984) generated-regressor problem, where the variance is inflated but the location is correct.

- The paper proposes two practical remedies: (i) a bias correction for the two-step estimator; and (ii) joint estimation of the latent variable and the regression parameters via a state-space-type likelihood.

2.2 A Simple Example: Policy Uncertainty

- Inspired by Baker et al. (2016), consider a regression of an outcome Y_i (e.g., next-month investment) on policy uncertainty θ_i:

$$Y_i = \alpha + \gamma\,\theta_i + \varepsilon_i, \qquad \mathbb{E}[\varepsilon_i \mid \theta_i] = 0. \tag{5}$$

- Policy uncertainty is latent. Baker et al. (2016) construct an index from the share of articles in a given month that mention policy uncertainty:

$$X_i \mid (C_i, \theta_i) \sim \text{Binomial}(C_i, \theta_i), \quad \hat{\theta}_i = X_i/C_i, \tag{6}$$

  where C_i is the number of articles sampled in month i.

- Note that $\hat{\theta}_i$ has sampling error of order $C_i^{-1/2}$. The two-step strategy regresses Y_i on $\hat{\theta}_i$ and reads off standard OLS estimates.

2.3 Drifting-Sequence Asymptotics

- To deliver a useful finite-sample approximation in *modern* applications (where ML is highly accurate but the sample is very large), Battaglia et al. (2024) introduce an asymptotic framework in which measurement error and sampling error remain *comparable* as $n \to \infty$.

- In the simple example, this is achieved by letting C_i grow with n so that

$$\sqrt{n}\,\mathbb{E}[C_i^{-1}] \to \kappa \in [0, \infty). \tag{7}$$

  The parameter κ indexes the relative magnitude of measurement error: larger κ means measurement error is more important relative to sampling error.

- This drifting-sequence device is not meant literally; it is a way to obtain a non-degenerate asymptotic distribution in which both measurement error and sampling error play a role, mirroring the finite-sample trade-off researchers actually face.

- This contrasts with classical measurement-error asymptotics, where the variance of the error is held fixed and the two-step estimator is inconsistent ($\sqrt{n}$-biased with no hope of bias

correction); and with the classical generated-regressor problem, where the error shrinks so fast that it inflates variance but does not shift location.

Main result for the simple example. Under regularity, the two-step OLS estimator of γ satisfies

$$\sqrt{n}(\hat{\gamma} - \gamma) \xrightarrow{d} \mathcal{N}\left(-\kappa\,\gamma\,\frac{\mathbb{E}[\theta_i(1-\theta_i)]}{\mathrm{Var}(\theta_i)},\ \frac{\mathbb{E}[\varepsilon_i^2(\theta_i - \mathbb{E}[\theta_i])^2]}{\mathrm{Var}(\theta_i)^2}\right), \tag{8}$$

and OLS standard errors are consistent for the asymptotic variance. Thus the usual 95% CI has correct width but incorrect centering; coverage falls below nominal whenever $\kappa > 0$. The bias is proportional to κ and to γ.

2.4 General Setup and Main Theorem

- The simple example had a scalar latent θ_i and a single regressor. We now generalize to a *vector* of latent variables $\boldsymbol{\theta}_i$ and a vector of observed controls $\mathbf{q}_i$. From here on $\boldsymbol{\theta}_i$ denotes the (possibly vector) latent variable of interest; the scalar θ_i of Section 2.2 is the special case $\dim(\boldsymbol{\theta}_i) = 1$.

- Let the downstream linear regression be

$$Y_i = \boldsymbol{\gamma}^\top \boldsymbol{\theta}_i + \boldsymbol{\alpha}^\top \mathbf{q}_i + \varepsilon_i, \qquad \mathbb{E}[\varepsilon_i \boldsymbol{\xi}_i] = \mathbf{0}, \tag{9}$$

  where $\boldsymbol{\theta}_i$ is a vector of latent variables and $\mathbf{q}_i$ a vector of observed controls. For each i, the researcher has unstructured data $\mathbf{x}_i$ (text, image, time use), from which an estimate $\hat{\boldsymbol{\theta}}_i$ is constructed.

- Let $\boldsymbol{\psi} = (\boldsymbol{\gamma}^\top, \boldsymbol{\alpha}^\top)^\top$, $\boldsymbol{\xi}_i = (\boldsymbol{\theta}_i^\top, \mathbf{q}_i^\top)^\top$, and $\hat{\boldsymbol{\xi}}_i = (\hat{\boldsymbol{\theta}}_i^\top, \mathbf{q}_i^\top)^\top$. The two-step OLS estimator is

$$\hat{\boldsymbol{\psi}} = \left(\frac{1}{n}\sum_{i=1}^{n} \hat{\boldsymbol{\xi}}_i \hat{\boldsymbol{\xi}}_i^\top\right)^{-1} \frac{1}{n}\sum_{i=1}^{n} \hat{\boldsymbol{\xi}}_i Y_i. \tag{10}$$

- The drifting-sequence condition is

$$\frac{1}{\sqrt{n}}\sum_{i=1}^{n} \hat{\boldsymbol{\theta}}_i(\hat{\boldsymbol{\theta}}_i - \boldsymbol{\theta}_i)^\top \xrightarrow{p} \kappa\,\boldsymbol{\Omega} \tag{11}$$

  for some constant $\kappa \geq 0$ and non-random matrix $\boldsymbol{\Omega}$. Here κ captures the magnitude of measurement error relative to sampling error, and $\boldsymbol{\Omega}$ encodes the *structure* of the error. Equation (11) allows non-classical measurement error (error correlated with $\boldsymbol{\theta}_i$), essential for binary labels.

Assumption 5 (High-level conditions)**.** *Standard moment conditions hold; sample second moments of* $\boldsymbol{\xi}_i$ *converge to nonsingular limits;* (11) *holds with some* $(\kappa, \boldsymbol{\Omega})$*;* $n^{-1/2}\sum_i(\hat{\boldsymbol{\theta}}_i - \boldsymbol{\theta}_i)\mathbf{q}_i^\top \xrightarrow{p} \mathbf{0}$*; and* $n^{-1/2}\sum_i \hat{\boldsymbol{\xi}}_i\varepsilon_i \xrightarrow{d} \mathcal{N}(\mathbf{0}, \mathbb{E}[\varepsilon_i^2\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top])$ *with the corresponding residual covariance consistently estimated by the usual OLS formula.*

Theorem 2 (Asymptotic distribution of two-step OLS; Battaglia et al., 2024, Theorem 1)**.** *Under Assumption 5,*

$$\sqrt{n}\big(\hat{\boldsymbol{\psi}} - \boldsymbol{\psi}\big) \xrightarrow{d} \mathcal{N}\left(-\kappa\,\mathbb{E}[\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top]^{-1}\begin{bmatrix}\boldsymbol{\Omega} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\boldsymbol{\psi},\ \ \mathbf{V}\right), \tag{12}$$

where $\mathbf{V} = \mathbb{E}[\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top]^{-1}\mathbb{E}[\varepsilon_i^2\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top]\,\mathbb{E}[\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top]^{-1}$ *is the infeasible-OLS variance. Moreover, the standard OLS variance estimator* $\hat{\mathbf{V}}$ *from regressing* Y_i *on* $\hat{\boldsymbol{\xi}}_i$ *is consistent for* $\mathbf{V}$*.*

- The bias term $-\kappa\,\mathbb{E}[\boldsymbol{\xi}_i\boldsymbol{\xi}_i^\top]^{-1}\mathrm{diag}(\boldsymbol{\Omega}, \mathbf{0})\,\boldsymbol{\psi}$ is a *first-order location shift*: the asymptotic distribution of $\sqrt{n}(\hat{\boldsymbol{\psi}} - \boldsymbol{\psi})$ is not centered at zero when $\kappa > 0$. This is the precise sense in which two-step inference is invalid: the two-step CI has the correct *width* but wrong *centering*.
- The bias can be *positive or negative* depending on the structure of $\boldsymbol{\Omega}$. Ex ante one cannot even know the sign.
- The generated-regressor setup of Pagan (1984) is different: there the $\hat{\theta}_i$ depend on a common finite-dimensional parameter estimated in the first stage, so (11) converges to a *random* variable, inflating variance but not shifting location. Here the $\hat{\theta}_i$ are estimated observation-by-observation from unit-level unstructured data $\mathbf{x}_i$, giving a deterministic limit and a location shift.

