Adversarial Debiasing for Parameter Recovery

Luke Sanford **Megan Ayers**
School of the Environment
Yale University
New Haven, Connecticut 06511, USA
luke.sanford@yale.edu

Matthew Gordon **Eliana Stone**
Department of Statistics and Data Science
Yale University
New Haven, Connecticut 06511, USA
m.ayers@yale.edu

Department of Economics
Paris School of Economics
Paris, France
matthew.gordon@psemail.eu

School of the Environment
Yale University
New Haven, Connecticut 06511, USA
eliana.stone@yale.edu

Abstract

Advances in machine learning and the increasing availability of high-dimensional data have led to the proliferation of social science research that uses the predictions of machine learning models as proxies for outcomes of interest. However, prediction errors from machine learning models can lead to bias in downstream estimation tasks, including regression. In this paper, we show how this bias can arise, propose a test for detecting bias, and demonstrate the use of an adversarial machine learning algorithm in order to generate predictions suitable for unbiased downstream estimation. Here, we focus on a setting where machine-learned predictions are the dependent variable in a regression. We conduct simulations and empirical exercises using ground truth and satellite data on forest cover in Africa. Using the predictions from a naive machine learning model leads to biased parameter estimates, while the predictions from the adversarial model recover the true coefficients. Our approach consistently matches or exceeds the performance of existing methods.

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

1 INTRODUCTION

Efforts to combat climate change, alleviate poverty, and improve food security depend on studies that evaluate the effectiveness of policy interventions. These studies increasingly rely on satellite remote sensing data and machine learning methods to measure outcomes (biodiversity, forest carbon, wealth, agricultural productivity) of interest. Converting multispectral, multitemporal, and spatial data into these dependent variables relies on machine learning methods paired with labeled data to generate detailed maps of variables of interest. With the increasing availability of remote sensing data, decreasing cost of computation, and continuing improvement of machine learning tools, we are in the midst of a measurement revolution in social, economic, and environmental sciences.

These advances in measurement derive from the field of remote sensing, which generally develops tools to improve measurement of particular outcomes, Y, using a small labeled sample and satellite data. These approaches use satellite imagery features (k) to train a machine learning model that generates predictions:

$$\widehat{Y}_i(k_i) = Y_i + \nu_i, \tag{1}$$

where ν_i is the measurement error for a given observation. $\widehat{Y}$ can be written as the output of a prediction model, $\widehat{Y} = f(k_i, w)$, finds weights w that minimize some function of ν in the labeled data, for example the sum of squared error: $\sum_j \nu_j^2 = \sum_j (\widehat{Y}_j(k_j) - Y_j)^2$. Researchers often use this approach when the labeled observations are too few or their geographic coverage does not fit the research question. For example, researchers might measure a small number of field plots

to measure forest carbon while seeking to understand the broader forest carbon effects of forest carbon offset projects, or use data from several air quality monitoring stations to infer ambient air quality across a city. While this may be a sensible approach for some descriptive tasks, it can generate bias when $\widehat{Y}$ is used as a proxy for Y in policy evaluations.

The bias occurs because of differential measurement error with respect to policy assignment. For example, say we want to study the effect β of road construction X on forest cover Y. In practice, we run linear regression for the model:

$$\widehat{Y}_i = \alpha + \tilde{\beta} X_i + e_i, \tag{2}$$

using our remote sensing proxy for forest cover as the dependent variable. Our estimate of $\tilde{\beta}$, the treatment effect of X on *predicted* forest cover, will obtain in expectation:

$$\mathbb{E}[\widehat{\beta}] = \beta + \frac{\text{Cov}(e, X)}{\text{Var}(X)} + \frac{\text{Cov}(\nu, X)}{\text{Var}(X)}. \tag{3}$$

Here, β represents the true treatment effect of X on Y and the second term is standard omitted variable bias. The third term captures measurement error bias, which arises from using our proxy variable rather than true values of forest cover. It will bias our estimate as long as $\text{Cov}(\nu, X) \neq 0$.

Previous approaches to correct this bias take model predictions $\widehat{Y}$ as given, and then attempt to correct for this type of error using a subsample of points where the ground truth measures of Y are known (Proctor et al., 2023; Angelopoulos et al., 2023; Egami et al., 2023; Kluger et al., 2025; Miao et al., 2024; Salerno et al., 2025; Wang et al., 2020). In contrast, we describe a class of models that generate $\widehat{Y}$ such that the magnitude of $\text{Cov}(\nu, X)$ can be controlled by design. In particular, we borrow an approach from the algorithmic fairness literature where prediction errors are constrained to be uncorrelated with legally protected categories like race or gender (Zhang et al., 2018). Rather than having the adversarial component of the model attempt to predict one or more protected categories, we task it with predicting the independent variable(s) X. This approach generates predictions that are both precise and unbiased for the researcher's estimation task.

1.1 Contributions

The primary contributions of this paper are threefold. First, we introduce an adversarial debiasing model explicitly designed to produce predictions suitable for causal estimation. Unlike existing methods that correct bias in off-the-shelf or fixed-model predictions, our approach directly generates predictions with minimized measurement error bias, eliminating the need for subsequent bias-correction steps and potentially improving precision. Second, we develop a statistical test for detecting measurement error bias in machine-learned proxies. Third, we propose a practical power calculation framework that enables researchers to determine the required amount of ground truth labeling needed to reliably identify and mitigate measurement error bias in their analyses.

1.2 Related Work

This paper contributes to a growing literature that has begun to document the problem of non-random measurement error in machine-learning models (see (Jain, 2020) for a review[1]). A number of recent papers address the setting where researchers seek to correct correlated measurement error for a downstream estimation task by using some gold-standard labeled data. One approach, prediction-powered-inference, develops a loss function for the downstream estimator that removes the bias (Angelopoulos et al., 2023; Torchiana et al., 2023; Zhang, 2021; Egami et al., 2023). Another method known as predict-then-debias, estimates the bias and subtracts it from the coefficient of interest (Kluger et al., 2025). A third set of papers use a variety of ad-hoc approaches to correct the predictions themselves (Wang et al., 2020; Proctor et al., 2023; Miao et al., 2024; Ratledge et al., 2021). Our approach is closest to this last group, however we improve upon previous approaches by directly generating predictions that are suitable for downstream estimation. Finally, our approach relies on a similar intuition to the deconfounding literature, though it targets removing correlated measurement error rather than a latent confounder estimated via correlation in multiple treatments.

2 ADVERSARIAL DEBIASING

2.1 Detecting Bias

Unlike for omitted variable bias, an estimate of measurement error bias is obtainable if researchers have access to or can generate some ground-truth values of Y (for example, by visually interpreting high-resolution satellite imagery). Let $j \in J$ index observations in the labeled set. Consider the regression:

$$\nu_j = X_j \gamma + u_j, \tag{4}$$

[1]For topic-specific reviews, see Balboni et al. (2022) on deforestation, Gibson et al. (2021) and Bluhm & McCord (2022) on night lights, and Fowlie et al. (2019) on air pollution.

with X as a vector of independent variables, and $\nu_j = \widehat{Y} - Y$ is the prediction error. Our estimate of γ will be $\widehat{\gamma} = (X'X)^{-1}X'\nu$, which will converge to the multivariate analog of the bias term in equation 3 under the assumption that the labeled set is representative of the full population.

Estimates of the standard errors of $\widehat{\gamma}$ can be used to test whether the bias is significantly different from zero, or to rule out biases greater than a certain size. We adapt standard power calculations to estimate a 'minimum detectable bias' given a certain number of observations and an estimate of the standard error of $\widehat{\gamma}$. Researchers can estimate the number of labeled observations which will likely be necessary to rule out some amount of measurement error bias. We demonstrate this procedure in our first two applications.

Direct estimates of bias can also be used to perform a 'bias correction' on estimates of $\widehat{\beta}$ from equation 3:

$$\widehat{\beta}_c = \widehat{\beta} - \widehat{\gamma}. \tag{5}$$

In expectation, our bias-corrected estimate $\widehat{\beta}_c$ will converge to the true value of β when the labeled sample is representative. This is identical to the un-tuned model developed in Kluger et al. (2025). An estimate of the standard error of $\widehat{\beta}_c$ can be produced by a normal approximation as described in Angelopoulos et al. (2023) and Kluger et al. (2025) or with a bootstrap procedure.

There are two main benefits to this approach. First, $\widehat{\beta}$ and $\widehat{\gamma}$ can be generated with off-the-shelf predictions and a relatively small amount of ground-truth data, which is straightforward for those who do not want to generate a custom machine learning model for their research question as we describe in the following section. Second, we expect this approach to perform well when a very small number of ground truth observations are available. There are shortcomings of this estimator, however. The estimation uncertainty of $\widehat{\gamma}$ inflates the variance of $\widehat{\beta}_c$. Secondly, this approach takes a set of $\widehat{Y}$ predictions as given, which bounds the precision of $\widehat{\beta}$. In the next section, we describe an adversarial approach to obtain precise predictions of $\widehat{Y}$ without differential measurement error for a given estimation problem.

