Active Validation: Strategic Sampling for Improving the Efficiency of Two-Stage Debiasing

July 4, 2026

Abstract

This paper considers a two-stage regression problem using machine-learning-generated variables. In the first stage, machine learning is used to predict a variable of interest, which is then included as an explanatory variable in the second-stage regression. However, machine learning prediction errors may bias the estimator. Within a moment estimation framework, I develop a new method to mitigate this problem. Compared with existing methods, it is more stable, less costly, and easier to generalize. Since the use of this method requires researchers to have an additional validation budget, I further examine how, under a budget constraint, machine learning models can be used to identify which data points are most worth validating, that is, active validation, thereby reducing implementation costs. Even when the validation budget is very limited, the method performs remarkably well on a real-world dataset. Compared with uniform validation, it can save more than 40% of the validation sample size and shorten the confidence interval width by nearly one quarter.
Keywords: machine learning; prediction error; active validation; inferential efficiency

1 Introduction

In recent years, methods that use machine learning to extract unstructured variables have been increasingly applied in empirical research in economics, management, marketing, and information systems. Such studies usually have a two-stage structure. In the first stage, researchers use a small amount of manually labeled data to train a machine learning model and apply it to a larger unlabeled sample, thereby obtaining a machine-learning-generated variable $\widehat{X}_i$. In the second stage, researchers include $\widehat{X}_i$ in the regression model as a substitute for the true variable X_i, and estimate its relationship with the outcome variable Y_i. Since manually labeling the full sample is often costly, this practice of approximating full-sample labels using a small number of human labels is highly attractive in empirical research.

However, this practice also introduces a new problem: machine-learning-generated variables usually contain prediction errors, and these errors are not necessarily equivalent to classical measurement errors. In the classical setting, errors are often assumed to be random noise independent of both the true variable and the error term in the outcome equation. Machine learning prediction errors, however, are often systematic. For example, the model may be more likely to make errors for observations with more complex text, lower image quality, rarer sample groups, or more ambiguous features. If these sample characteristics also affect the outcome variable, prediction errors will be correlated with the regression error term, which in turn leads to systematic bias in the naive regression estimator obtained by directly using $\widehat{X}_i$.

Existing research has developed numerous methods to correct estimation bias caused by machine learning prediction errors. One major strand of the literature develops correction methods using validation data. The basic idea is to observe both the true outcome variable and the machine-learning-generated proxy variable in a smaller validation sample, and to use the conditional distribution or conditional moments of the true outcome variable relative to the proxy variable and other observable information revealed by the validation sample to correct

estimation bias in the main sample. Compared with estimators that use only the validation sample, these methods can exploit the sample-size advantage of the main sample and thereby improve estimation efficiency. However, existing methods still have several limitations. First, some methods rely on strong assumptions, which are often difficult to satisfy in many empirical applications. Second, although more general correction methods can relax some assumptions, they face difficulties such as high implementation costs when handling continuous-label problems or unstable estimators in real data.

How to achieve reliable statistical inference at low validation cost while fully exploiting the information contained in large-scale proxy variables in the main sample is the core issue of this paper. To this end, this paper proposes a method called "active validation." The method first constructs proxy moment conditions using the machine-learning-generated variables observable in the full sample. It then observes the true variable in a small validation sample and applies inverse probability weighted correction to the difference between the true moment condition and the proxy moment condition. Under an appropriate sampling design and regularity conditions, the corrected moment can recover the target moment condition, thereby correcting the estimation bias caused by machine learning prediction errors.

Furthermore, drawing on the idea of active statistical inference, this paper prioritizes, under a given validation budget, observations that are more likely to change the moment conditions and exert a larger influence on the target parameter estimate, based on information observable before validation. Compared with uniform validation, this method can substantially improve estimation precision, that is, it can obtain a smaller asymptotic variance and shorter confidence intervals under the same validation budget. From a practical perspective, this approach can substantially reduce the cost of debiasing.

This paper contributes to the existing literature in three main ways. First, it proposes an easy-to-implement moment correction method for correcting estimation bias caused by machine learning prediction errors entering a regression model. Our method does not require stringent assumptions; it only needs the difference between the true moment condition and the proxy moment condition to perform the correction, and it can fully exploit large-scale proxy-variable information in the main sample. Second, this paper emphasizes the selection strategy for validation samples. When moment contamination is heterogeneous across samples, compared with uniform validation, the proposed method can improve estimation precision and thereby save budget costs. Third, I apply the idea of active statistical inference to the debiasing problem under a moment estimation framework, thereby extending the scope of this method.

The remainder of the paper is organized as follows. Section 2 introduces the related literature. Section 3 establishes a unified moment-condition correction framework and proves the consistency and asymptotic normality of the corrected estimator. Section 4 derives the optimal validation probability under a given validation budget. Section 5 discusses specific applications of active validation in ordinary least squares, instrumental variables, difference-in-differences, and regression discontinuity. Section 6 uses a real-world dataset to validate the effectiveness of the proposed method.