2.5 Three Applications with Explicit Bias Expressions

The paper specializes the general result to three running applications.

2.5.1 AI/ML-Generated Labels

- Here $\theta_i \in \{0, 1\}$ is a binary latent label and $\hat{\theta}_i = \pi(\mathbf{x}_i) \in \{0, 1\}$ is the classifier's output. Under mild conditions,

$$\kappa = \lim_{n\to\infty}\sqrt{n}\,\mathbb{E}[\hat{\theta}_i(1 - \theta_i)], \qquad \boldsymbol{\Omega} = 1. \tag{13}$$

  The quantity $\mathbb{E}[\hat{\theta}_i(1 - \theta_i)] = \Pr(\hat{\theta}_i = 1, \theta_i = 0)$ is the paper's *unconditional* false-positive probability. It is called FPR in the paper, but it differs from the conventional conditional false-positive rate $\Pr(\hat{\theta}_i = 1 \mid \theta_i = 0)$.

- Thus the bias is driven by $\sqrt{n} \times$ FPR. In modern applications with large n, even a small unconditional false-positive probability can produce large first-order bias. In the Lightcast remote-work example, $n = 16{,}315$ and FPR ≈ 0.009 give $\kappa \approx 1.1$, large enough that the two-step CI under-covers substantially.

2.5.2 Topic Models

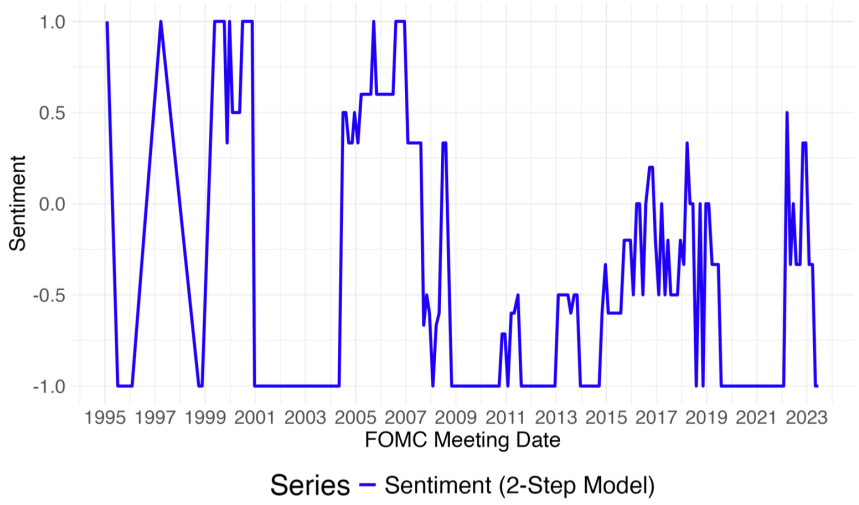
- Here $\mathbf{x}_i$ is a V-dimensional feature count ($V =$ vocabulary size), and $\boldsymbol{\theta}_i \in \Delta^{K-1}$ is a vector of topic weights. Under an LDA-type generative model $\mathbf{x}_i \mid (C_i, \boldsymbol{w}_i) \sim \text{Multinomial}(C_i, \mathbf{B}^\top \boldsymbol{w}_i)$ and provided the topic-loading estimator is accurate enough, in particular $\sqrt{n}(\hat{\mathbf{B}} - \mathbf{B}) \xrightarrow{p} 0$,

$$\kappa = \lim_{n\to\infty} \sqrt{n}\, \mathbb{E}[C_i^{-1}], \quad \boldsymbol{\Omega} = \boldsymbol{S}(\mathbf{B}\mathbf{B}^\top)^{-1}\mathbf{B}\, \text{diag}(\mathbf{B}^\top \mathbb{E}[\boldsymbol{w}_i])\, \mathbf{B}^\top (\mathbf{B}\mathbf{B}^\top)^{-1}\boldsymbol{S}^\top - \mathbb{E}[\boldsymbol{\theta}_i \boldsymbol{\theta}_i^\top]. \quad (14)$$

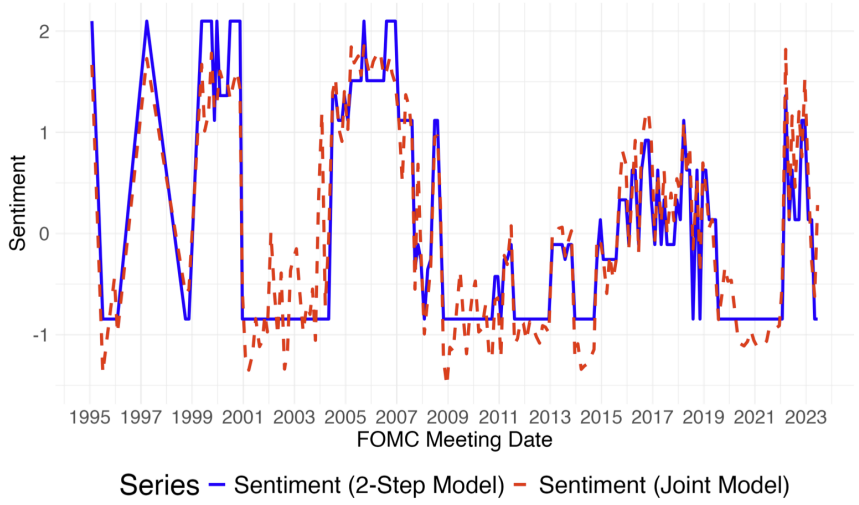
- Intuitively, documents with more features (larger C_i) give more accurate topic weights; sample-size-averaged $1/C_i$ controls the bias.

2.5.3 AI/ML-Generated Indices

- For index construction (e.g., EPU (Baker et al., 2016), hawkishness (Gorodnichenko et al., 2023)), the structure combines classification error and aggregation. The paper does not develop a closed-form bias correction for the central-bank index application; instead, it demonstrates that joint estimation works well (see below).



(a) Unscaled Series



(b) Scaled Series

Note: The left panel plots two-step sentiment $\hat{\theta}_i$. The right panel plots standardized series for $\hat{\theta}_i$ and estimated sentiment from the joint model. The sample period covers 200 FOMC meetings from Feb 1995 through June 2023.

Figure 1: Time Series of FOMC Statement Sentiment

Figure 3: FOMC hawkish sentiment: two-step vs. joint estimation. Source: Battaglia et al. (2024, Figure 1).

2.6 Bias Correction

- Given consistent estimators $\hat{\kappa}$ and $\hat{\mathbf{\Omega}}$ of κ and $\mathbf{\Omega}$, two bias-corrected estimators are available:

$$\hat{\boldsymbol{\psi}}^{bca} = \left(\mathbf{I} + \frac{\hat{\kappa}}{\sqrt{n}} \left(\frac{1}{n} \sum_i \hat{\boldsymbol{\xi}}_i \hat{\boldsymbol{\xi}}_i^\top \right)^{-1} \begin{bmatrix} \hat{\mathbf{\Omega}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \right) \hat{\boldsymbol{\psi}}, \tag{15}$$

$$\hat{\boldsymbol{\psi}}^{bcm} = \left(\mathbf{I} - \frac{\hat{\kappa}}{\sqrt{n}} \left(\frac{1}{n} \sum_i \hat{\boldsymbol{\xi}}_i \hat{\boldsymbol{\xi}}_i^\top \right)^{-1} \begin{bmatrix} \hat{\mathbf{\Omega}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \right)^{-1} \hat{\boldsymbol{\psi}}. \tag{16}$$

  The "additive" estimator $\hat{\boldsymbol{\psi}}^{bca}$ subtracts the estimated bias; the "multiplicative" estimator $\hat{\boldsymbol{\psi}}^{bcm}$ inverts the contraction that maps $\boldsymbol{\psi}$ to its biased limit.

- The $1-a$ bias-corrected confidence interval for the j-th entry of $\boldsymbol{\psi}$ is centered at the bias-corrected estimator. Asymptotically it uses the same variance $\mathbf{V}$ as infeasible OLS, but in finite samples the paper recommends adjusted covariance formulas when $(\kappa, \mathbf{\Omega})$ are estimated from an audit sample.