2.2 Adversarial Model

We investigate prediction models that directly avoid differential measurement error by construction. Given bias in the form of $\frac{\text{Cov}(\nu,X)}{\text{Var}(X)}$, we can formulate the model's objective function as a constrained optimization problem. In particular, we seek a model with parameters w^* that satisfy:

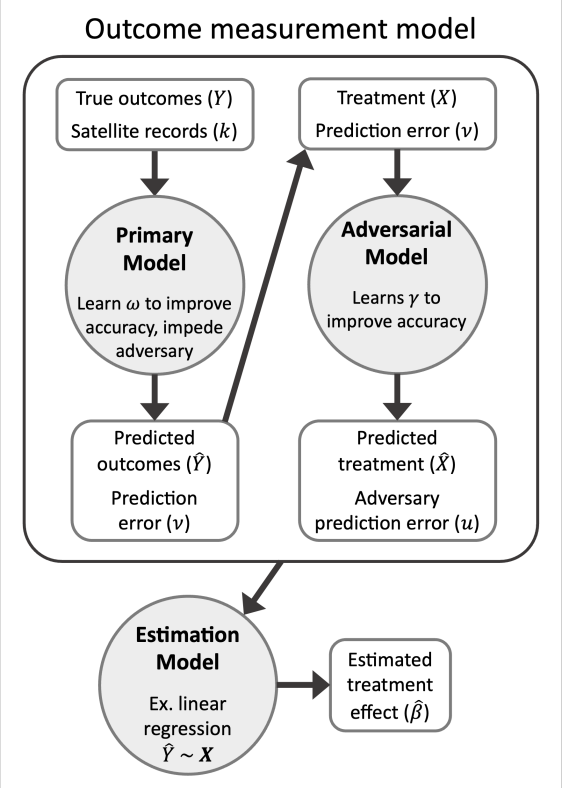


Figure 1: The model architecture of the debiasing approach.

$$\begin{aligned} \omega^* &= \arg\min_{\omega} L_p(\widehat{Y}(\omega), Y, k) \\ &\text{such that } \text{Cov}(X, Y - \widehat{Y}(\omega)) = 0, \end{aligned} \tag{6}$$

where L_p is a standard loss function, such as mean squared error. Now consider the adversarial debiasing model proposed by Zhang et al. (2018), which was originally used to debias machine learning model predictions with respect to race or gender. Zhang et al. (2018) introduce a secondary model, called the adversary, with model weights γ and loss function $L_a(\hat{X}(\gamma), X, Y, \widehat{Y}(\omega))$. Intuitively, the adversary tries to predict a 'protected' variable using the measurement errors from the primary model. In Zhang et al. (2018), the protected variable is race or gender; in our setup, the protected variable is X: an observation's treatment status.

Adversarial debiasing models are trained to minimize L_p, while maximizing L_a, subject to the adversary choosing γ in such a way as to minimize L_a (Figure 1). Formally, this can be written as the following objective function:

$$\min_{\omega} \max_{\gamma} \left\{ L_p\left(\hat{Y}(\omega), Y, k\right) - \alpha \cdot L_a\left(X, \hat{X}\left(\gamma\right), Y, \hat{Y}(\omega), k\right) \right\}$$
$$\text{such that } \gamma \in \operatorname{argmin}\, L_a(X, Y, \widehat{Y}(\omega), \gamma, k) \tag{7}$$

where α is a parameter that controls the weight on the adversary's loss function, and must be tuned (e.g. by cross fitting). Intuitively, by maximizing the adversary's loss function, the primary model tries to make sure that the measurement errors contain as little 'information' as possible about X. When linear regression is used as the downstream estimation model, we are specifically concerned with 'information' in the form of a linear relationship. Now consider an adversary model that is a linear model of the exact form of the bias test above:

$$\nu_j = X_j\gamma + \epsilon_j \tag{8}$$

with the adversary loss function as the sum of squared errors. The loss function of this adversary is minimized with respect to γ when $\gamma = (X'X)^{-1}X'\nu$. The primary model will try to choose ω such that the prediction errors ν maximize the adversary's loss function:

$$L_a = \frac{1}{N}\|\nu - X(X'X)^{-1}X'\nu\|^2,$$

while balancing this task with overall accuracy.

Consider prediction errors at step t and $t+1$ of training: ν_t and ν_{t+1}. Holding accuracy of the primary model predictions constant, moving in the opposite direction of the adversarial loss gradient means the bias at time $t+1$ will be smaller than at time t, $|\gamma_{t+1}| < |\gamma_t|$, assuming that treatment X is univariate (Proof in Appendix A). Thus, if α is sufficiently high, choosing ω to optimize equation 7 implies that the bias term will shrink if accuracy is stable. In practice, the model will balance the goal of maximizing the adversary's loss with the goal of overall prediction accuracy.

In our tests, a similar objective function that penalizes the covariance of X and ν directly in the loss function has improved performance:

$$\min_{\omega} L_p(\widehat{Y}(\omega), Y, k) - \alpha\left|\sum_j (x_j - \bar{x})\nu_j\right|. \tag{9}$$

This method is also applicable to bias terms that can be derived for regressions with controls, or an instrumental variables setting, with slight modifications (details in Appendix B). The associated algorithm is presented in Appendix C.

We experiment with both approaches; tests of the linear regression adversary are in Appendix H. For both methods, the choice of α is important. With too low of a value of α the model does not effectively debias the results and minimizes squared prediction error instead of maximizing the adversarial loss. However, when α is too high the model may produce uncorrelated but poor measurements to inflate the adversary's loss (e.g. a random $\widehat{Y}$ would satisfy our bias constraint). We use cross-fitting within the labeled data to set α. A practical rule-of-thumb is to increase α sequentially until the difference in average estimated biases between the α_{t+1} and α_t models, and between the α_{t+2} and α_t models, stays within a tolerance range of 1/2 the estimated bias standard error of the α_t model.

Kluger et al. (2025) show that a convex combination of the coefficient generated using the predicted $\widehat{Y}$, which we call $\widehat{\beta}_{\mathrm{D}}$ and the coefficient generated using the true values of Y estimated in the labeled ground-truth data, which we call $\widehat{\beta}_{\mathrm{GT}}$ reduces the variance of the resulting estimate, which we call $\widehat{\beta}^*$. In particular if:

$$\widehat{\beta}^* = (1-\lambda)\widehat{\beta}_{\mathrm{GT}} + \lambda\,\widehat{\beta}_{\mathrm{D}}, \tag{10}$$

we can choose λ to minimize $\mathrm{Var}(\widehat{\beta}^*)$ resulting in:

$$\lambda^* = \frac{\mathrm{Cov}(\widehat{\beta}_{\mathrm{GT}},\, \widehat{\beta}_{\mathrm{GT}} - \widehat{\beta}_{\mathrm{D}})}{\mathrm{Var}(\widehat{\beta}_{\mathrm{GT}} - \widehat{\beta}_{\mathrm{D}})}. \tag{11}$$

Note that λ is not a user-chosen parameter, but a weight that can be calculated directly. We refer to $\widehat{\beta}^*$ as estimates from the "tuned adversarial" model. We compare these with untuned estimates in Appendix H.

2.3 Standard Errors

The variance of $\widehat{\beta}_{\mathrm{D}}$ can be expressed as:

$$\begin{aligned}\mathrm{Var}(\widehat{\beta}_{\mathrm{D}}) =& \mathrm{Var}\left((X'X)^{-1}X'\widehat{Y}\right) \\ =& \mathrm{Var}\left((X'X)^{-1}X'e\right) + \mathrm{Var}\left((X'X)^{-1}X'\nu\right) + \\ & 2\mathrm{Cov}\left((X'X)^{-1}X'e, (X'X)^{-1}X'\nu\right)\end{aligned} \tag{12}$$

Holding the covariance term constant, larger prediction errors will inflate the variance of estimates of β through the second term, compared to when researchers have ground-truth labels for every observation. This highlights the potential efficiency gains from adversarial debiasing compared to bias correction. The bias correction methods take the set of prediction errors as given, and these are possibly from a non-optimized model. Formally, if adversarial model prediction errors are ν_a, standard model prediction errors are ν, and we assume covariance terms are 0, we have that

$$\begin{aligned}\text{Var}(\widehat{\beta}_{\text{D}}) =& \text{Var}((X'X)^{-1}X'\nu_a) + \text{Var}((X'X)^{-1}X'e)\\ \text{Var}(\widehat{\beta}_c) =& \text{Var}((X'X)^{-1}X'\nu) + \text{Var}((X'X)^{-1}X'e) +\\ & \text{Var}((\tilde{X}'\tilde{X})^{-1}\tilde{X}'u),\end{aligned} \tag{13}$$

where here $\tilde{X}$ and X distinguish data from the labeled data set the evaluation set respectively and β_c is from equation 5. Depending on the relative precision of the debiasing model predictions and the predictions used for the bias correction approach, and the number of labeled samples, the debiasing approach can result in efficiency gains. Assuming that debiased model predictions have a similar variance to the off-the-shelf predictions, the bias correction approach will be less efficient due to the additional uncertainty in estimating the bias term.

Using typical OLS standard errors or heteroskedasticity-consistent standard errors to estimate the variance of $\widehat{\beta}_{\text{D}}$ will underestimate uncertainty, because these do not take into account uncertainty about the prediction model itself. If the labeled training set was sampled again by the same mechanism and from the same source, it might lead to a different trained model, a different set of outcome measurements, and a different estimate of β. Even fixing a training sample, re-training these models with different seeds results in different estimates. We account for this by bootstrapping over the training of the machine learning model. This approach is computationally intensive, since it requires many training runs of a potentially complex machine learning model[2].