2 Literature

This paper is first related to the literature on measurement error. Measurement error usually refers to the discrepancy between the variable observed by researchers and its true value. Under the classical measurement error framework, measurement error is typically assumed to be independent of the true variable and the regression error term. In this case, if the error-contaminated variable enters a linear regression as an explanatory variable, the ordinary least squares estimator usually suffers from attenuation bias toward zero. More general nonclassical measurement error allows measurement error to be correlated with the true variable or other covariates, so that the bias is no longer a simple attenuation form. Existing studies have sys-

tematically examined measurement error models and their correction methods; relevant reviews include Fuller (2009), Chen et al. (2011), and Schennach (2016). The two-stage bias introduced by machine-learning-generated variables is essentially a form of nonclassical measurement error, because unless the machine learning model achieves 100% predictive accuracy, even if the algorithm converges to the optimal prediction function, it cannot guarantee that the regression error term satisfies the conditional mean zero assumption when predicted variables are used in place of true explanatory variables.

An important way to mitigate estimation bias caused by machine learning prediction errors is to use auxiliary data, or validation samples, to correct measurement error in machine-learning-generated variables. In such settings, researchers usually observe only the machine-learning-generated proxy variable in the main sample, while observing both the proxy variable and its true value in a smaller validation sample. The auxiliary sample therefore provides information about the relationship between the true variable and the proxy variable, thereby correcting estimation bias. Compared with estimation using only the validation sample, such methods can exploit a large number of observations in the main sample and therefore have a significant sample-size advantage.

A few pioneering studies have specifically examined estimation bias caused by using machine learning predicted variables as substitutes for true variables. Yang et al. (2018), Fong and Tyler (2020), and Qiao and Huang (2021) propose corresponding bias-correction methods under different model settings. However, these methods usually rely on strong identifying assumptions, such as the "peripheral features have no effect" assumption, which states that unstructured data affect the outcome variable only through the focal feature of interest, while other peripheral features used by the machine learning model do not directly affect the outcome variable. This assumption may be difficult to satisfy in many practical applications, because machine learning models often use non-focal information such as background color, text style, image quality, or co-occurring objects to improve predictive accuracy, and these peripheral features themselves may also directly affect the outcome variable.

To relax the "peripheral features have no effect" assumption, recent research has begun to focus on more general nonclassical measurement error settings, in which machine learning prediction errors are allowed to be correlated with the regression error term. This problem is also closely related to the literature on correcting measurement error using auxiliary data, with Chen et al. (2005) being a representative contribution. That paper allows arbitrary correlation between measurement error and the true variable, but usually relies on a key transportability condition: conditional on the proxy variable, the conditional distribution of the true variable is the same in the main sample and the auxiliary sample. If the auxiliary sample comes from a different data environment, labeling rule, or sample selection mechanism, this condition may be difficult to satisfy, thereby affecting the consistency of the corrected estimator. Two recent papers further develop this approach. Wei and Malik (2026) train $q(w_i \mid \widehat{w}_i, x_i, y_i)$ and approximate the oracle estimation criterion by weighting the latent true label w_i in the MLE function according to its conditional probability, thereby correcting bias without imposing the "peripheral features have no effect" assumption. However, when w_i is continuous, the implementation cost becomes high. Allon et al. (2023) generalize the method of Chen et al. (2005) to more model settings, but the corrected estimator is unstable in real-world data.

Inspired by the above studies, this paper proposes a new moment correction method based on the idea of using auxiliary data to correct estimation bias caused by prediction errors. This method does not require a transportability assumption, is easier to implement, and performs stably in real data.

In addition to developing a new debiasing method, we also design optimal validation probabilities to improve the efficiency with which the validation budget is used, an issue not addressed in the above literature. We focus on validation-sample efficiency for two reasons. First, the method proposed in this paper requires a certain amount of additional validation budget. Sec-

ond, both existing studies and the real-data simulations in this paper show that as the validation budget increases, the performance of the correction method improves dramatically. Researchers therefore naturally care about whether a more efficient sample selection mechanism can deliver better estimation performance than uniform validation under a lower validation budget.

Motivated by this issue, this paper draws on the idea of the active statistical inference literature [10]. Its basic intuition is that validation resources should be allocated with priority to observations with higher prediction uncertainty; for samples where model predictions are relatively reliable, prediction information can be fully used. In this way, active statistical inference methods can obtain inferential precision close to that of large-scale labeled data using fewer true labels. However, the existing active statistical inference literature is mainly based on the convex M-estimation framework and focuses on how to design validation probabilities to reduce estimator variance when the outcome variable Y_i is missing. This paper differs from it in two respects. First, unlike the problem studied in the existing active statistical inference literature, we focus on how to use $\widehat{X}_i$ to construct corrected statistics and restore consistency when X_i is missing, and on which X_i's should be validated to reduce the variance of the corrected estimator. Second, this paper is not limited to M-estimation, but constructs a unified active validation correction framework starting from moment conditions. As a result, the method in this paper not only covers problems that can be written as M-estimation, such as OLS, but can also handle broader econometric identification designs such as IV, GMM, DiD, and RD. In this way, this paper extends the idea of active statistical inference to the regression debiasing problem in which machine learning predicted variables are used as explanatory variables.

3 A Unified Moment-Condition Correction Framework

This section presents the general theoretical framework of the method. We use an OLS problem to introduce the intuition of the proposed method and then extend it to a moment framework, thereby proving the desirable statistical properties of "active validation."