- The multiplicative correction is more aggressive and is useful when the bias is large and the required inverse is well behaved. Simulations show the additive correction can be inadequate when FPR or κ is large, while multiplicative correction can over-correct if the approximation is poor.

Theorem 3 (Validity of bias correction; Battaglia et al., 2024, Theorem 4)**.** *Under Assumption 5 and consistency of* $(\hat{\kappa}, \hat{\mathbf{\Omega}})$*, the bias-corrected estimators satisfy*

$$\sqrt{n}(\hat{\boldsymbol{\psi}}^{bcm} - \boldsymbol{\psi}) = \sqrt{n}(\hat{\boldsymbol{\psi}}^{bca} - \boldsymbol{\psi}) + o_p(1) \xrightarrow{d} \mathcal{N}(\mathbf{0}, \mathbf{V}),$$

and the bias-corrected CIs have correct asymptotic coverage.

Estimating κ for AI/ML labels without a full validation sample. A key practical challenge is estimating the unconditional FPR without labeling the entire dataset. The authors propose drawing a random sub-sample of size $m \ll n$; for each sampled observation i with $\hat{\theta}_i = 1$ (only), the analyst inspects $\mathbf{x}_i$ and assigns a true label θ_i. Because $\hat{\theta}_i(1-\theta_i) = 0$ whenever $\hat{\theta}_i = 0$, no label is needed for observations with $\hat{\theta}_i = 0$, and the estimator

$$\widehat{\mathrm{FPR}} = \frac{1}{m} \sum_{i=1}^{m} \hat{\theta}_i (1 - \theta_i), \qquad \hat{\kappa} = \sqrt{n}\, \widehat{\mathrm{FPR}} \tag{17}$$

is still well defined. Battaglia et al. (2024) show that $\hat{\kappa}$ is consistent whenever $n/m^2 \to 0$, which accommodates a validation sub-sample much smaller than n. In modern applications where n is in the millions, this drastically reduces labeling cost.

2.7 Joint Estimation

- When the bias correction is hard to derive in closed form (e.g., for generated indices), Battaglia et al. (2024) propose an alternative: jointly estimate the downstream regression and the latent-variable model by maximum likelihood / Bayesian computation.
- In the simple example (5)–(6), the joint likelihood (after integrating out θ_i) is

$$f(Y_i, X_i \mid C_i; \gamma, \alpha, \sigma) = \int_0^1 \frac{1}{\sigma} f\left(\frac{Y_i - \alpha - \gamma\theta_i}{\sigma}\right) \binom{C_i}{X_i} \theta_i^{X_i} (1-\theta_i)^{C_i - X_i} \, g(\theta_i) \, d\theta_i,$$

  where g is a prior for θ_i (e.g., $U[0,1]$). Maximum likelihood (or posterior sampling via Hamiltonian Monte Carlo) on $\prod_i f(Y_i, X_i \mid C_i; \cdot)$ delivers valid frequentist inference on (γ, α, σ) under standard regularity.
- In complex applications, analytic integration is intractable and probabilistic programming (e.g., NumPyro, Stan) is used. The paper's FOMC central-bank communication application uses Hamiltonian Monte Carlo to jointly model aggregated classification counts, classifier probabilities, and the downstream regression. Under this approach, the effect of hawkish sentiment on the path factor nearly triples compared to the two-step estimate (Table 3).

Table 3: Impact of FOMC statement sentiment on longer-term yields (Table 3 of Battaglia et al., 2024).

	Two-Step	Joint
Sentiment (θ_i)	0.038	0.114
	[0.005, 0.071]	[0.027, 0.198]
Policy Rate ($\mathbf{q}_i$)	−0.004	−0.003
	[−0.013, 0.004]	[−0.011, 0.004]
R^2	0.0425	0.1429

2.8 Empirical Evidence

- **Remote work and wage inequality (Hansen et al., 2023).** A fine-tuned language model classifies each of 16,315 San Diego food-service job postings as offering remote work

or not, with test-set accuracy of 99%. The two-step coefficient on remote work in a log-wage regression is 0.36; bias-corrected, it rises to 0.64 (76% larger). The unconditional FPR is only 0.9%, estimated from an audit of $m = 1000$ postings, with 26 predicted positives inspected and 9 false positives. Even so, n is large enough that bias correction matters substantially. See Table 4.

- **CEO behavior and firm performance (Bandiera et al., 2020).** Topic-model estimates of a leadership behavior index are used as a regressor for log sales. In the full sample (916 CEOs), bias correction and joint estimation give similar results to two-step, indicating low κ (about 0.44 in the paper's empirical diagnostic). When the analysis is re-done with a 10% time-use sub-sample (larger κ), two-step becomes insignificant while bias correction and joint estimation retain significance.

- **Central bank communication (Gorodnichenko et al., 2023).** See Table 3 above.

Table 4: Remote work on posted wages (adapted from Table 1 of Battaglia et al., 2024). 95% CIs in brackets.

Controls	Two-Step	Bias Correction	Joint
None	0.648	1.052	0.563
	[0.600, 0.697]	[0.778, 1.327]	[0.532, 0.595]
SOC2 FEs	0.364	0.641	0.448
	[0.322, 0.406]	[0.446, 0.836]	[0.415, 0.480]

2.9 Practical Recommendations

- **Diagnose κ.** Whether naive two-step inference is valid depends on the relative magnitude of measurement error and sampling error. For labels, $\kappa \approx \sqrt{n} \times \text{FPR}$, where FPR is the unconditional false-positive probability: even a small value causes problems in large samples. For topic models, $\kappa \approx \sqrt{n} \times \mathbb{E}[C_i^{-1}]$.

- **Report both bias-corrected estimators** (15)–(16) together with the two-step estimate, but check the invertibility/eigenvalue condition behind the multiplicative correction and use the finite-sample adjusted covariance formulas when the audit sample is small.

- **Use a small audited sub-sample** to estimate the unconditional FPR; (17) only requires inspecting observations classified as $\hat{\theta}_i = 1$.

- **Joint estimation** à la Section 2.7 is valuable when closed-form bias expressions are hard to derive or when κ is large. The Python package `ValidMLInference` implements the paper's bias corrections and standard-error formulas; probabilistic programming frameworks (Stan, NumPyro) implement joint estimation.

3 Large Language Models as a General Empirical Tool

This section is based on Ludwig et al. (2024).

3.1 Motivation

- Large language models (LLMs) make text-based empirical analysis tractable at unprecedented scale, including predicting stock returns from earnings calls, measuring partisanship in social media, backcasting historical sentiment from newspapers, or simulating survey responses cheaply. The temptation is simply to "plug in" LLM outputs as if they were data.
- Unlike the algorithms studied in Sections 1 and 2, LLMs resist traditional statistical analysis: they are complex, often proprietary, constantly evolving, and trained on sprawling corpora that defy tractable modeling. Ludwig et al. (2024) therefore adopt a *black-box* approach, treating LLMs only through their input-output behavior, and identify high-level conditions under which LLM outputs can be incorporated validly into empirical research.
- They organize the discussion around two distinct uses:
  1. **Prediction problems**: predict an economic outcome Y_r (e.g., stock returns) from a text piece r (e.g., a news headline) by computing $\widehat{m}(r;t)$.
  2. **Estimation problems**: measure an economic concept V_r expressed in r (e.g., the policy topic of a Congressional bill, or the sentiment of a headline) and use it as a variable in a downstream regression.
- The paper delivers two main messages, one for each use:
  - For prediction, valid out-of-sample evaluation requires *no training leakage* between the LLM's training corpus and the researcher's evaluation sample. This is achievable through deliberate choices of model and research design.
  - For estimation, plugging in LLM outputs is generally *not* valid: LLM measurement errors can correlate with covariates in unpredictable ways, and seemingly innocuous prompt or model changes can flip the sign of downstream estimates. The fix is a small

validation sample on which the researcher applies the existing measurement procedure $f^*(\cdot)$, and uses it to debias the plug-in estimate. This approach has a long history in econometrics (Chen et al., 2005, 2008).