3 APPLICATIONS

To demonstrate the efficacy of our approach, we conduct simulations and empirical exercises predicting forest cover in settings where we have access to ground truth data. We use a dataset of 20,621 points in West Africa (Bastin et al. 2017); each point is labeled with percent treecover and a binary forest/not forest label. The data was originally collected in part to show the biases of the widely used Hansen et al. (2013) data set in dryland areas. Researchers used high-resolution imagery from between 2011 and 2015 to label the data — shown in Figure 2. We use a sample of the data in West Africa within 30 km of a road.

As inputs to our machine learning models, we use data from the Landsat 7 ETM sensor. This sensor records the surface reflectance of light at visible, near-infrared, and infrared wavelengths (called 'bands' in the remote sensing literature) at a 30-meter resolution. We generate three popular indices from these bands: Normalized Differenced Vegetation Index, Normalized Differenced Built Index, and Enhanced Vegetation Index. Over the course of a year, each location is observed up to 28 times (cloudiness obscures locations in some areas at some times). We take the 25th, 50th, and 75th percentiles of each of the eight bands and three indices, and use those 24 variables as inputs. This feature-engineering strategy mirrors the approach in Hansen et al. (2013). With a simple 1-layer neural network (a logistic regression) we are able to predict binary forest cover using this data with 75% accuracy.

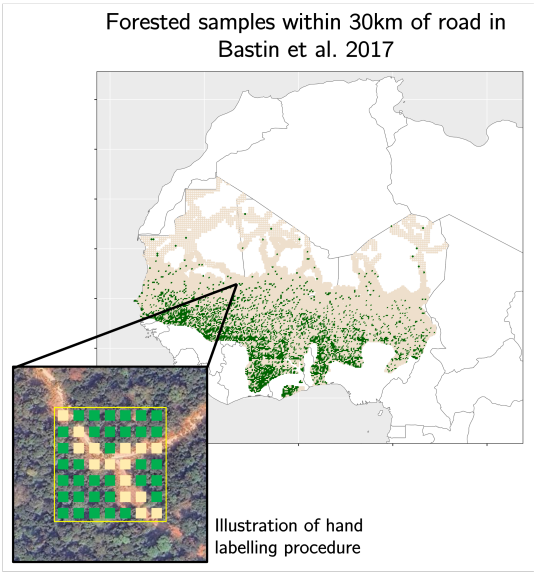


Figure 2: Colored areas show labeled pixels from Bastin (2017) — green for forested and beige for non-forested. Inset shows an example of how percent forested labels were generated from high-resolution satellite data.

3.1 Simulation experiment

All of the following empirical exercises take the following structure. First, we apply three-fold cross-fitting of a standard machine learning model so we have a ground-truth value of Y and a prediction $\widehat{Y}$ for each point. Then we add the adversarial constraint - in this case, the constraint that penalizes the absolute value of $\text{Cov}(X, \nu)$ - and conduct the same cross-fitting procedure. Finally, we estimate our regression of interest using the ground truth data, the baseline predictions, and adversarial model predictions. We also test multiple imputation (Proctor et al., 2023), the tuned PTD method (Kluger et al., 2025), PPI (Angelopoulos et al., 2023), and PostPI (Wang et al., 2020). For all approaches, we bootstrap standard errors using the same sample.

In this simulation and the next simple example, we

[2]In training the adversarial model takes 27.4 epochs compared to the 25.1 for the traditional model when using early stopping with a lookback window of five epochs.

focus on the cross-sectional relationship between roads and forest cover. Previous work has found that roads are an important driver of deforestation (Asher et al., 2020). Roads and other infrastructure are non-randomly placed, so this cross-sectional relationship is likely to generate confounder-induced bias (Supplemental Figure 8.c). Consider X to be the construction of a road, Y to be forest cover, and W to be some omitted geographic variable, like slope, that induces a correlation between the placement of roads X and measurement errors ν.

Our first simulation follows this procedure:

1. Draw 20,000 observations of W from a Poisson distribution with shape parameter of 1.
2. Assign each observation $X \sim$ Bernoulli with $p = \max(1 - W/4, 0)$, so that treatment is more likely when W is lower.
3. Assign each observation a random forest cover Y and associated satellite data k from the Bastin points.
4. If $W > 0$, make the satellite data artificially 'greener' without changing the true forest cover label. In practice, this is done by replacing k with the satellite data from a different point with a higher percent forest cover.

The true treatment effect of X is zero, since the forest labels Y are assigned randomly. However, the last step mimics a real source of bias — remotely sensed forest cover tends to be overestimated on steeper slopes since images are taken from above and tend to capture more trees in a smaller spatial area when on a slope. Because of this bias, and selection into treatment, it will appear that roads are associated with lower forest cover. Note also that there are no "traditional" confounders here — nothing in the simulation is associated with both road proximity and true forest cover.

Figure 3 shows the distribution of coefficient estimates from 100 bootstrapped runs of each of the models with $\alpha = 1$ and regressions using 10,000 training points and 10,000 unlabeled points. As predicted, using the baseline machine learning model results in a negative and significant estimate of the effect of X on Y. Failing to bootstrap standard errors that include prediction model uncertainty (e.g., using satellite data products as proxies for true values) leads to dramatically overstated precision. Both the bias correction method and the adversarial model result in coefficient distributions correctly centered around 0. [3]

[3]Model specifics and computational resources are described in Appendix E and Appendix F

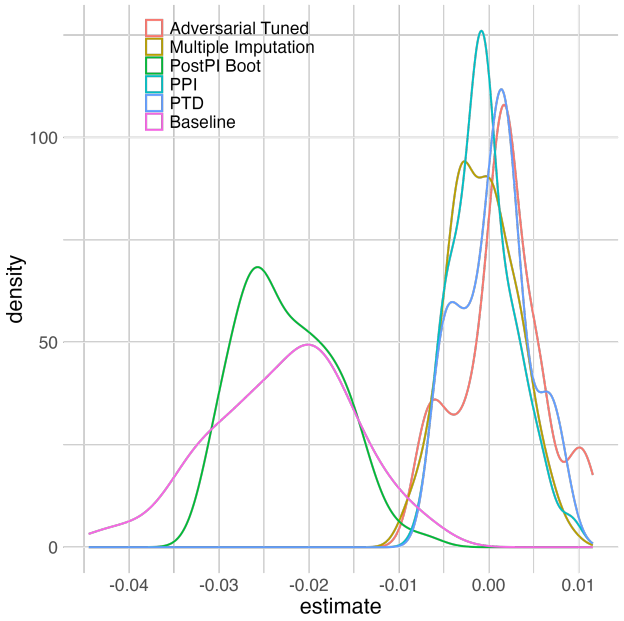


Figure 3: Simulation experiment estimate distributions from our proposed adversarial model ("Adversarial Tuned"), a baseline neural network model, and competing methods, each trained with 10,000 labeled observations. Each distribution includes the coefficients from each model after 100 runs on bootstrapped training data.

We test the performance of each model using a progressively increasing sample of labeled points. For each given sample size of labeled points J, we use the J labels to train the model, and then predict on the remaining points so that the sample size for the regression is always $N = 20,000$. For each J we bootstrap 100 different versions of the model to estimate standard errors of all models.

Figure 4 shows the results of this exercise. The baseline model generates biased estimates. The debiasing models, aside from PostPI, are centered on $\beta = 0$, and precision increases as the sample size of labeled points increases. Figure 3 shows the distributions of estimates at 10,000 samples.

In this setting, PPI and PTD perform best because the satellite data contains no information about the source of the bias, since it has been replaced by imagery from randomly drawn pixels with higher forest cover. This means that the adversarial model has to do a worse job predicting the low W, high Y observations so that the measurement error is balanced across X. Note that the adversary never has access to W, yet is still able to adjust for W-induced measurement error. In real-world cases, we expect the adversarial model to outperform PTD and PPI methods by learning representations

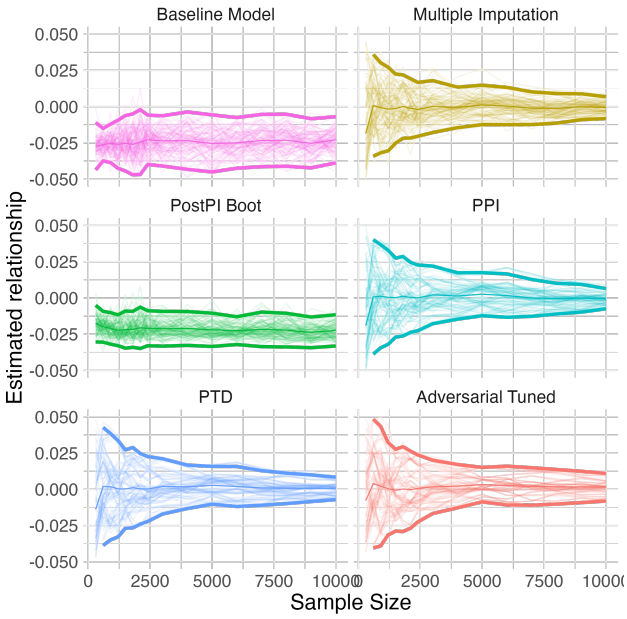


Figure 4: Estimates from our proposed adversarial model ("Adversarial Tuned"), a baseline neural network model, and competing methods across training sample sizes. Each light colored line is an individual training run where researchers label progressively more observations. The thick lines represent the mean and 2σ intervals.