3.1 Starting from an Example

Consider a univariate linear regression model. To see more clearly where the bias comes from, temporarily ignore the intercept term, or equivalently, interpret both Y_i and X_i as centered variables after removing the intercept. The model can then be written as

$$Y_i = \beta_1 X_i + \varepsilon_i,$$

and the oracle moment condition corresponding to the true variable X_i is

$$E[X_i(Y_i - \beta_1 X_i)] = 0.$$

Now suppose that we cannot observe the true X_i, and can only observe the predicted variable $\widehat{X}_i$ generated by the machine learning model. Let the prediction error be $e_i = \widehat{X}_i - X_i$. If we directly replace X_i with $\widehat{X}_i$, the naive estimator actually solves

$$E[\widehat{X}_i(Y_i - \beta_1 \widehat{X}_i)] = 0.$$

To understand why this moment condition generally no longer holds, substitute the true model $Y_i = \beta_1 X_i + \varepsilon_i$ into it:

$$E[\widehat{X}_i(Y_i - \beta_1 \widehat{X}_i)] = E[(X_i + e_i)\{Y_i - \beta_1(X_i + e_i)\}]$$

$$= E[(X_i + e_i)(\varepsilon_i - \beta_1 e_i)] = E[X_i\varepsilon_i] + E[e_i\varepsilon_i] - \beta_1 E[X_i e_i] - \beta_1 E[e_i^2].$$

Since the oracle moment condition guarantees $E[X_i\varepsilon_i] = 0$, the expression becomes

$$E[\widehat{X}_i(Y_i - \beta_1 \widehat{X}_i)] = E[e_i\varepsilon_i] - \beta_1 E[X_i e_i] - \beta_1 E[e_i^2].$$

This expression clearly illustrates the source of bias in the naive estimator. Even if the prediction error e_i is uncorrelated with both the true explanatory variable X_i and the error term ε_i, that is, $E[e_i\varepsilon_i] = 0$ and $E[X_i e_i] = 0$, the last term $-\beta_1 E[e_i^2]$ is still generally nonzero. In other words, as long as the machine learning predicted variable $\widehat{X}_i$ contains prediction error, the naive moment condition is typically not equal to zero at the true parameter.

In the most classical case, if e_i is a mean-zero prediction error uncorrelated with both X_i and ε_i, then the population slope of the naive regression can be written as

$$\beta_1^{naive} = \frac{E[\widehat{X}_i Y_i]}{E[\widehat{X}_i^2]} = \frac{\beta_1 E[X_i^2]}{E[X_i^2] + E[e_i^2]}.$$

Since $E[X_i^2] + E[e_i^2] > E[X_i^2]$, the useful covariance signal in the numerator mainly comes from X_i, while the denominator additionally includes the useless variation induced by the prediction error e_i. Therefore, the coefficient shrinks toward zero. However, if the explanatory variable is generated by machine learning, the prediction error e_i need not be independent of X_i or ε_i. For example, the machine learning model may systematically overestimate or underestimate X_i for certain types of samples, or it may perform worse near samples with higher or lower values of the outcome variable Y_i. In such cases, $E[e_i\varepsilon_i]$ and $E[X_i e_i]$ are nonzero, and the direction of the estimator bias is arbitrary.

This simple example illustrates the basic starting point of this paper: whether the estimator is biased depends on whether the moment condition continues to hold after using $\widehat{X}_i$. Therefore, what we truly need to focus on is the contamination of the moment condition caused by machine learning prediction error:

$$d_i^{OLS}(\beta) = X_i(Y_i - \beta X_i) - \widehat{X}_i(Y_i - \beta \widehat{X}_i).$$

If all X_i's were observable, then the moment contamination of each sample would be observed, and correcting the estimation would be straightforward. However, validating X_i is often costly, so we cannot obtain the true values of all X_i's. A natural idea is to validate only part of the sample and use this part to correct the moment-condition contamination caused by the machine-learning-generated variable. Specifically, let $\xi_i \in \{0, 1\}$ indicate whether sample i is validated, and let $\pi_i = P(\xi_i = 1)$ denote its validation probability. Construct the following corrected moment condition:

$$g_i^{AI}(\beta) = \widehat{X}_i(Y_i - \beta \widehat{X}_i) + \frac{\xi_i}{\pi_i}\left[X_i(Y_i - \beta X_i) - \widehat{X}_i(Y_i - \beta \widehat{X}_i)\right].$$

The intuition of this expression is as follows. The first term uses the naive moment condition for all samples. The second term appears only for validated samples, and compensates for the moment-condition contamination caused by the machine learning variable by computing the difference between the true moment condition and the naive moment condition. Since validated samples are only a subset of all observations, they need to be multiplied by the inverse probability weight $1/\pi_i$, so that these validated samples can represent the population samples to which they correspond. It is easy to prove that $E[g_i(\theta_0; X_i)] = E[g_i^{AI}(\theta_0)]$. The detailed proof is given in Section 3.2.