- The estimation result connects conceptually to Section 2: both are generated-variable problems in which small errors can matter for downstream regression. LMR estimate the bias directly from validation data, while Battaglia et al. (2024) use a drifting-sequence approximation and structural estimates of $(\kappa, \mathbf{\Omega})$.

3.2 Setup and Notation

- Let $\mathcal{R} \subseteq \Sigma^*$ denote the universe of *economically relevant text pieces* r (e.g., bill descriptions, news headlines). Each r is linked to observable economic variables (Y_r, W_r), where Y_r is an outcome and W_r a vector of linked covariates.

- There is an idealized but costly *existing measurement procedure* $f^*(\cdot)$ that produces a label $V_r := f^*(r)$ for each text piece. Typical examples include domain experts carefully reading each text or trained annotators labeling Congressional bills' policy topics. The text-processing problem is that f^* is too expensive to scale.

- The *researcher's dataset* is summarized by sampling indicators $\{D_r\}_{r\in\mathcal{R}}$, where $D_r = 1$ if and only if the researcher collected r. Let $N = \sum_r D_r$.

- The *LLM* is modeled as a deterministic mapping $\widehat{m}(\cdot; t) : \Sigma^* \to \Sigma^*$ obtained by training on a dataset summarized by sampling indicators t. The researcher prompts $\widehat{m}$ with each r and obtains either a prediction $\widehat{Y}_r = \widehat{m}(r; t)$ (prediction problem) or a label $\widehat{V}_r = \widehat{m}(r; t)$ (estimation problem). In estimation problems we write the measurement error as $\Delta_r := \widehat{V}_r - V_r$; prediction problems are evaluated through a loss $\ell(Y_r, \widehat{m}(r; t))$ instead.

- Two layers of uncertainty are formalized:

  - The *research context* $Q(\cdot) \in \mathcal{Q}$ is a joint distribution over (D, T), encoding both the target population for prediction (through D) and the researcher's uncertainty about the LLM's training data (through T). Let $q_r^D = Q(D_r = 1)$, $q_r^T(t_r) = Q(T_r = t_r)$, and $q_r^{T|D}(t_r) = Q(T_r = t_r \mid D_r = 1)$.
  - The *LLM guarantee* $\mathcal{M}$ is a collection of text generators consistent with what the researcher knows (e.g., benchmark performance). The researcher only knows that $\widehat{m}(\cdot; t) \in \mathcal{M}$.

Remark 2 (Notation reconciliation with Sections 1–2)**.** The notation here differs from earlier sections in three ways. First, D_r is a *sampling* indicator, not the binary treatment $D \in \{0,1\}$ of Section 1. Second, the latent label V_r plays the role of $\boldsymbol{\theta}_i$ in Battaglia et al. (2024) (Section 2); the LLM output $\widehat{V}_r = \widehat{m}(r;t)$ corresponds to $\hat{\boldsymbol{\theta}}_i$. Third, W_r collects the linked covariates and outcomes, playing the role of $\mathbf{q}_i$ (or, in regressions where V_r is the dependent variable, of Y_i) in Section 2. The downstream regression coefficient is denoted β throughout this section.

3.3 Prediction with LLMs: No Training Leakage

- In a prediction problem, the researcher computes the sample average loss

$$\widehat{L} := \frac{1}{N} \sum_{r \in \mathcal{R}} D_r \, \ell(Y_r, \widehat{m}(r;t)) \,, \tag{18}$$

  for some loss $\ell(\cdot,\cdot)$, and would like $\widehat{L}$ to reflect predictive performance on the target population, $\mathbb{E}_Q[\sum_r D_r \, \ell(Y_r, \widehat{m}(r;t))]$.

- Two technical conditions are useful: across strings, (D_r, T_r) are independent (but not identically distributed), and the researcher's expected sample size does not depend on the LLM's training corpus. These are stated as Assumption 1 of Ludwig et al. (2024).

Definition 1 (Generalization in a research context; Ludwig et al., 2024, Definition 1)**.** The LLM $\widehat{m}(\cdot;t)$ with guarantee $\mathcal{M}$ *generalizes* in research context $Q(\cdot)$ if, for every $\widehat{m} \in \mathcal{M}$,

$$\mathbb{E}_Q\left[\sum_{r \in \mathcal{R}} D_r \, \ell(\widehat{m}(r), Y_r) \;\middle|\; T = t\right] = \mathbb{E}_Q\left[\sum_{r \in \mathcal{R}} D_r \, \ell(\widehat{m}(r), Y_r)\right].$$

Up to the normalization by expected sample size in the paper, the expectation conditional on $T = t$ matches the average sample loss in (18); the unconditional version is the target. They agree if and only if there is *no training leakage*.

Proposition 3 (No training leakage; Ludwig et al., 2024, Lemma 1, Proposition 1)**.** Under Assumption 1 of Ludwig et al. (2024), the LLM $\widehat{m}(\cdot;t)$ generalizes in research context Q if and only if

$$\mathbb{E}_Q\left[\sum_{r \in \mathcal{R}} D_r \left(\frac{q_r^{T|D}(t_r)}{q_r^{T}(t_r)} - 1\right) \ell(Y_r, \widehat{m}(r))\right] = 0. \tag{19}$$

- The expression $q_r^{T|D}(t_r)/q_r^{T}(t_r) - 1$ quantifies how observing that the researcher sampled text piece r changes the perceived likelihood that r appeared in the LLM's training set. *No training leakage* occurs if this term is either always zero or uncorrelated with the LLM's

loss. In other words, the leakage condition parallels omitted variable bias: if the researcher's sampled text pieces are more likely to have been included in the LLM's training data *and* the LLM performs better on those pieces, the sample average loss will underestimate the true loss and falsely suggest higher predictive performance.

- **Empirical evidence of leakage**. Ludwig et al. (2024) document substantial leakage in two economically relevant datasets:
  - *Congressional bills.* GPT-4o predicts whether a bill passes the U.S. House (Senate) with 91.2% (92.5%) accuracy from the bill's description alone, a striking number given that only 6–7% of bills pass. When prompted to complete bills' descriptions from the first half, GPT-4o reproduces 344 out of 10,000 bill descriptions *verbatim*, indicating direct memorization.
  - *Financial news headlines.* On a dataset of 4 million headlines covering 6,000 publicly traded companies, GPT-4o reproduces 60 out of 10,000 sampled 2019 headlines exactly. Sarkar and Vafa (2024) document additional "lookahead bias": prompting Llama 2 to forecast risks from September–November 2019 earnings calls, the model mentions Covid-19 in over 25% of cases.
- Prompt-engineering fixes such as "ignore information after date τ" do *not* reliably eliminate leakage in either application.

Practical recipes for ruling out leakage by design. Ludwig et al. (2024, Section 3.3) highlight several common scenarios:

- *Time-stamped models.* Use open-source LLMs with fixed published weights (e.g., the Llama family) and construct evaluation samples consisting only of documents published after the model's training cutoff. Then $q_r^T(t_r) = 0$ mechanically.
- *Random sampling from a known corpus.* If the evaluation sample is drawn at random from a well-defined corpus, then $q_r^{T|D}(t_r) = q_r^T(t_r)$, mirroring how randomization eliminates omitted-variable bias in causal inference.
- *Confidential documents* (e.g., administrative case notes) are mechanically out of sample.
- Closed proprietary models (GPT, Claude) are problematic because their training data is undisclosed and may be continuously updated.

3.4 Estimation with LLMs: Measurement Error and Sensitivity

In an estimation problem, the researcher specifies a parameter $\theta \in \Theta$ identified by a moment condition $g(\cdot)$. If V_r were observed for all sampled units, she would compute the *target* estimator

$$\widehat{\theta}^* = \arg\min_{\theta\in\Theta} \frac{1}{N} \sum_{r\in\mathcal{R}} D_r \, g(V_r, W_r; \theta), \tag{20}$$

e.g., for $g(V_r, W_r; \beta) = (V_r - W_r'\beta)^2$ this is the OLS regression of V_r on W_r. Replacing V_r with the LLM label $\widehat{V}_r = \widehat{m}(r; t)$ yields the *plug-in* estimator

$$\widehat{\theta} = \arg\min_{\theta\in\Theta} \frac{1}{N} \sum_{r\in\mathcal{R}} D_r \, g(\widehat{m}(r; t), W_r; \theta). \tag{21}$$

- Conditional on the LLM's training data, the difference between the plug-in and target moment conditions decomposes into two terms (Ludwig et al. 2024, Lemma 2): a *measurement-error* term $\mathbb{E}_Q[\sum_r D_r\{g(\widehat{m}(r), W_r; \theta) - g(V_r, W_r; \theta)\}]$ that arises because $\widehat{m}(r; t) \neq V_r$ in general, and a *leakage* term identical to that of Proposition 3. As discussed in Section 3.3, the leakage term is mechanically zero when the researcher randomly samples text pieces or sees the entire population. We therefore focus on the measurement-error term.