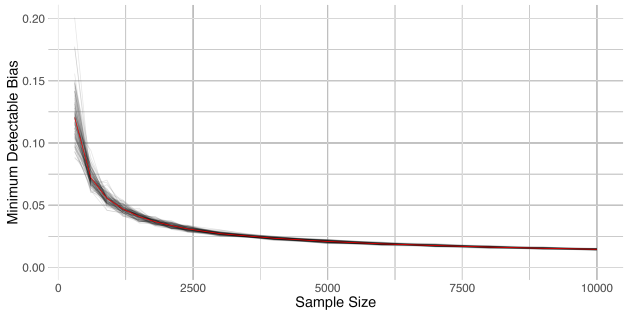


Figure 5: Minimum detectable bias (MDB) estimates across sample sizes at power of 0.8 and $\alpha = 0.05$. Each black line represents an estimate of MDB using a different random sample of labeled data, and the red line is the true MDB using standard errors estimated with the whole dataset.

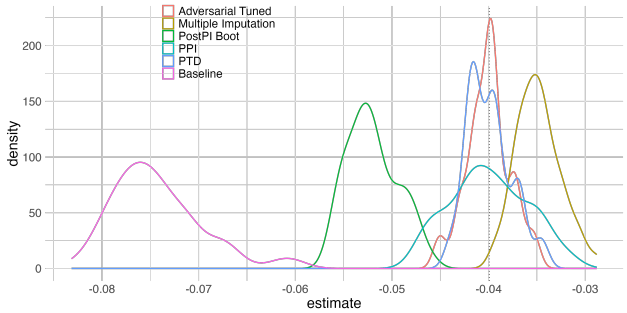


Figure 6: Distributions of estimated effects of roads on forest cover from our proposed adversarial model ("Adversarial Tuned"), a baseline neural network model, and competing methods, each trained with 10,000 labeled observations. Each distribution includes coefficients from each model after 100 runs on bootstrapped training data. The dashed vertical line represents a "true" estimate using all 20,000 ground truth labels.

that predict bias and adjusting for them.

Finally, we conduct power analyses of our bias test to estimate the minimum detectable bias (MDB) at different sample sizes. The results are presented in Figure 5. Each line represents a different random draw of points to label. At each sample size, for each set of points, we estimate the standard error of γ and use that to perform a standard power calculation using 0.8 power and 95% statistical significance. Given that the true magnitude of the bias is 0.025 in this simulation, researchers would need to label more than 2,500 points to detect this bias as statistically different from zero 80% of the time.

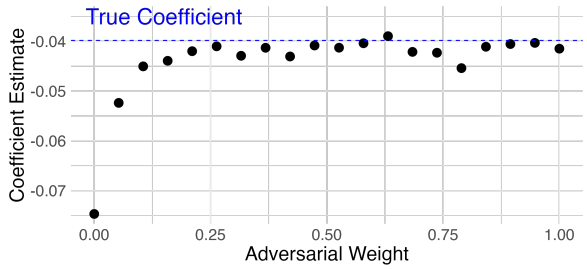
3.2 Empirical Application: Roads and Forest Cover

Next we use the true Bastin (2017) data, and data on the African road network from Meijer et al. (2018), to estimate the gradient of forest cover with respect to distance to the nearest road. Whereas before the independent variable was binary and the outcome was continuous, now our independent variable is continuous, log distance to the nearest road, and our outcome is binary (forested or not). We apply the same cross-fitting procedure as above for a standard model, an adversarial model using a 3-layer neural network, the bias correction approach, and the multiple imputation method recommended by Proctor et al. (2023). We then run the same regressions as in Section 3.1. The results are shown in Figure 6.

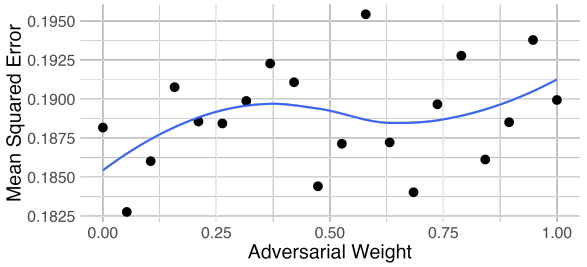
The baseline model overestimates the negative relationship between proximity to roads and forest cover. The multiple imputation and PostPI approaches produce biased estimates. PPI and PPD are unbiased but not as efficient as the tuned adversarial model, which produces unbiased and precise estimates.[4]

We use this setting to run experiments on the weight on the adversary in the loss function α. The results are summarized in Figure 7 for two different primary pre-

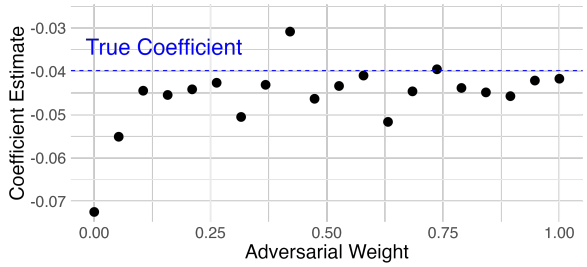
[4]Results from 1200, 5000 labeled observation tests are included in Appendix G



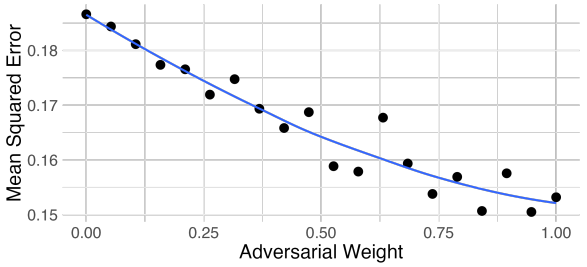
(a) Bias: Logistic Regression Primary Model



(b) Accuracy: Logistic Regression Primary Model



(c) Bias: DNN Primary Model



(d) Accuracy: DNN Primary Model

Figure 7: Tuning α: Tradeoffs between bias and accuracy. Graphs show how coefficient bias (left column) and overall predictive accuracy (MSE - right column) change as the weight on the adversary increases. Top row shows results for a primary model that is a logistic regression. Bottom row shows results for a Deep Neural Net (DNN - 3 layers). Blue lines show Loess-smoothed best-fit curves.

diction models, a logistic regression and a deep neural net (DNN). The left column shows that increasing the weight on the adversary from zero (standard model) to 1 quickly eliminates bias in the estimated coefficients. For each value of α, we run three-fold cross validation, then average the resulting coefficient estimates and accuracy metrics. We expected the MSE to increase at higher α values, and this is the case for the logistic primary model. For the DNN, however, increasing the adversary's weight improves prediction accuracy. This result can be understood as a kind of regularization effect. In some settings with many model parameters, it is well known that a regularization penalty can reduce overfitting and improve out-of-sample prediction performance (Tibshirani, 1996). While it is difficult to say precisely when adversarial models will improve prediction accuracy, the adversary seems to be serving a regularization function in this example.

4 DISCUSSION

We have introduced an adversarial machine learning approach to debiasing predictions for downstream inference. Our results show that it effectively debiases predictions in a simulation where the bias is artificially constructed, and in an application where researchers estimate the relationship between roads and forest cover. We also construct a test for measurement error bias and guidelines for researchers to estimate the number of observations they will need to label to detect and remove the bias. We also show that this approach can match or exceed the performance of competing algorithms. We show how this debiasing algorithm can be applied when machine-learned predictions will be used as an outcome variable for an inference task, and does not require any knowledge of the structure of measurement error bias.

We encourage researchers who use machine learned predictions to test and correct for measurement error bias using the following procedure. First, obtain some ground-truth labels, either from an existing data source or by labeling them yourself. Then, use the procedure from section 2.1 to test for measurement error bias, noting that the power of the test depends on the number of labeled points. If researchers discover bias, they have several options for debiasing. If they conducted the machine learning prediction stage, they should implement the adversarial debiasing approach described above, as it appears to efficiently generate unbiased estimates. If they are using predictions from a model that is not available for debiasing, we recommend the predict-then-debias approach (Kluger, 2025), as it is the most efficient of the post-hoc corrections in our tests.

The main limitation of our approach is its accessibility compared to competing methods. It requires researchers to build an adversarial measurement model,

which, to our knowledge, is only feasible for methods that use stochastic gradient descent-based optimization algorithms. This approach necessitates internalizing the measurement part of the research pipeline, potentially broadening the expertise needed for a project and increasing computational demands. This is in contrast to approaches such as PPI or PTD that are relatively easy to implement and require few computational resources.

In future work, we plan to expand the results here to a broader set of downstream tasks, including handling errors in independent variables in regressions as well as estimating other population parameters. We also plan to extend the types of models for which this approach can be implemented beyond deep learning. As researchers in remote sensing or other computational fields increasingly turn to constructing their own measurement models, we expect our approach to become more useful compared to approaches that debias existing predictions.

Acknowledgements

Thanks to Dan Kluger, the Berkeley Political Methodology Seminar, the Washington University St. Louis Political Methodology Seminar, the LSE/Imperial College Workshop, and participants at the ASSA, AGU, ICLR, TWEEDS, and AERE conferences for helpful feedback and support.