Not all samples have the same validation value. How, then, should π_i be chosen? A "good" π_i can improve the efficiency of the budget, thereby yielding a more efficient estimator with smaller variance or shorter confidence intervals under the same validation budget; detailed proofs are given in Sections 3.4 and 4.2. A simple intuition is that we should label X_i with higher probability for points with larger moment contamination. This requires π_i to reflect the extent to which points with features $\widehat{X}_i$ contaminate the moment condition. A large π_i indicates that points with features $\widehat{X}_i$ are likely to have large contamination of the moment condition; conversely, a small π_i indicates small contamination. The choice of π_i will be discussed in detail in Section 4.1.

3.2 Unbiasedness of the Corrected Moment Condition

Next, we extend the intuition in Section 3.1 to a more general moment framework and provide a formal proof. Let

$$\xi_i = \begin{cases} 1, & \text{sample } i \text{ is validated and the true } X_i \text{ is observed,} \\ 0, & \text{sample } i \text{ is not validated.} \end{cases}$$

Let $\xi_i \sim Bern(\pi(\widehat{X}_i))$. The validation probability is $P(\xi_i = 1 \mid \widehat{X}_i) = \pi_i$, with $0 < \pi_i \leq 1$, representing the probability that an observation is validated conditional on the predicted value of the covariate. Define the true moment function and the substitute moment function as $g_i(\theta; X_i)$ and $g_i(\theta; \widehat{X}_i)$, respectively. Their difference is $d_i(\theta) = g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)$, which represents the contamination of the moment condition caused by the machine-learning-generated variable. Since X_i can be observed only in the validation sample, $d_i(\theta)$ can be computed only when $\xi_i = 1$. This paper constructs the following corrected moment function:

$$g_i^{AI}(\theta) = g_i(\theta; \widehat{X}_i) + \frac{\xi_i}{\pi_i}\left[g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)\right].$$

Equivalently,

$$g_i^{AI}(\theta) = g_i(\theta; X_i) + \left(1 - \frac{\xi_i}{\pi_i}\right) d_i(\theta),$$

or

$$g_i^{AI}(\theta) = g_i(\theta; \widehat{X}_i) + \frac{\xi_i}{\pi_i} d_i(\theta).$$

This correction function essentially uses the difference between the true moment and the proxy moment in the validated sample to perform inverse probability weighted correction. Since the validation probability is π_i, each validated sample represents approximately $1/\pi_i$ similar samples' information about moment contamination. It is straightforward to show that this corrected moment function is unbiased.

Proposition 1: Unbiasedness of the corrected moment condition. Suppose that the validation decision satisfies $P(\xi_i = 1 \mid \widehat{X}_i) = \pi_i$, with $0 < \pi_i \leq 1$. Then, for any θ,

$$E[g_i^{AI}(\theta)] = E[g_i(\theta; X_i)].$$

Consequently, if the true parameter θ_0 satisfies $E[g_i(\theta_0; X_i)] = 0$, then the corrected moment condition also satisfies $E[g_i^{AI}(\theta_0)] = 0$.

Proof. By definition, $g_i^{AI}(\theta) = g_i(\theta; \widehat{X}_i) + \frac{\xi_i}{\pi_i}\left[g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)\right]$. Here, the validation probability is specified to depend only on the machine learning predicted value $\widehat{X}_i$ of the main explanatory variable, that is, $\pi_i = \pi(\widehat{X}_i)$. Conditional on $(X_i, \widehat{X}_i)$, $g_i(\theta; X_i)$, $g_i(\theta; \widehat{X}_i)$, and π_i are all fixed. Therefore, $E\left[\frac{\xi_i}{\pi_i}\{g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)\} \mid X_i, \widehat{X}_i\right] = \frac{g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)}{\pi_i} E[\xi_i \mid X_i, \widehat{X}_i]$. Since the validation probability depends only on $\widehat{X}_i$, $E[\xi_i \mid X_i, \widehat{X}_i] = \pi_i$. Hence, $E\left[\frac{\xi_i}{\pi_i}\{g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)\} \mid X_i, \widehat{X}_i\right] = g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)$. Thus, $E[g_i^{AI}(\theta) \mid X_i, \widehat{X}_i] = g_i(\theta; \widehat{X}_i) + g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i) = g_i(\theta; X_i)$. Taking unconditional expectations gives $E[g_i^{AI}(\theta)] = E[g_i(\theta; X_i)]$. When $\theta = \theta_0$, the true moment condition $E[g_i(\theta_0; X_i)] = 0$ implies $E[g_i^{AI}(\theta_0)] = 0$. This completes the proof.

In practice, the validation probability usually need not depend only on $\widehat{X}_i$. It may depend on a richer information set A_i observable before validation. Therefore, we can write the validation probability as $\pi_i = \pi(A_i)$, which does not change the main structure of the proof.

3.3 Consistency and Asymptotic Normality of the Corrected GMM Estimator

Based on the corrected moment function, define the sample moment: $\widehat{m}_n^{AI}(\theta) = \frac{1}{n}\sum_{i=1}^n g_i^{AI}(\theta)$. Given a positive definite weighting matrix W_n, define the corrected GMM estimator:

$$\widehat{\theta}^{AI} = \arg\min_{\theta\in\Theta} \widehat{m}_n^{AI}(\theta)' W_n \widehat{m}_n^{AI}(\theta).$$

The corresponding population objective function is

$$Q^{AI}(\theta) = m^{AI}(\theta)' W m^{AI}(\theta).$$

Since Proposition 1 shows that $m(\theta) = E[g_i(\theta; X_i)] = E[g_i^{AI}(\theta)] = m^{AI}(\theta)$, we have $Q^{AI}(\theta) = m^{AI}(\theta)' W m^{AI}(\theta) = m(\theta)' W m(\theta) = Q(\theta)$. That is, the population objective function of the corrected GMM coincides with the GMM population objective function using the true X_i.