- A natural hope is that an LLM accurate "everywhere" yields a small bias. The next result shows this hope is misplaced.

Proposition 4 (Pointwise accuracy is insufficient; Ludwig et al., 2024, Lemma 3, Proposition 2)**.** Suppose the LLM satisfies the guarantee $\mathcal{M}(Q, \delta) := \{\widehat{m} : \|\widehat{m}(\cdot) - f^*(\cdot)\|_{\infty,Q} \leq \delta\}$ and no training leakage. For any sensitive moment condition $g(\cdot)$ and any θ,

$$\left|\mathbb{E}_Q\left[\textstyle\sum_r D_r\{g(\widehat{m}(r), W_r; \theta) - g(V_r, W_r; \theta)\}\right]\right| \;\leq\; G\delta,$$

but there exist text generators $\widehat{m} \in \mathcal{M}(Q, \delta)$ that achieve nearly this upper bound. Consequently, $\widehat{m}(\cdot; t)$ is a *general-purpose technology for estimation* (the plug-in $\widehat{\theta}$ recovers the target estimand $\widehat{\theta}^*$ for all moment conditions $g \in \mathcal{G}$ and contexts $Q \in \mathcal{Q}$) if and only if $\widehat{m} \in \mathcal{M}(Q, 0)$, i.e., $\widehat{m}(r; t) = f^*(r)$ for all r.

- The lesson is that bounded errors δ provide only a *worst-case* bias bound; what matters for estimation is the precise pattern of errors, especially their correlation with W_r. The condition $\widehat{m} = f^*$ *everywhere* is essentially unverifiable in modern applications, given the brittleness evidence in Section 2.3 of the paper (e.g., LLM performance is sensitive to prompt phrasing, ordering of multiple-choice options, and seemingly trivial reformulations).

Linear-regression specialization. The connection to Battaglia et al. (2024) is most explicit in the linear case. Consider the regression

$$V_r = W_r'\beta^* + \varepsilon_r, \qquad \mathbb{E}_Q[\varepsilon_r W_r] = 0, \tag{22}$$

and the plug-in version $\widehat{V}_r = W_r'\beta + \widetilde{\varepsilon}_r$.

Proposition 5 (Bias of plug-in linear regression; Ludwig et al., 2024, Proposition 3, after Bound et al., 1994)**.** Suppose no training leakage holds. The probability limit β of the plug-in OLS coefficient in (22) satisfies

$$\beta \;=\; \beta^* + \lambda_{\Delta|W},$$

where $\lambda_{\Delta|W}$ is the population OLS coefficient from regressing the LLM error $\Delta_r = \widehat{m}(r;t) - V_r$ on W_r. When the latent variable instead enters as a regressor (i.e., $W_r = V_r'\alpha^* + \nu_r$ with $\widehat{V}_r$ plugged in), the limit takes the form $\alpha = \lambda_{V|\widehat{V}}\,\alpha^* + \lambda_{\eta|\widehat{V}}$, an attenuation-plus-correlation analogue of classical errors-in-variables.

- Proposition 5 is the classical measurement-error bias formula of Bound et al. (1994) applied to LLM-generated variables. It is closely related to Theorem 2, but it is not the same drifting-sequence result: here the bias is the population projection of LLM error on the regression variables. In particular, the bias depends on whether errors Δ_r correlate with W_r, not on their pointwise size.

- **Empirical evidence on prompt and model sensitivity.** Ludwig et al. (2024, Section 4.2.1) document large dispersion across models and prompts. In the financial-headlines application they use five LLMs and nine prompts; in the Congressional-bills application they use a related but not identical set of models and prompt strategies. The plug-in regression coefficient $\widehat{\beta}^{m,p}$ varies dramatically:

  - *Financial news headlines.* Regressing 1-, 5-, and 10-day realized returns on positive/negative sentiment labels, model-assessed magnitude, interactions, and lagged returns, the coefficient on LLM sentiment flips sign, magnitude, and significance across (m, p) pairs (Figure 4).
  - *Congressional bills.* Regressing the labeled policy topic on covariates such as the sponsor's party affiliation reveals similarly large variation across prompts and models (Figure 5).

  Because each (m, p) pair targets a slightly different population object, this specification dispersion is a sharp form of p-hacking risk.

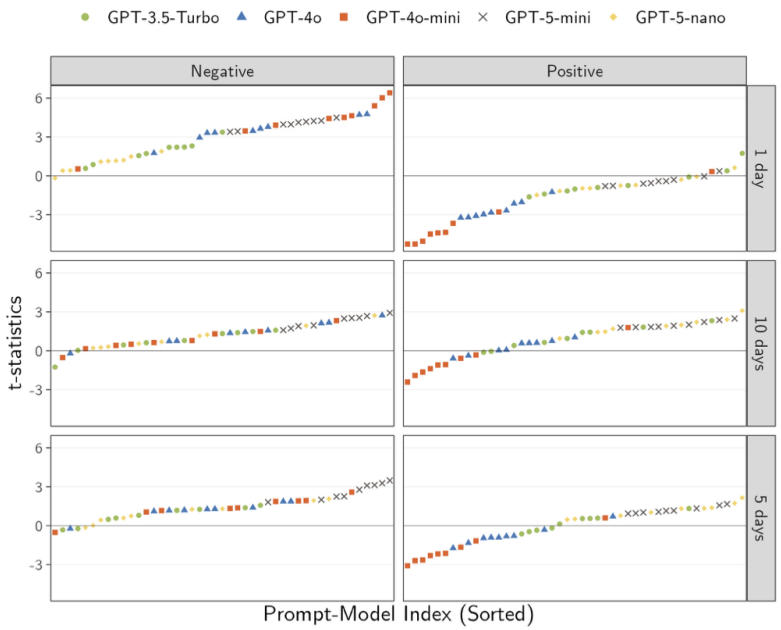


Figure 1: Variation in t-statistics for realized returns across large language models and prompting strategies on financial news headlines.

Notes: On financial news headlines from 2019, we prompt GPT-3.5-Turbo, GPT-4o-mini, GPT-4o, GPT-5-mini, and GPT-5-nano to label each headline for whether it expressed positive, negative or uncertain news about the respective company using alternative prompting strategies. For each model m and prompt p, we regress the realized returns of each stock within 1 day, 5 days or 10 days of the headline's publication date on each large language model's labels $\widehat{V}_r^{m,p}$, the large language model's assessed magnitude denoted $S_r^{m,p}$ and their interaction, controlling for lagged realized returns. We separately report the t-statistics associated with the regression coefficients on whether the headline is labeled as positive or negative news (standard errors are two-way clustered at the date and company level). In each subplot, the t-statistics are sorted in ascending order for clarity. See Section 4.2.1 for discussion.

Figure 4: Variation in plug-in regression coefficients of realized stock returns on positive/negative LLM-generated sentiment labels across LLMs and prompts. Source: Ludwig et al. (2024, Figure 1).