References

Anastasios N. Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I. Jordan, and Tijana Zrnic. Prediction-Powered Inference, November 2023. URL http://arxiv.org/abs/2301.09633. arXiv:2301.09633.

Sam Asher, Teevrat Garg, and Paul Novosad. The Ecological Impact of Transportation Infrastructure. *The Economic Journal*, 130(629):1173–1199, July 2020. ISSN 0013-0133. doi: 10.1093/ej/ueaa013. URL https://doi.org/10.1093/ej/ueaa013.

Claire Balboni, Aaron Berman, Robin Burgess, and Benjamin Olken. The Economics of Tropical Deforestation. *Working Paper*, 2022. URL https://economics.mit.edu/sites/default/files/2022-09/ARE_Tropical_Deforestation-3.pdf.

Jean-François et al. Bastin. The extent of forest in dryland biomes. *Science*, 356(6338): 635–638, May 2017. doi: 10.1126/science.aam6527. URL https://www.science.org/doi/10.1126/science.aam6527. Publisher: American Association for the Advancement of Science.

Richard Bluhm and Gordon C. McCord. What Can We Learn from Nighttime Lights for Small Geographies? Measurement Errors and Heterogeneous Elasticities. *Remote Sensing*, 14(5):1190, January 2022. ISSN 2072-4292. doi: 10.3390/rs14051190. URL https://www.mdpi.com/2072-4292/14/5/1190. Number: 5 Publisher: Multidisciplinary Digital Publishing Institute.

Naoki Egami, Musashi Jacobs-Harukawa, Brandon M. Stewart, and Hanying Wei. Using Large Language Model Annotations for Valid Downstream Statistical Inference in Social Science: Design-Based Semi-Supervised Learning, June 2023.

Meredith Fowlie, Edward Rubin, and Reed Walker. Bringing Satellite-Based Air Quality Estimates Down to Earth. *AEA Papers and Proceedings*, 109:283–288, May 2019. ISSN 2574-0768. doi: 10.1257/pandp.20191064. URL https://www.aeaweb.org/articles?id=10.1257/pandp.20191064.

John Gibson, Susan Olivia, Geua Boe-Gibson, and Chao Li. Which night lights data should we use in economics, and where? *Journal of Development Economics*, 149:102602, March 2021. ISSN 0304-3878. doi: 10.1016/j.jdeveco.2020.102602. URL https://www.sciencedirect.com/science/article/pii/S0304387820301772.

M. C. Hansen, P. V. Potapov, R. Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland, A. Kommareddy, A. Egorov, L. Chini, C. O. Justice, and J. R. G. Townshend. High-Resolution Global Maps of 21st-Century Forest Cover Change. *Science*, 342(6160):850–853, November 2013. doi: 10.1126/science.1244693. URL https://www.science.org/doi/10.1126/science.1244693. Publisher: American Association for the Advancement of Science.

Meha Jain. The Benefits and Pitfalls of Using Satellite Data for Causal Inference. *Review of Environmental Economics and Policy*, 14(1):157–169, January 2020. ISSN 1750-6816. doi: 10.1093/reep/rez023. URL https://academic.oup.com/reep/article/14/1/157/5735430. Publisher: Oxford Academic.

Dan M. Kluger, Kerri Lu, Tijana Zrnic, Sherrie Wang, and Stephen Bates. Prediction-Powered Inference with Imputed Covariates and Nonuniform Sampling, April 2025.

Johan R. Meijer, Mark A. J. Huijbregts, Kees C. G. J. Schotten, and Aafke M. Schipper. Global patterns of current and future road infrastructure. *Environmental Research Letters*, 13(6):064006, May 2018. ISSN 1748-9326. doi: 10.1088/1748-9326/aabd42.

Jiacheng Miao, Yixuan Wu, Zhongxuan Sun, Xinran Miao, Tianyuan Lu, Jiwei Zhao, and Qiongshi Lu.

Valid inference for machine learning-assisted genome-wide association studies. *Nature Genetics*, 56(11): 2361–2369, November 2024. ISSN 1546-1718. doi: 10.1038/s41588-024-01934-0.

Jonathan Proctor, Tamma Carleton, and Sandy Sum. Parameter Recovery Using Remotely Sensed Variables. *NBER Working Paper*, 2023.

Nathan Ratledge, Gabriel Cadamuro, Brandon De la Cuesta, Matthieu Stigler, and Marshall Burke. Using Satellite Imagery and Machine Learning to Estimate the Livelihood Impact of Electricity Access, September 2021. URL `https://www.nber.org/papers/w29237`.

Stephen Salerno, Jiacheng Miao, Awan Afiaz, Kentaro Hoffman, Anna Neufeld, Qiongshi Lu, Tyler H McCormick, and Jeffrey T Leek. Ipd: An R package for conducting inference on predicted data. *Bioinformatics*, 41(2):btaf055, February 2025. ISSN 1367-4811. doi: 10.1093/bioinformatics/btaf055.

Robert Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. ISSN 0035-9246. URL `https://www.jstor.org/stable/2346178`. Publisher: [Royal Statistical Society, Wiley].

Adrian L. Torchiana, Ted Rosenbaum, Paul T. Scott, and Eduardo Souza-Rodrigues. Improving Estimates of Transitions from Satellite Data: A Hidden Markov Model Approach. *The Review of Economics and Statistics*, pp. 1–45, March 2023. ISSN 0034-6535. doi: 10.1162/rest_a_01301. URL `https://doi.org/10.1162/rest_a_01301`.

Siruo Wang, Tyler H. McCormick, and Jeffrey T. Leek. Methods for correcting inference based on outcomes predicted by machine learning. *Proceedings of the National Academy of Sciences*, 117(48):30266–30275, December 2020. doi: 10.1073/pnas.2001238117.

Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating Unwanted Biases with Adversarial Learning, January 2018. URL `http://arxiv.org/abs/1801.07593`. arXiv:1801.07593 [cs].

Han Zhang. How Using Machine Learning Classification as a Variable in Regression Leads to Attenuation Bias and What to Do About It. preprint, SocArXiv, May 2021. URL `https://osf.io/453jk`.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]

(c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

Adversarial Debiasing for Parameter Recovery

1 PROOF THAT MAXIMIZING ADVERSARIAL LOSS DECREASES BIAS WHEN ACCURACY IS HELD CONSTANT

Define $P = X(X'X)^{-1}X'$ as the n x n symmetric and idempotent projection matrix, and $\mathbb{I}$ as the n x n identity matrix. The adversary's loss function is:

$$\begin{aligned} L_a &= (\nu - P\nu)'(\nu - P\nu) \\ &= v'(\mathbb{I} - P)'(\mathbb{I} - P)\nu \\ &= v'(\mathbb{I} - P)\nu = \nu'\nu - \nu' P\nu \end{aligned} \tag{1}$$

by the properties of the projection matrix. Consider prediction errors at step t and $t+1$ of training: ν_t and ν_{t+1}, and suppose that their prediction accuracy is equivalent. Maximizing the adversarial loss implies that $L_a(\nu_{t+1}) > L_a(\nu_t)$. Therefore,

$$\nu'_{t+1}\nu_{t+1} - \nu'_{t+1}P\nu_{t+1} > \nu'_t\nu_t - \nu'_t P\nu_t \tag{2}$$

$$\nu'_{t+1}P\nu_{t+1} < \nu'_t P\nu_t$$

$$\nu'_{t+1}PP\nu_{t+1} < \nu'_t PP\nu_t \tag{3}$$

$$\nu'_{t+1}X(X'X)^{-1}X'X(X'X)^{-1}X'\nu_{t+1} < \nu'_t X(X'X)^{-1}X'X(X'X)^{-1}X'\nu_t$$

$$\gamma_{t+1}X'X\gamma_{t+1} < \gamma_t X'X\gamma_t$$

$$|\gamma_{t+1}| < |\gamma_t|. \tag{4}$$

Concluding the proof.

2 DEBIASING WITH CONTROL VARIABLES AND INSTRUMENTS

Adding Control Variables

Assume we want to estimate β_1 in the regression

$$\widehat{Y}_i = \beta_1 x_1 + x_2 \beta_2 + e_i \tag{5}$$

where x_1 is an n x 1 vector of the treatment variable, and x_2 is an n x k matrix of control variables. By the Frisch-Waugh-Lovell theorem, we can write $\widehat{\beta}_1$ as:

$$\widehat{\beta}_1 = (\tilde{X}_1' \tilde{X}_1)^{-1} \tilde{X}_1' \tilde{\widehat{Y}} \tag{6}$$

where $\tilde{X}_1$ are the residuals of the regression of X_1 on X_2, and $\tilde{\widehat{Y}}$ are the residuals of the regression of $\widehat{Y}$ on X_2. $\widehat{\beta}_1$ can thus be rewritten as follows:

$$\begin{aligned} \widehat{\beta}_1 &= (\tilde{X}_1' \tilde{X}_1)^{-1} \tilde{X}_1' (\mathbb{I} - X_2 (X_2' X_2)^{-1} X_2')(Y + \nu) \\ &= (\tilde{X}_1' \tilde{X}_1)^{-1} \tilde{X}_1' (\mathbb{I} - X_2 (X_2' X_2)^{-1} X_2') Y + \\ &\quad (\tilde{X}_1' \tilde{X}_1)^{-1} \tilde{X}_1' (\mathbb{I} - X_2 (X_2' X_2)^{-1} X_2') \nu \\ &= (\tilde{X}_1' \tilde{X}_1)^{-1} \tilde{X}_1' \tilde{Y} + (\tilde{X}_1' \tilde{X}_1)^{-1} \tilde{X}_1' \tilde{\nu} \end{aligned} \tag{7}$$

where $\tilde{\nu}$ are the residuals of the regression of ν on X_2. The expectation of this estimate is

$$\mathbb{E}[\beta_1] = \beta_1 + \frac{\mathrm{Cov}(\tilde{X}_1, \tilde{\nu})}{\mathrm{Var}(\tilde{X}_1)}. \tag{8}$$

Intuitively this makes sense – if the residual variation in X_1 is correlated with the residual prediction error, after controlling for X_2 in both cases, our estimate will be biased. Thus following the same logic as above, we can make the adversary a linear regression of $\tilde{\nu}$ on $\tilde{X}_1$. The same logic applies for when adding additional covariates to the regression.