Under the following four assumptions, we can prove the consistency of the corrected GMM estimator.

Assumption 1. There exists a unique $\theta_0 \in \Theta$ such that $E[g_i(\theta_0; X_i)] = 0$. Moreover, for any $\epsilon > 0$, $\inf_{\|\theta-\theta_0\|>\epsilon} m(\theta)' W m(\theta) > 0$.

Assumption 2. The validation indicator satisfies $P(\xi_i = 1 \mid \widehat{X}_i) = \pi_i$, and there exists a lower bound $\pi_n > 0$, possibly varying with the sample size, such that $\pi_i \geq \pi_n > 0$. At the same time, the number of validation samples satisfies $n_b = \sum_{i=1}^n E[\xi_i] \to \infty$. This assumption allows $n_b/n \to 0$, that is, the validation-sample proportion may converge to zero, but the number of validation samples itself must diverge.

Assumption 3. The corrected moment function satisfies a uniform law of large numbers: $\sup_{\theta\in\Theta} \|\widehat{m}_n^{AI}(\theta) - m(\theta)\| \xrightarrow{p} 0$. A sufficient condition is that $g_i^{AI}(\theta)$ is continuous in θ, its magnitude is controlled by an integrable envelope function, and the IPW weights do not explode excessively, that is, $E\left[\sup_{\theta\in\Theta}\left\|\frac{\xi_i}{\pi_i}\{g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)\}\right\|\right] < \infty$, or that the corresponding triangular-array law of large numbers conditions hold.

Assumption 4. $W_n \xrightarrow{p} W$, where W is a positive definite matrix.

Theorem 1: Consistency of the corrected GMM estimator. Under Assumptions 1–4,

$$\widehat{\theta}^{AI} \xrightarrow{p} \theta_0.$$

Proof. By Assumption 3, $\sup_{\theta\in\Theta} \|\widehat{m}_n^{AI}(\theta) - m(\theta)\| \xrightarrow{p} 0$. Together with Assumption 4, $W_n \xrightarrow{p} W$. Hence the sample objective function $\widehat{Q}_n^{AI}(\theta) = \widehat{m}_n^{AI}(\theta)' W_n \widehat{m}_n^{AI}(\theta)$ converges uniformly to the population objective function $Q(\theta) = m(\theta)' W m(\theta)$, that is, $\sup_{\theta\in\Theta} |\widehat{Q}_n^{AI}(\theta) - Q(\theta)| \xrightarrow{p} 0$. By the identification condition, $Q(\theta)$ has a unique minimum at θ_0. It follows that $\widehat{\theta}^{AI} \xrightarrow{p} \theta_0$. This completes the proof.

Next, we prove asymptotic normality. Let $G = \frac{\partial m(\theta)}{\partial \theta'}\Big|_{\theta=\theta_0} = E\left[\frac{\partial g_i(\theta; X_i)}{\partial \theta'}\Big|_{\theta=\theta_0}\right]$. Since the corrected moment condition satisfies $E[g_i^{AI}(\theta)] = E[g_i(\theta; X_i)]$, under appropriate regularity conditions its derivative also satisfies $E\left[\frac{\partial g_i^{AI}(\theta)}{\partial \theta'}\Big|_{\theta=\theta_0}\right] = G$. Define the variance matrix of the corrected moment function at the true parameter as

$$\Omega^{AI} = Var(g_i^{AI}(\theta_0)).$$

By the preceding definition, $g_i^{AI}(\theta_0) = g_i(\theta_0; \widehat{X}_i) + \frac{\xi_i}{\pi_i}\left[g_i(\theta_0; X_i) - g_i(\theta_0; \widehat{X}_i)\right]$, which can also be written as

$$g_i^{AI}(\theta_0) = g_i(\theta_0; X_i) + \left(\frac{\xi_i}{\pi_i} - 1\right) d_i(\theta_0).$$

This shows that the corrected moment function consists of two parts. The first part is the ideal moment function using the true X_i. The second part is the IPW sampling noise generated by validating only a subset of samples. Under conditionally independent sampling, the additional variance term is approximately

$$E\left[d_i(\theta_0)d_i(\theta_0)'\left(\frac{1}{\pi_i}-1\right)\right].$$

Therefore,

$$\Omega^{AI} = Var(g_i(\theta_0; X_i)) + E\left[d_i(\theta_0)d_i(\theta_0)'\left(\frac{1}{\pi_i}-1\right)\right] + \text{cross terms},$$

where the cross terms are zero under several common conditions, for example, when validation sampling is independent of the structural error and the correction error has conditional mean zero. Even if the cross terms are nonzero, Ω^{AI} can be directly estimated using the sample covariance matrix of the corrected moment functions. Then $\widehat{\Omega}^{AI}$ can be plugged into a sandwich-type variance formula to obtain an asymptotic variance estimator for $\widehat{\theta}^{AI}$.