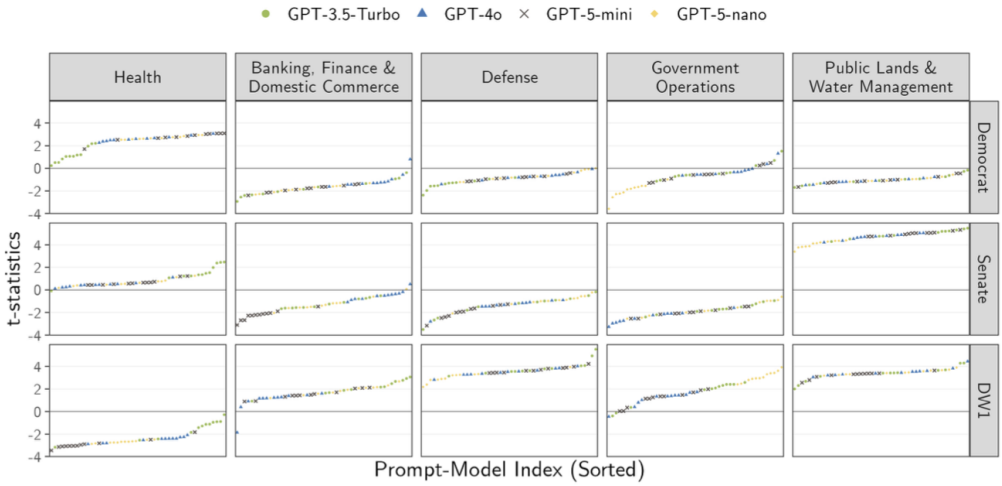


Figure 2: Variation in t-statistics across large language models and prompting strategies on congressional legislation.

Notes: On 10,000 Congressional bills, we prompt GPT-3.5-Turbo, GPT-4o, GPT-5-mini, and GPT-5-nano to label each description for its policy topic area using alternative prompting strategies. For each model m and prompt p, we regress $\widehat{V}_r^{m,p}$ on the linked covariate W_r, where $\widehat{V}_r^{m,p}$ are indicators for the policy topic of the bill and the covariates W_r are whether the bill's sponsor was a Democrat, whether the bill originated in the Senate, and the DW1 score of the bill's sponsor. In each subplot, the t-statistics were sorted in ascending order for clarity. See Section 4.2.1 for discussion.

Figure 5: Variation in plug-in regression coefficients of LLM-labeled Congressional bill topics on linked covariates across LLMs and prompts. Source: Ludwig et al. (2024, Figure 2).

3.5 Validation-Sample Debiasing

The proposed solution is to apply the existing measurement procedure $f^*(\cdot)$ on a small randomly drawn subset of text pieces and use it to debias the plug-in estimate. We illustrate the mechanics for the linear regression with the LLM label as the dependent variable, following Ludwig et al. (2024, Section 4.3 and Appendix C.2).

- For this subsection, recode the sampling indicator as $D_r \in \{0, 1, 2\}$, where $D_r = 0$ means unsampled, $D_r = 1$ means the primary sample, and $D_r = 2$ means the validation sample. The researcher observes only $(\widehat{V}_r, W_r)$ in the primary sample (size N_p, fraction ρ_p), and additionally applies f^* in the validation sample (size N_v, fraction ρ_v) to obtain V_r and the error $\Delta_r = \widehat{V}_r - V_r$. Sampling is uniform random, so $\rho_p + \rho_v$ is the overall sampling rate.

- The plug-in OLS on the primary sample is

$$\widehat{\beta} = \left(\sum_{r:D_r=1} W_r W_r' \right)^{-1} \sum_{r:D_r=1} W_r \widehat{V}_r,$$

and the bias estimate from regressing Δ_r on W_r in the validation sample is

$$\widehat{\lambda}_{\Delta|W} = \left(\sum_{r:D_r=2} W_r W_r' \right)^{-1} \sum_{r:D_r=2} W_r \Delta_r.$$

The *debiased* estimator is simply

$$\widehat{\beta}^{\text{debiased}} = \widehat{\beta} - \widehat{\lambda}_{\Delta|W}. \tag{23}$$

Inference can be obtained by bootstrapping the primary and validation samples jointly.

Theorem 4 (Asymptotic distribution of $\widehat{\beta}^{\text{debiased}}$; Ludwig et al., 2024, Appendix C.2)**.** *Assume scalar W_r, no training leakage, bounded $W_r, V_r, \widehat{m}(r;t)$, and finite-population uniform random sampling into the three groups $D_r \in \{0,1,2\}$ with positive limiting fractions (ρ_p, ρ_v). Let $|\mathcal{R}| \to \infty$. Then*

$$\sqrt{|\mathcal{R}|}\left(\widehat{\beta}^{debiased} - \beta^*\right) \xrightarrow{d} \mathcal{N}\left(0,\ \Omega^{debiased}\right),$$

with

$$\Omega^{debiased} = \sigma_W^{-4}\left(\frac{1-\rho_p}{\rho_p}\, \sigma^2_{\widehat{V}W} + 2\, \sigma_{\widehat{V}W}\, \sigma_{\Delta W} + \frac{1-\rho_v}{\rho_v}\, \sigma^2_{\Delta W} \right), \tag{24}$$

where σ^2_W is the variance of W_r in the universe $\mathcal{R}$, and $\sigma^2_{\widehat{V}W}, \sigma^2_{\Delta W}$ are the variances of $W_r\widehat{V}_r$ and $W_r\Delta_r$, respectively. The validation-only OLS estimator $\widehat{\beta}^{validation}$ (using V_r on the validation subsample) has limiting variance $\Omega^{validation} = \sigma_W^{-4}\, \frac{1-\rho_v}{\rho_v}\, \sigma^2_{VW}$.

- Comparing (24) with $\Omega^{\text{validation}}$ shows that

$$\Omega^{\text{debiased}} \le \Omega^{\text{validation}} \quad \Longleftrightarrow \quad \frac{1-\rho_p}{\rho_p}\, \sigma^2_{\widehat{V}W} + 2\, \sigma_{\widehat{V}W}\, \sigma_{\Delta W} \le \frac{1-\rho_v}{\rho_v}\left(\sigma^2_{VW} - \sigma^2_{\Delta W}\right).$$

  The right-hand side is large when validation samples are scarce (ρ_v small) *and* the LLM error $\sigma_{\Delta W}$ is small relative to σ_{VW}. In other words, when the LLM is reasonably accurate and the inequality holds, the debiased estimator can be more precise than throwing the LLM away and using validation alone. LLM outputs serve to *amplify* a small validation sample, not to substitute for it.

- This phenomenon is the same one identified by prediction-powered inference (e.g., Kluger et al., 2025) and is closely related to the design-based supervised learning approach of Egami et al. (2022), the unifying framework of Carlson and Dell (2025), and the classical validation-data econometrics literature (Chen et al., 2005, 2008). It is also conceptually related to the bias corrections in Theorem 2: there the bias is identified through the structure $(\kappa, \mathbf{\Omega})$ of the measurement error; here it is identified directly from a labeled subset.

Monte-Carlo evidence on Congressional legislation. Ludwig et al. (2024, Section 4.3) run an extensive simulation in which the bill-topic labels V_r from the Congressional Bills Project are treated as ground truth, $\widehat{V}_r$ is generated by various LLM-prompt pairs, and the researcher reveals V_r on a random 5% (250 of 5,000) of bills. Across all combinations of bill topic, linked covariate, LLM, and prompt:

- The plug-in estimator $\widehat{\beta}$ exhibits substantial bias for the target β^*, while the debiased estimator $\widehat{\beta}^{\text{debiased}}$ is approximately unbiased (Figure 6).
- The nominal 95% CI centered at $\widehat{\beta}^{\text{debiased}}$ has approximately correct coverage; the plug-in CI suffers from severe coverage distortions (Table 5).
- The debiased estimator has *lower* mean squared error than the validation-only estimator across virtually all specifications in the main simulation, illustrating the precision gain from amplifying validation data with LLM outputs (Figure 7). Appendix results show that gains are strongest at smaller validation fractions and become more modest as the validation sample grows.
- In this Congressional-bill Monte Carlo, validation samples as small as 125 of 5,000 bills already give substantial bias and coverage gains.