Instrumental Variables

A similar argument can be extended to the instrumental variables case with controls. Following the two-stage least squares estimation procedure, we first use covariates X_2 and instruments Z to predict X_1:

$$\widehat{X}_1^{IV} = C(C'C)^{-1} C' X_1 \tag{9}$$

where $C = [1, X_2, Z]$. Then, we regress the outcome against the predicted values of X_1 and the covariates X_2 and take the estimated coefficient for $\widehat{X}_1^{IV}$ in this second stage regression as our estimate of the true β_1. By the Frisch-Waugh-Lovell theorem, we have

$$\begin{aligned} \widehat{\beta_1^{2SLS}} &= (\tilde{\widehat{X}}_1{}' \tilde{\widehat{X}}_1)^{-1} \tilde{\widehat{X}}_1{}' \tilde{\widehat{Y}} \\ &= (\tilde{\widehat{X}}_1{}' \tilde{\widehat{X}}_1)^{-1} \tilde{\widehat{X}}_1{}' \tilde{Y} + (\tilde{\widehat{X}}_1{}' \tilde{\widehat{X}}_1)^{-1} \tilde{\widehat{X}}_1{}' \tilde{\nu} \end{aligned} \tag{10}$$

where $\widetilde{\widehat{X_1}}$ are the residuals from regressing $\widehat{X}_1^{IV}$ on X_2, $\tilde{Y}$ are the residuals from regressing Y on X_2, and $\tilde{\nu}$ are the residuals from regressing ν on X_2.

In the case of a single instrument Z, this estimator of β_1 can be rewritten more simply following the indirect least-squares procedure. For this approach, we perform linear regressions for the models

$$\widehat{Y}_i = \gamma_0 + \gamma_1 w_i + \gamma_2' X_{2i} + w_i; \tag{11}$$

$$\widehat{X}_{1i} = \alpha_0 + \alpha_1 Z_i + \alpha_2' X_{2i} + u_i \tag{12}$$

This produces the following estimate of β_1, which coincides with $\widehat{\beta_1}^{2SLS}$ for this special case:

$$\begin{aligned} \widehat{\beta}_1^{ILS} = \frac{\widehat{\gamma_1}}{\widehat{\alpha_1}} = \frac{(\tilde{Z}'\tilde{Z})^{-1}\tilde{Z}'\tilde{\tilde{Y}}}{(\tilde{Z}'\tilde{Z})^{-1}\tilde{Z}'\tilde{X}_1} = \frac{\mathrm{Cov}(\tilde{Z}, \tilde{\tilde{Y}})}{\mathrm{Cov}(\tilde{Z}, \tilde{X}_1)} \\ = \frac{\mathrm{Cov}(\tilde{Z}, \tilde{Y})}{\mathrm{Cov}(\tilde{Z}, \tilde{X}_1)} + \frac{\mathrm{Cov}}{\mathrm{Cov}(\tilde{Z}, \tilde{X}_1)} \end{aligned} \tag{13}$$

In the above expressions for $\widehat{\beta_1}^{2SLS}$ and $\widehat{\beta}_1^{ILS}$, we see that the coefficient estimates are the sum of the coefficient estimate that we would obtain from performing these procedures given Y without measurement error - which is consistent for β_1 given IV assumptions - and an additional bias term involving the measurement error ν. To minimize this bias, we propose an adversary in the form of a regression of $\tilde{\nu}$ on $\tilde{Z}$ for the single instrument case, or $\tilde{\nu}$ on $\tilde{\tilde{X}}_1$ more generally.

3 ADVERSARIAL DEBIASER ALGORITHM

Algorithm 1 Adversarial Debiasing Algorithm

Require: Labeled data $\{(k_j, x_j, y_j)\}_{j=1}^N$; primary model $P(\cdot;\omega)$; adversary $A(\cdot;\gamma)$; loss functions L_p and L_a; learning rates $\eta_\omega,\ \eta_\gamma$; adversarial weight α; number of training iterations T.

1: Initialize primary model parameters ω.
2: Initialize adversary parameters γ.
3: **for** $t = 1, \ldots, T$ **do**
4: **Primary Model Forward Pass:**
5: For each sample j, compute $\widehat{Y}_j \leftarrow P(k_j;\omega)$.
6: Compute prediction errors $\nu_j \leftarrow y_j - \widehat{Y}_j(\omega)$.
7: **Adversary Update (minimize L_a):**
8: $\gamma \leftarrow \gamma - \eta_\gamma \nabla_\gamma \big[L_a(x_j, \nu_j, \gamma)\big]$.
9: **Primary Model Update (minimize $L_p - \alpha\, L_a$):**
10: $\omega \leftarrow \omega - \eta_\omega \nabla_\omega \Big[L_p(\widehat{Y},\, y) - \alpha\, L_a(x_j, \nu_j, \gamma)\Big]$.
11: **end for**
return Trained primary model parameters ω^*

4 POTENTIAL SOURCES OF BIAS

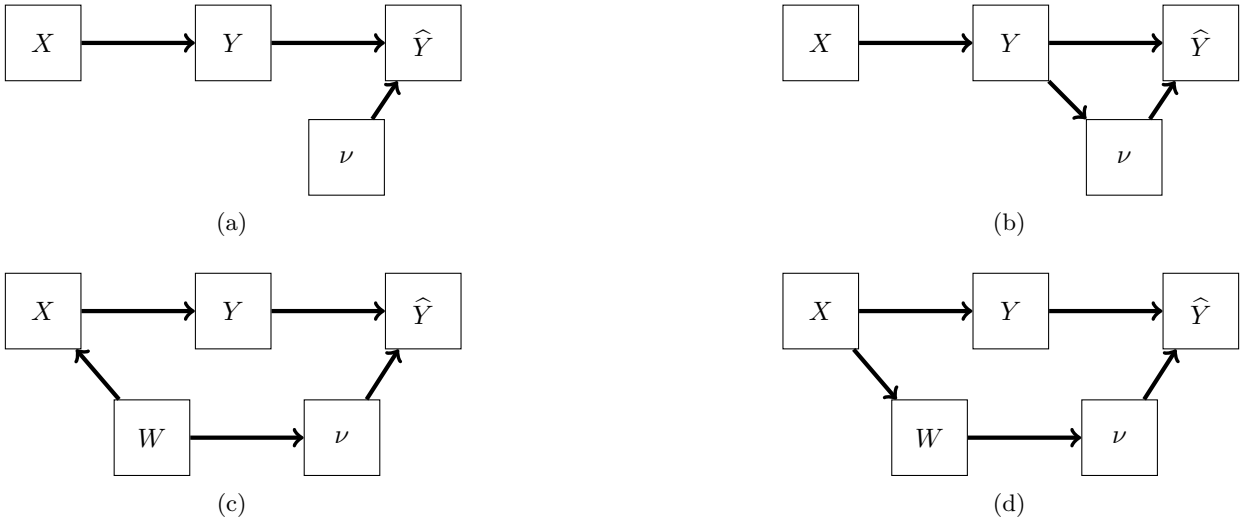


Figure 1: Four Directed-Acrylic-Graphs illustrating potential relationships between treatment (X), outcomes (Y), measurement error (ν), machine learning model predictions ($\widehat{Y}$), and unobserved variables (W). (a) is classical measurement error, (b) shows outcome-induced bias, (c) shows confounder-induced bias, and (d) shows treatment-induced bias.

5 TRAINING AND TEST DETAILS

All details of the model training and testing can be found in the code package, but we provide a quick overview here. For the baseline model we use a two layer neural network with 32 nodes in each layer, followed by a single final layer with a sigmoid activation function if the outcome is categorical or a linear activation if the outcome is continuous. For categorical outcomes we use a binary cross-entropy loss function and for continuous outcomes we use mean squared error. We apply batch normalization before the first layer. We allow the model to train for up to 50 epochs with an early stopping rule on improvement of the mae with a patience of 10. We use an adam optimizer and we set the number of cross-fit folds to 3.