Assumption 5: Local smoothness. In a neighborhood of θ_0, $g_i^{AI}(\theta)$ is differentiable in θ, and

$$\sup_{\theta\in N(\theta_0)}\left\|\frac{\partial g_i^{AI}(\theta)}{\partial\theta'}\right\|$$

satisfies appropriate integrability conditions. In addition, $G'WG$ is nonsingular.

Assumption 6: Central limit theorem. The corrected moment function satisfies

$$\sqrt{n}\,\widehat{m}_n^{AI}(\theta_0) = \frac{1}{\sqrt{n}}\sum_{i=1}^{n} g_i^{AI}(\theta_0) \xrightarrow{d} N(0, \Omega^{AI}).$$

When $\pi_i \asymp n_b/n$ and $n_b/n \to 0$, the IPW term may dominate the variance, and the natural effective convergence rate is usually $\sqrt{n_b}$. In this case, equivalently, one can write

$$\sqrt{n_b}\,\widehat{m}_n^{AI}(\theta_0) \xrightarrow{d} N(0, \Omega_b^{AI}),$$

where Ω_b^{AI} is the variance matrix rescaled by the effective size of the validation sample. For a unified notation, the following uses the $\sqrt{n}$ form and allows Ω^{AI} to vary with n through the validation probabilities.

Theorem 2: Asymptotic normality of the corrected GMM estimator. Under Assumptions 1–6,

$$\sqrt{n}(\widehat{\theta}^{AI} - \theta_0) \xrightarrow{d} N(0, V^{AI}),$$

where

$$V^{AI} = (G'WG)^{-1}G'W\Omega^{AI}WG(G'WG)^{-1}.$$

If the optimal GMM weighting matrix

$$W = (\Omega^{AI})^{-1}$$

is used, then

$$V^{AI} = (G'(\Omega^{AI})^{-1}G)^{-1}.$$

Proof. The first-order condition of the corrected GMM estimator is $\frac{\partial\widehat{m}_n^{AI}(\widehat{\theta}^{AI})'}{\partial\theta}W_n\widehat{m}_n^{AI}(\widehat{\theta}^{AI}) = 0$. Expanding $\widehat{m}_n^{AI}(\widehat{\theta}^{AI})$ around θ_0 gives $\widehat{m}_n^{AI}(\widehat{\theta}^{AI}) = \widehat{m}_n^{AI}(\theta_0) + G(\widehat{\theta}^{AI} - \theta_0) + o_p(\|\widehat{\theta}^{AI} - \theta_0\|)$. Substituting this into the first-order condition and ignoring higher-order terms yields $0 = G'W\left[\widehat{m}_n^{AI}(\theta_0) + G(\widehat{\theta}^{AI} - \theta_0)\right] + o_p(n^{-1/2})$. Rearranging, $G'WG(\widehat{\theta}^{AI} - \theta_0) = -G'W\widehat{m}_n^{AI}(\theta_0) +$

$o_p(n^{-1/2})$. Therefore, $\sqrt{n}(\widehat{\theta}^{AI} - \theta_0) = -(G'WG)^{-1}G'W\sqrt{n}\,\widehat{m}_n^{AI}(\theta_0) + o_p(1)$. By Assumption 6, $\sqrt{n}\,\widehat{m}_n^{AI}(\theta_0) \xrightarrow{d} N(0, \Omega^{AI})$. By Slutsky's theorem, $\sqrt{n}(\widehat{\theta}^{AI} - \theta_0) \xrightarrow{d} N(0, V^{AI})$, where $V^{AI} = (G'WG)^{-1}G'W\Omega^{AI}WG(G'WG)^{-1}$. This completes the proof.

In practice, the asymptotic variance can be estimated from the sample. Let $\widehat{g}_i^{AI} = g_i^{AI}(\widehat{\theta}^{AI})$. The sample variance matrix is $\widehat{\Omega}^{AI} = \frac{1}{n}\sum_{i=1}^n \widehat{g}_i^{AI}\widehat{g}_i^{AI\prime}$. If demeaning is needed, one can also use

$$\widehat{\Omega}^{AI} = \frac{1}{n}\sum_{i=1}^{n}\left(\widehat{g}_i^{AI} - \widehat{m}_n^{AI}(\widehat{\theta}^{AI})\right)\left(\widehat{g}_i^{AI} - \widehat{m}_n^{AI}(\widehat{\theta}^{AI})\right)'.$$

The derivative matrix is estimated as $\widehat{G} = \frac{1}{n}\sum_{i=1}^n \left.\frac{\partial g_i^{AI}(\theta)}{\partial \theta'}\right|_{\theta=\widehat{\theta}^{AI}}$. The parameter variance estimator is then

$$\widehat{V}^{AI} = (\widehat{G}'W_n\widehat{G})^{-1}\widehat{G}'W_n\widehat{\Omega}^{AI}W_n\widehat{G}(\widehat{G}'W_n\widehat{G})^{-1}.$$

Thus, the standard error of the j-th parameter is $\widehat{se}(\widehat{\theta}_j^{AI}) = \sqrt{\widehat{V}_{jj}^{AI}/n}$, and the corresponding $1-\alpha$ confidence interval is $\widehat{\theta}_j^{AI} \pm z_{1-\alpha/2}\widehat{se}(\widehat{\theta}_j^{AI})$.