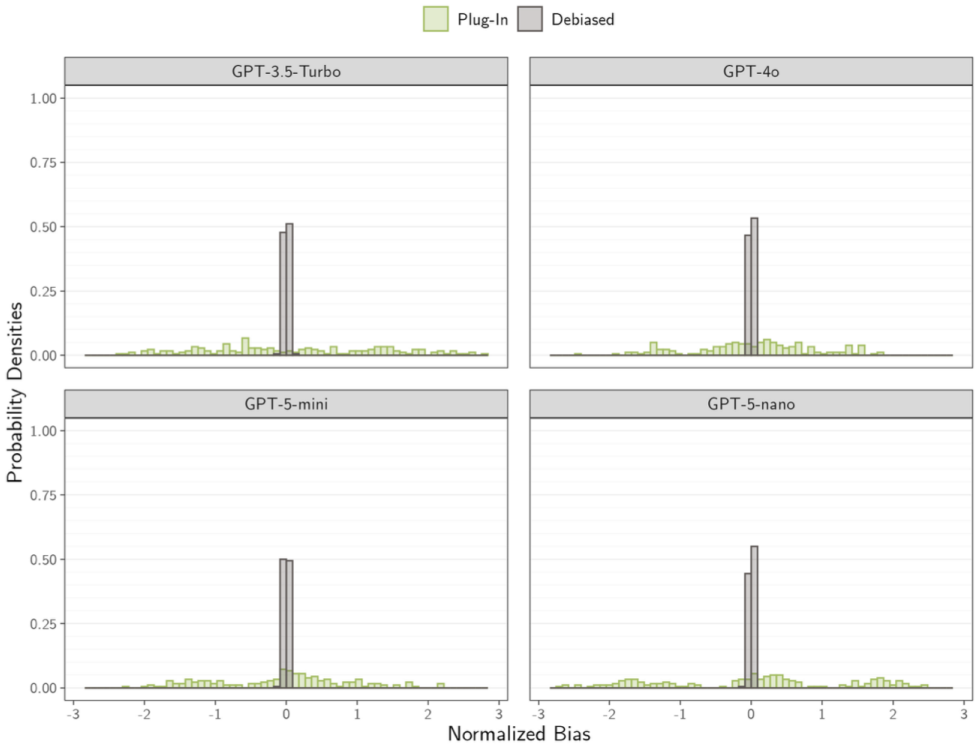


Figure 3: Normalized bias of the plug-in regression and bias-corrected regression across Monte Carlo simulations based on congressional legislation.

Notes: The normalized bias reports the average bias of the plug-in regression coefficient $\widehat{\beta}$ and the debiased coefficient $\widehat{\beta}^{debiased}$ for the target regression coefficient divided by their respective standard deviations across simulations. We summarize the distribution of normalized bias and coverage across regression specifications, choice of large language model and prompting strategies. For each combination of model topic V_r, linked covariate W_r, large language model m and prompting strategy p, we randomly sample $5,000$ Congressional bills and calculate the plug-in regression coefficient $\widehat{\beta}$ and the bias-corrected regression coefficient $\widehat{\beta}^{debiased}$ based on a 5% validation sample. See Section 4.3 for discussion.

Figure 6: Distribution of bias (normalized by standard deviation) for the plug-in coefficient $\widehat{\beta}$ and the debiased coefficient $\widehat{\beta}^{\text{debiased}}$ across LLM-prompt pairs, bill topics, and covariates in Congressional legislation Monte Carlo. Source: Ludwig et al. (2024, Figure 3).

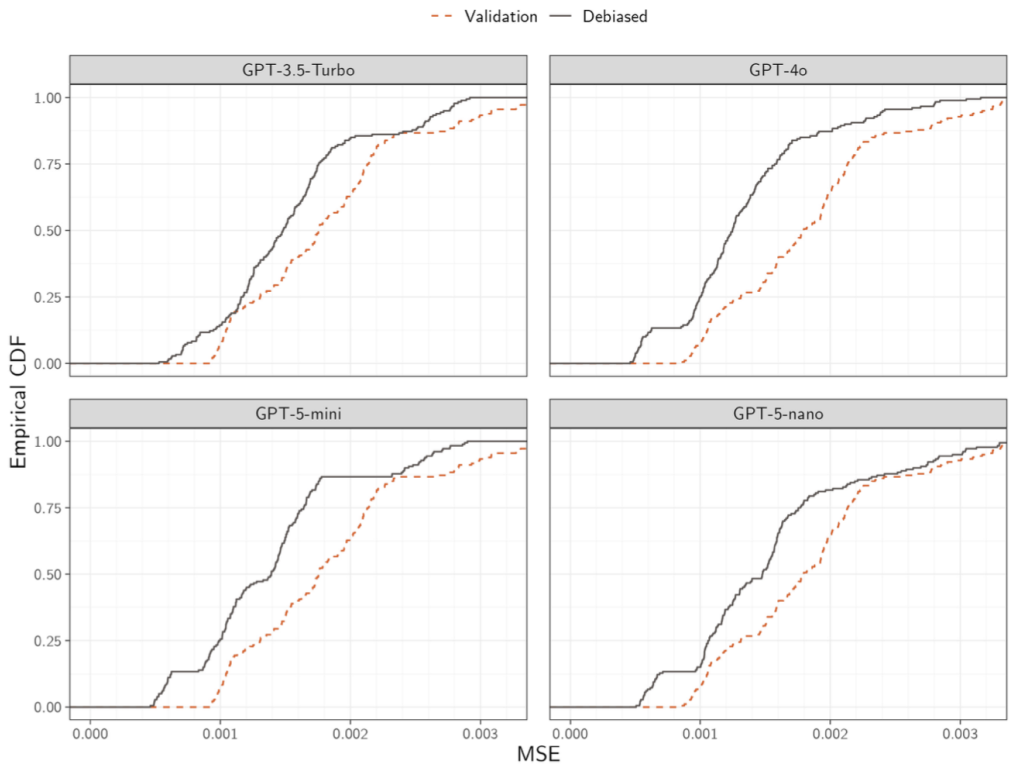


Figure 4: Cumulative distribution function of mean square error for the bias-corrected estimator against validation-sample only estimator.

Notes: For each combination of model topic V_r, covariate W_r, large language model m and prompting strategy p, we randomly sample 5,000 Congressional bills and calculate the bias-corrected regression coefficient using a 5% validation sample and the validation-sample only regression coefficient. We calculate the mean square error of $\widehat{\beta}^{debiased}$ and $\widehat{\beta}^{*}$ for the target regression, and we average the results over 1,000 simulations. We summarize the distribution of average mean square error across regression specifications, choice of large language model and prompting strategies. See Section 4.3 for discussion.

Figure 7: Distribution of MSE ratio MSE($\widehat{\beta}^{\text{validation}}$)/MSE($\widehat{\beta}^{\text{debiased}}$) across simulation cells. Values > 1 indicate lower MSE for the debiased estimator than for validation-only. Source: Ludwig et al. (2024, Figure 4).

Table 5: Coverage of nominal 95% CIs in the Congressional Bills Monte Carlo (Table 3 of Ludwig et al., 2024). Median coverage across LLM-prompt-topic-covariate cells; validation sample 5% of 5,000 bills.

Model	Plug-in $\widehat{\beta}$	Debiased $\widehat{\beta}^{\text{debiased}}$
GPT-3.5-Turbo	0.820	0.930
GPT-4o	0.920	0.927
GPT-5-mini	0.906	0.927
GPT-5-nano	0.779	0.930

3.6 Practical Recommendations

Ludwig et al. (2024, Section 6) provide a checklist; the key items mirror Section 1.10 and 2.9.

- **Diagnose the problem first.** Is the LLM being used to predict an outcome (prediction problem, Section 3.3) or to measure an economic concept used in a downstream regression (estimation problem, Sections 3.4–3.5)? The required conditions differ.
- **For prediction: rule out leakage by design.** Specify the target population, the sampling procedure, and how each relates to plausible LLM training corpora. Prefer open-source LLMs with fixed weights or time-stamped training data, paired with documents postdating the model's cutoff. Prompt-engineering "ignore information after τ" is unreliable.
- **For estimation: collect a small validation sample and debias.** On a random subset, apply the existing measurement f^*. Report both the plug-in estimate and the debiased estimate (23); bootstrap for inference. In their Congressional-bill Monte Carlo, validation samples in the hundreds perform well. Validation data also disciplines the choice of LLM and prompt: under the debiasing approach, alternative prompts target the *same* parameter β^*, eliminating the p-hacking incentive documented in Section 3.4.
- **Do not redefine the LLM output as ground truth.** Treating "the concept is whatever GPT-4o produces" makes plug-in estimation valid by definition but leaves the researcher unable to interpret estimates economically and forces the empirical literature to revisit every published estimate when models are updated.