For the adversarial models we use the same machine learning model described above with the addition of the adversarial loss component described in the paper. For the correlational adversary described in the paper we set the adversarial weight $\alpha = 1$

6 COMPUTATION RESOURCES

Our main tests consisted of 100 runs of each algorithm at the following sample sizes of labeled observations out of 20,000 total observations: 300, 600, 900, 1200, 1500, 1800, 2100, 2400, 3000, 4000, 5000, 6000, 7000, 8000, 9000, and 10000. Each faint line in Figure 4 corresponds to one run where 300 points were sampled, then 300 more, progressively sampling up to 10,000 labeled observations. We estimated coefficients of interest for each debiasing method for each sample size and run.

This task was split over 20 job submissions, each containing 5 runs. Each submission used one node with 4 CPUs and 8gb of memory per CPU. In total the process took 3.5 hours.

7 RESULTS FROM DIFFERENT SAMPLE SIZES

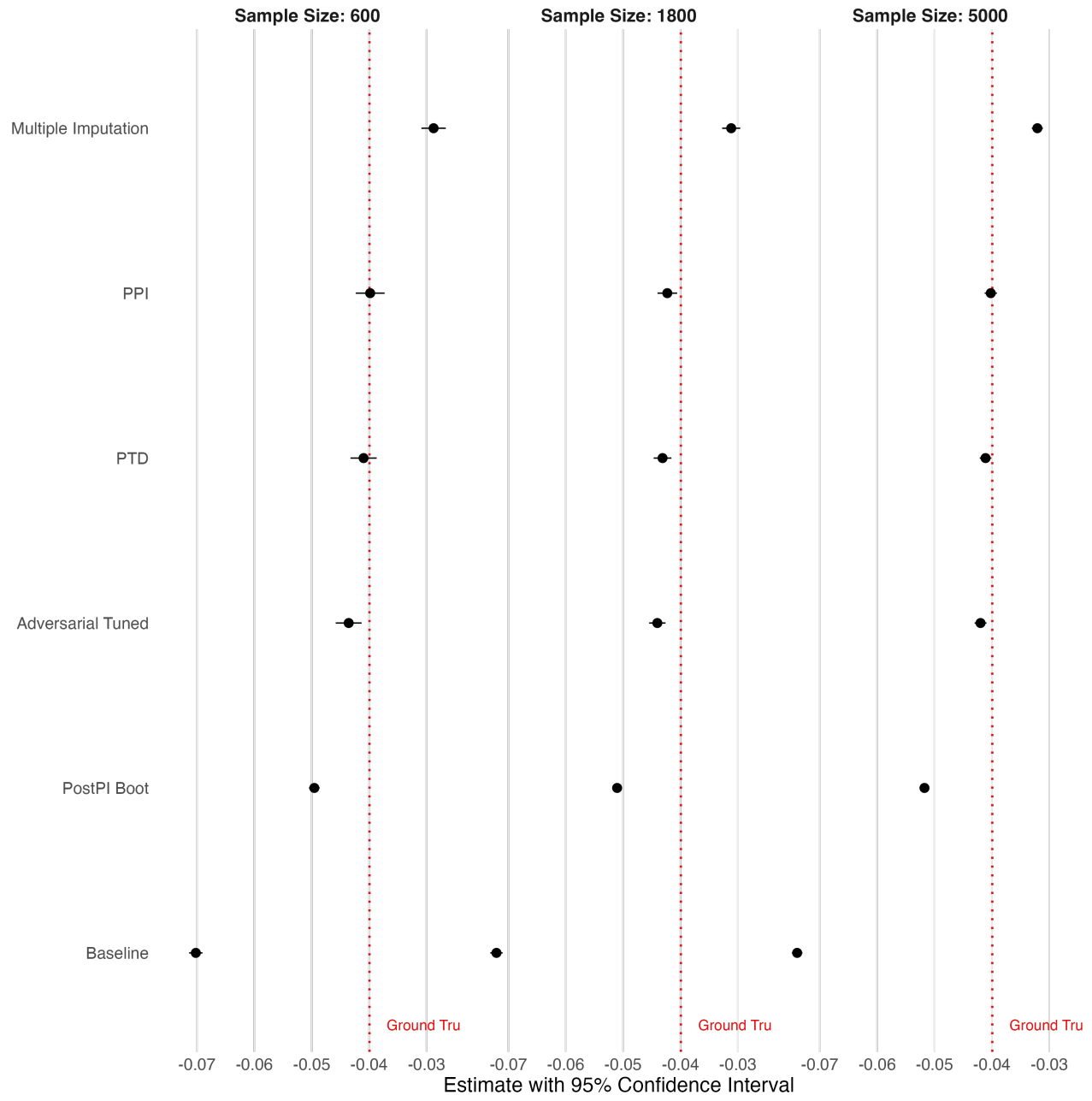


Figure 2: Estimates from various debiasing approaches across labeled sample sizes for the roads example. Figure 6 shows the distribution after 10,000 labeled samples.

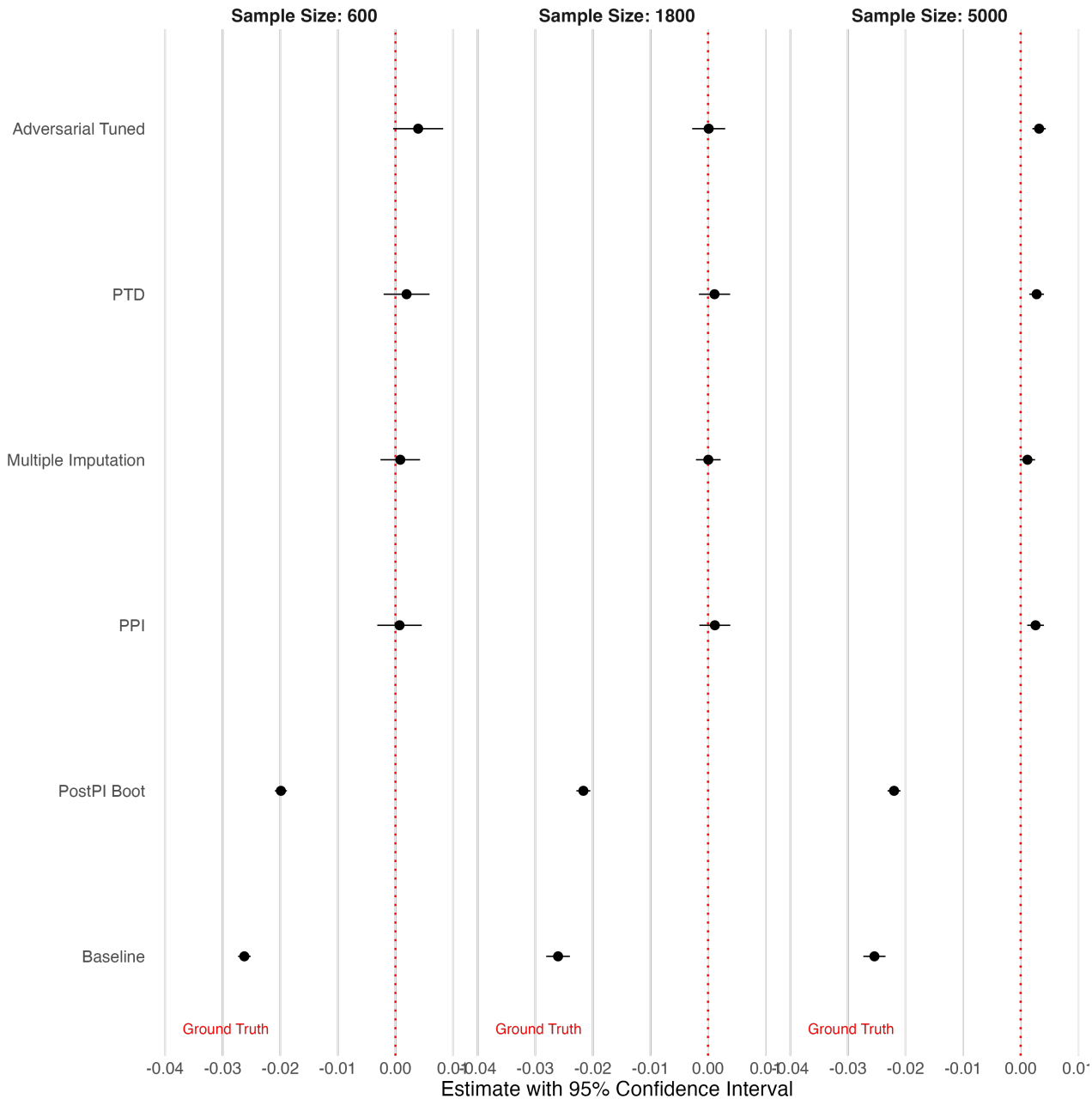


Figure 3: Estimates from various debiasing approaches across labeled sample sizes for the simulated example. Figure 3 shows the distribution after 10,000 labeled samples.