3.4 Variance Comparison

To clearly illustrate the efficiency advantage of the proposed method, we need to conduct a variance analysis of the corrected estimator. In this section, we compare only the variance of the corrected moment-condition estimator with that of the estimator using only the validation sample. The variance performance of corrected moment estimators under two different validation rules, uniform sampling and strategic sampling, is related to the choice of validation probabilities and will be discussed in Section 4.

Define the estimator that uses only the validation sample:

$$\widehat{\theta}^{V} = \arg\min_{\theta}\left[\frac{1}{n_b}\sum_{i:\xi_i=1} g_i(\theta; X_i)\right]' W \left[\frac{1}{n_b}\sum_{i:\xi_i=1} g_i(\theta; X_i)\right].$$

If the validation set is uniformly randomly sampled and $n_b \to \infty$, then this estimator is consistent. However, it uses only n_b true samples, so its variance is approximately

$$\frac{1}{n_b}V^{V} = \frac{1}{n_b}(G'WG)^{-1}G'W\Omega_D WG(G'WG)^{-1},$$

where

$$\Omega_D = Var(g_i(\theta_0; X_i)).$$

Recall that the variance of the active inference corrected estimator defined above is

$$\frac{1}{n}V^{AI} = \frac{1}{n}(G'WG)^{-1}G'W\Omega^{AI}WG(G'WG)^{-1},$$

where

$$\Omega^{AI}_{\text{active}} = Var(g_i(\theta_0; X_i)) + E\left[\left(\frac{1}{\pi_i} - 1\right)d_i(\theta_0)d_i(\theta_0)'\right].$$

It can be seen that its asymptotic variance differs from that of the validation-only estimator in two core respects. First, the two estimators use different sample sizes: the validation-only estimator relies only on n_b validated samples, whereas the corrected estimator constructs corrected moment conditions on n samples. Second, the corrected estimator introduces an additional IPW correction variance term, namely the sampling fluctuation generated by magnifying the contamination $d_i(\theta) = g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)$ through ξ_i/π_i. In the above expression, this appears as

$$E\left[\left(\frac{1}{\pi_i} - 1\right)d_i(\theta_0)d_i(\theta_0)'\right],$$

which may amplify estimator instability. We next discuss whether this problem is serious and how it can be mitigated.

If the machine-learning-generated variable is sufficiently accurate in terms of the second-stage moment condition, that is, $g_i(\theta_0; \widehat{X}_i) \approx g_i(\theta_0; X_i)$, then the moment contamination term $d_i(\theta_0) = g_i(\theta_0; X_i) - g_i(\theta_0; \widehat{X}_i)$ is close to zero. In this case, the variance of the corrected moment estimator is approximately the variance when using the full-sample oracle moment, that is, close to $(1/n)V^{AI}$, which is usually much smaller than the variance $(1/n_b)V^V$ when using only the validation sample, because $n_b \ll n$. Of course, machine-learning-generated variables cannot be completely accurate for all samples. Nevertheless, we can choose validation probabilities π_i so that when a sample has small moment contamination, that is, $d_i(\theta_0) \approx 0$, it is assigned a lower validation probability; when a sample may generate large moment contamination, it is assigned a higher validation probability. This can keep the additional IPW variance term $d_i(\theta_0)d_i(\theta_0)'\left(\pi_i^{-1} - 1\right)$ as small as possible under the budget constraint, thereby keeping the additional variance induced by sampling at a low level. In particular, for samples with no moment contamination at all, that is, $g_i(\theta_0; \widehat{X}_i) = g_i(\theta_0; X_i)$, one could ideally set $\pi_i = 0$, because validating such samples does not reduce the additional variance of the corrected moment.

More intuitively, the variance of the validation-only estimator mainly comes from the sample-size loss caused by retaining only n_b true observations. By contrast, the active inference corrected estimator uses prediction information from all n samples and only applies IPW correction to the contamination term between the predicted moment and the true moment. Therefore, although the corrected estimator introduces additional IPW sampling variance, in real settings $n_b \ll n$, and validation-only estimation discards a large amount of useful information contained in the unvalidated samples. Thus, as long as the additional IPW variance is smaller than the information loss caused by the reduced sample size, the overall variance of the corrected estimator can be substantially smaller than that of the validation-only estimator.

Is the condition that "the additional IPW variance is smaller than the information loss caused by the reduced sample size" easy to satisfy? Intuitively, the difference $d_i(\theta_0)$ between the true moment condition and the proxy moment condition usually concentrates in a subset of observations with larger prediction errors or more severe moment contamination. In this case, compared with estimation using only a small number of validation samples, the corrected estimator can preserve the information provided by a large number of low-contamination observations in the full sample, and only needs validation samples to correct relatively limited moment contamination. Therefore, as long as IPW weights are appropriately controlled, the additional sampling variance introduced by the corrected estimator may be smaller than the sample-size loss caused by using only the validation sample. The experimental results in Sections 6 and 7 show that when the validation budget is not too low, existing validation samples already provide enough training information for the risk function, allowing it to predict the intensity of moment contamination to some extent. In this case, the active validation rule can allocate more validation resources to samples with larger expected moment contamination and fewer resources to samples for which the difference between the proxy moment and the true moment is small, thereby effectively reducing the additional variance of the IPW correction term. Therefore, in the real-data environment of this paper, as long as the validation budget is sufficient to support basic training of the risk function, the active corrected estimator exhibits higher statistical efficiency than both the validation-only estimator and the uniform validation correction estimator. At the same time, by mixing active validation probabilities with uniform validation probabilities, as introduced in detail in Section 4.4, this paper further avoids extreme inverse probability weights caused by excessively small validation probabilities for some samples.