4 Comparison and Takeaways

All three papers address the *same empirical temptation*: plug AI/ML outputs in for the latent variable of interest. They diagnose the problem differently and propose different fixes. We organize the comparison around three questions.

Where does the AI/ML output enter the regression?

- In Rambachan et al. (2024) (Section 1), the AI/ML output replaces a missing *outcome* in a randomized experiment. The RSV R is post-outcome (the outcome causes the RSV), so the relationship between R and Y is a measurement-error problem learned on a separate observational sample.
- In Battaglia et al. (2024) (Section 2), the AI/ML output replaces a missing *regressor* (or label) in an observational linear regression. The unstructured data $\mathbf{x}_i$ generates $\widehat{\boldsymbol{\theta}}_i$ but is not otherwise in the regression equation.

- In Ludwig et al. (2024) (Section 3), the LLM output can be used either to predict an outcome (prediction problem, Section 3.3) *or* to measure an economic concept that enters a downstream regression as a dependent variable or regressor (estimation problem, Sections 3.4–3.5). The estimation case is closest to Section 2 when the LLM output is a generated regressor, and closest to a classical validation-data problem when the LLM output is a generated dependent variable.

What does it take to recover valid inference?

- Rambachan et al. (2024) combine the experimental and observational samples through a stability assumption on $R \mid (X, D, Y)$ and identify the ATE via a conditional-moment representation that admits any *relevant* predictive representation of the RSV. Inference is valid *without* rate conditions on the learned representation, only a probability-limit assumption.

- Battaglia et al. (2024) introduce drifting-sequence asymptotics and obtain an explicit first-order location shift for two-step OLS, $-\kappa\,\mathbb{E}[\boldsymbol{\xi}_i\boldsymbol{\xi}_i']^{-1}\mathrm{diag}(\boldsymbol{\Omega}, \mathbf{0})\,\boldsymbol{\psi}$. They propose two bias-corrected estimators (additive and multiplicative) that use application-specific estimates of $(\kappa, \boldsymbol{\Omega})$, such as a small audit of false positives for generated labels.

- Ludwig et al. (2024) require, for prediction, the "no training leakage" condition (19), satisfiable through deliberate model choice and research design. For estimation, they show pointwise accuracy is insufficient (Proposition 4) and recommend a *validation-sample* debiased estimator (23), with asymptotic variance (24). The bias correction in Section 3.5 is a validation-data analogue of the generated-variable corrections in Section 2: both learn how ML errors enter the target moment, but LMR estimate that error projection directly on labeled validation data.

What new asymptotic framework is needed?

- Rambachan et al. (2024): representation-learning-compatible inference, exploiting infinite-order Neyman orthogonality of the conditional moment, so any reasonable ML method works.

- Battaglia et al. (2024): drifting-sequence asymptotics in which measurement error and sampling error remain comparable as $n \to \infty$, mirroring the modern setting of high-quality but imperfect ML on very large samples.

- Ludwig et al. (2024): a black-box framework that treats the LLM only through its input-output behavior and the joint distribution of (D, T), sidestepping the impossibility of modeling LLM training algorithms directly.

A common practical theme: diagnose before estimating.

- RSV (Section 1): inspect stability using available (D, Y) strata and alternative representations, and check RSV relevance (a weak-instrument-style test of $\mathbb{E}[\widetilde{H}(R)\,\Delta^o] \neq 0$).
- BCGS (Section 2): estimate κ (e.g., via a small audit of the unconditional FPR for labels, or via $\sqrt{n}\,\mathbb{E}[C_i^{-1}]$ for topic models) to gauge the magnitude of first-order bias.
- LMR (Section 3): for prediction, document training leakage by attempting verbatim text completion; for estimation, plot the dispersion of plug-in estimates across LLM and prompt choices, and use the validation sample to compare $\widehat{\beta}^{\text{debiased}}$ to $\widehat{\beta}$.
- In all three estimation settings, the naive two-step strategy is most problematic precisely when the ML algorithm is "good enough" that researchers are tempted to trust it: small unconditional false-positive probabilities compound in large samples, near-perfect satellite predictors still fail when the causal direction is reversed, and high-accuracy LLMs still produce errors whose correlation with covariates can flip downstream signs.

References

Athey, S., R. Chetty, G. W. Imbens, and H. Kang (2024). The surrogate index: Combining short-term proxies to estimate long-term treatment effects more rapidly and precisely. *NBER Working Paper Series*.

Baker, S. R., N. Bloom, and S. J. Davis (2016). Measuring economic policy uncertainty. *Quarterly Journal of Economics 131*(4), 1593–1636.

Bandiera, O., A. Prat, S. Hansen, and R. Sadun (2020). Ceo behavior and firm performance. *Journal of Political Economy 128*(4), 1325–1369.

Battaglia, L., T. Christensen, S. Hansen, and S. Sacher (2024). Inference for regression with variables generated by ai or machine learning. *arXiv preprint arXiv:2402.15585*.

Bound, J., C. Brown, G. J. Duncan, and W. L. Rodgers (1994). Evidence on the validity of cross-sectional and longitudinal labor market data. *Journal of Labor Economics 12*(3), 345–368.

Carlson, J. and M. Dell (2025). A unifying framework for robust and efficient inference with unstructured data. *arXiv preprint arXiv:2505.00282*.

Chamberlain, G. (1987). Asymptotic efficiency in estimation with conditional moment restrictions. *Journal of Econometrics 34*(3), 305–334.

Chen, X., H. Hong, and E. Tamer (2005). Measurement error models with auxiliary data. *The Review of Economic Studies 72*(2), 343–366.

Chen, X., H. Hong, and A. Tarozzi (2008). Semiparametric efficiency in GMM models with auxiliary data. *The Annals of Statistics 36*(2), 808 – 843.

Chernozhukov, V., J. C. Escanciano, H. Ichimura, W. K. Newey, and J. M. Robins (2022). Locally robust semiparametric estimation. *Econometrica 90*(4), 1501–1535.

Egami, N., C. J. Fong, J. Grimmer, M. E. Roberts, and B. M. Stewart (2022). How to make causal inferences using texts. *Science Advances 8*(42), eabg2652.

Gorodnichenko, Y., T. Pham, and O. Talavera (2023). The voice of monetary policy. *American Economic Review 113*(2), 548–584.

Hansen, S., P. J. Lambert, N. Bloom, S. J. Davis, R. Sadun, and B. Taska (2023). Remote work across jobs, companies, and space. *NBER Working Paper Series*.

Hansen, S., M. McMahon, and A. Prat (2018). Transparency and deliberation within the fomc. *Quarterly Journal of Economics 133*(2), 801–870.

Jack, B. K., S. Jayachandran, N. Kala, and R. Pande (2025). Money (not) to burn: payments for ecosystem services to reduce crop residue burning. *Working Paper*.

Kluger, D. M., K. Lu, T. Zrnic, S. Wang, and S. Bates (2025). Prediction-powered inference with imputed covariates and nonuniform sampling. *arXiv preprint arXiv:2501.18577*.

Ludwig, J., S. Mullainathan, and A. Rambachan (2024). Large language models: An applied econometric framework. *Annual Review of Economics 18*.

Mackey, L., V. Syrgkanis, and I. Zadik (2018). Orthogonal machine learning: Power and limitations. *International Conference on Machine Learning*.

Muralidharan, K., P. Niehaus, and S. Sukhtankar (2023). Identity verification standards in welfare programs: experimental evidence from india. *Working Paper*.

Newey, W. K. and D. McFadden (1994). Large sample estimation and hypothesis testing. *Handbook of econometrics 4*, 2111–2245.

Pagan, A. (1984). Econometric issues in the analysis of regressions with generated regressors. *International Economic Review 25*(1), 221–247.

Prentice, R. L. (1989). Surrogate endpoints in clinical trials: definition and operational criteria. *Statistics in Medicine 8*(4), 431–440.

Rambachan, A., R. Singh, and D. Viviano (2024). Program evaluation with remotely sensed outcomes. *arXiv preprint arXiv:2411.10959*.

Sarkar, S. K. and K. Vafa (2024). Lookahead bias in pretrained language models. *SSRN Working Paper*.