8 RESULTS FROM DIFFERENT ADVERSARIES

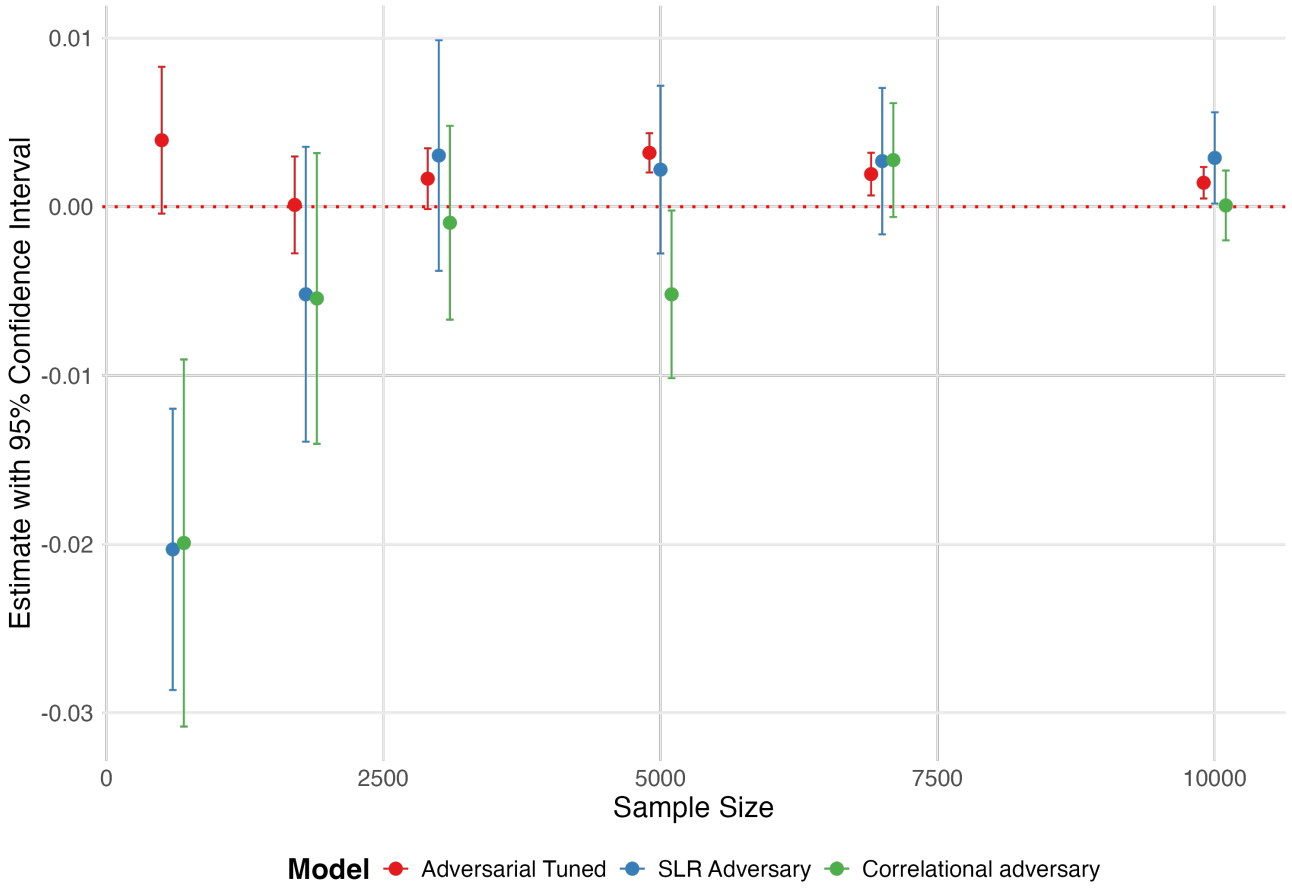


Figure 4: Estimates from various adversarial approaches across labeled sample sizes for the simulated example.

9 SENSITIVITY TO α TUNING PARAMETER

Table 1: Alpha-sensitivity results for the tuned adversarial measurement model (yhat_adv_corr) at labeled sample size 10,000. For each debiasing weight (0.5, 1.0, 2.0, 5.0), 50 independent replications were run with randomized data order/folds; within each weight, the tuning parameter λ was estimated from replication-level covariance/variance and used to form tuned estimates. The table reports the mean tuned estimate with a 95% confidence interval and the mean measurement-model MSE (unlabeled portion) with a 95% confidence interval.

Debias Weight	Mean Estimate	95% CI (Estimate)	Mean MSE	95% CI (MSE)	n
0.5	-0.0026	[-0.0036, -0.0017]	0.0679	[0.0631, 0.0726]	50
1.0	-0.0026	[-0.0035, -0.0016]	0.0732	[0.0682, 0.0783]	50
2.0	-0.0027	[-0.0037, -0.0018]	0.0682	[0.0650, 0.0714]	50
5.0	-0.0026	[-0.0036, -0.0016]	0.0700	[0.0664, 0.0737]	50

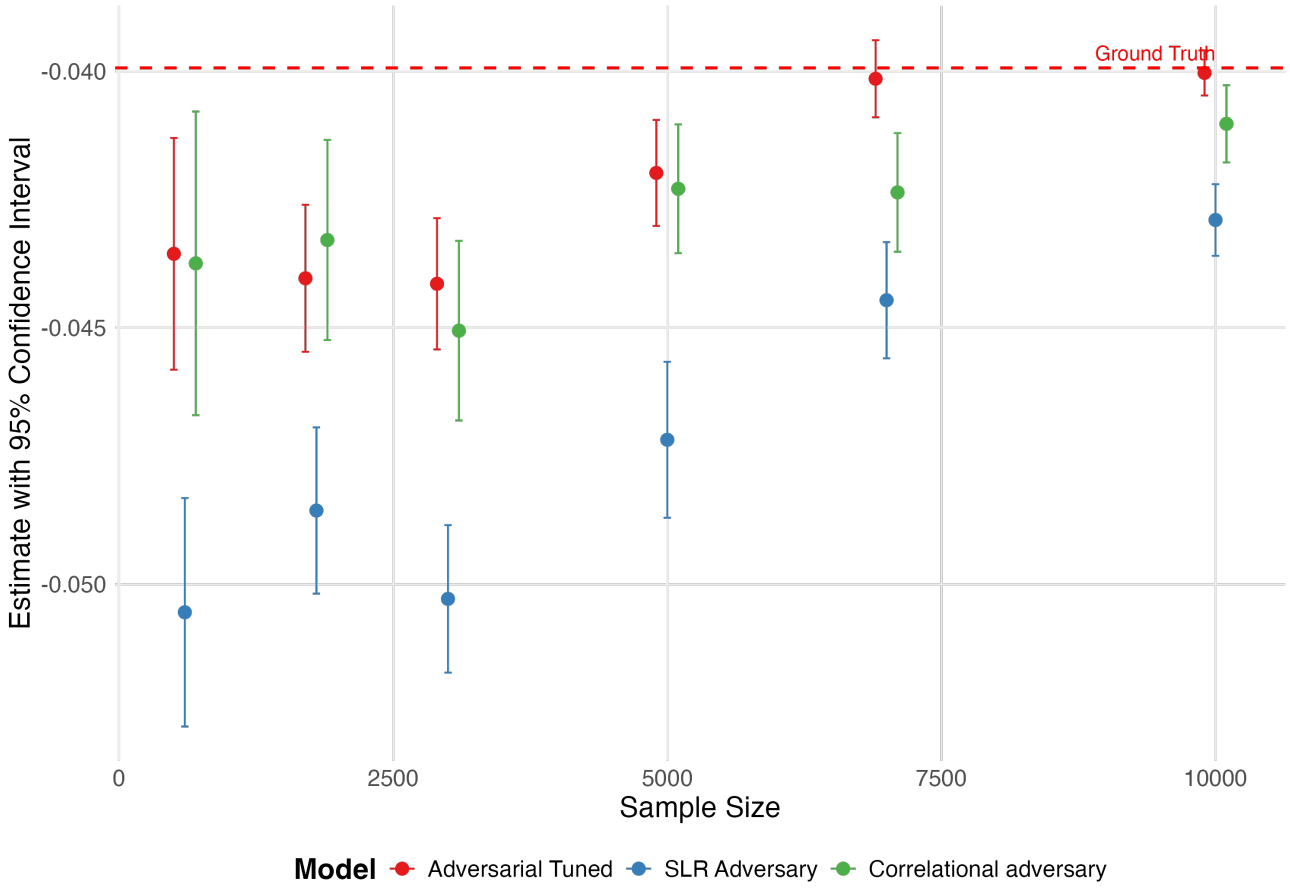


Figure 5: Estimates from various adversarial approaches across labeled sample sizes for the roads example.

Table 2: Alpha-sensitivity results for the tuned adversarial measurement model (yhat_adv_corr) on roads data at labeled sample size 10,000. For each debiasing weight (0.5, 1.0, 2.0, 5.0), 50 independent replications were run with randomized data order/folds; within each weight, the tuning parameter λ was estimated from replication-level covariance/variance and used to form tuned estimates. The table reports the mean tuned estimate with a 95% confidence interval and the mean measurement-model MSE (unlabeled portion) with a 95% confidence interval.

Debias Weight	Mean Estimate	95% CI (Estimate)	Mean MSE	95% CI (MSE)	n
0.5	-0.0403	[-0.0410, -0.0397]	0.1649	[0.1627, 0.1671]	50
1.0	-0.0401	[-0.0408, -0.0394]	0.1508	[0.1492, 0.1524]	50
2.0	-0.0402	[-0.0409, -0.0395]	0.1427	[0.1411, 0.1442]	50
5.0	-0.0401	[-0.0408, -0.0394]	0.1395	[0.1383, 0.1407]	50