4 Validation Probability

In this section, we seek the optimal validation probability that minimizes the variance of the estimator. Furthermore, to control possible variance instability in practice, we also explore more robust sampling rules.

Although what we should ultimately care about is the variance of the estimator, first examining the additional variance of the corrected moment provides intuition for choosing validation probabilities. According to the analysis in the previous section, given the complete data and pre-validation information, the additional fluctuation introduced by validation randomness in the correction term can be written as $\left(\frac{\xi_i}{\pi_i} - 1\right) d_i(\theta)$. Its conditional variance is $Var\left[\left(\frac{\xi_i}{\pi_i} - 1\right) d_i(\theta) \mid X_i, \widehat{X}_i\right] = d_i(\theta) d_i(\theta)' \left(\frac{1}{\pi_i} - 1\right)$. In the scalar case, the additional variance of a point can be written as

$$Var\left[\left(\frac{\xi_i}{\pi_i} - 1\right) d_i(\theta) \mid X_i, \widehat{X}_i\right] = d_i(\theta)^2 \left(\frac{1}{\pi_i} - 1\right).$$

When $d_i(\theta)$ is large, a small π_i will substantially amplify the variance of the correction term; when $d_i(\theta)$ is small, even if the validation probability of this sample is low, its contribution to the overall variance is relatively limited. Therefore, under a given validation budget, a better validation rule should assign higher validation probabilities to samples with larger expected moment contamination and lower validation probabilities to samples with smaller expected moment contamination. If validation samples can be selected according to this rule, then the final variance of interest,

$$E\left[Var\left[\left(\frac{\xi_i}{\pi_i} - 1\right) d_i(\theta) \mid X_i, \widehat{X}_i\right]\right],$$

can be kept at a lower level.

4.1 Optimal Validation Probability

Although the previous discussion has focused on the additional variance of the corrected moment, the final object of interest should be the variance of the estimator. We first analyze how moment contamination is transmitted to the parameter estimator, and then analyze how to choose validation probabilities to minimize the variance of the estimator. Suppose there are q moment conditions and p parameters, that is, $g_i(\theta) \in \mathbb{R}^q$ and $\theta \in \mathbb{R}^p$. At the true parameter θ_0, define the moment-condition contamination caused by prediction error as

$$d_i(\theta_0) = g_i(\theta_0; X_i) - g_i(\theta_0; \widehat{X}_i) \in \mathbb{R}^q.$$

$d_i(\theta_0)$ is a q-dimensional vector, representing the contamination caused by sample i along different moment-condition directions. However, researchers usually care not about the moment conditions themselves but about the parameter estimator. Therefore, it is necessary to further explain how $d_i(\theta_0)$ is transmitted to the target estimator.

Let the sample average moment be $\widehat{m}_n(\theta_0) = n^{-1} \sum_{i=1}^n g_i(\theta_0)$. Under a GMM first-order expansion,

$$\widehat{\theta} - \theta_0 \approx -M \widehat{m}_n(\theta_0), \qquad M = (G'WG)^{-1} G'W.$$

The error in the sample average moment does not directly equal the parameter error. Rather, it first enters $\widehat{m}_n(\theta_0)$, and is then transmitted to $\widehat{\theta} - \theta_0$ through the matrix M. The matrix M maps moment-condition contamination into the first-order error of the parameter estimate. In other words, if the sample moment contains the contamination term $d_i(\theta_0)$, then the first-order influence of sample i on the parameter vector is $-n^{-1} M d_i(\theta_0)$. When deriving the optimal validation probability, the constant $-n^{-1}$ does not affect the ranking across samples, so it suffices to focus on $M d_i(\theta_0)$.

Now let us apply the above theory to the setting studied in this paper. To focus on the target parameter of interest, let

$$M = (G'WG)^{-1}G'W,$$

where M denotes the first-order influence matrix through which moment-condition contamination is transmitted to the target parameter. After mapping the corrected moment $g_i^{AI}(\theta_0) = g_i(\theta_0; X_i) + \left(\frac{\xi_i}{\pi_i} - 1\right) d_i(\theta_0)$ to the target parameter through the GMM first-order influence matrix, the additional noise becomes

$$\left(\frac{\xi_i}{\pi_i} - 1\right) M d_i(\theta_0).$$

Since $E[\xi_i/\pi_i - 1 \mid X_i, \widehat{X}_i] = 0$, this term is mean-zero noise and does not introduce systematic bias, but it does introduce additional variance. Conditional on $X_i, \widehat{X}_i$, $Md_i(\theta_0)$ is fixed, and the only randomness comes from the validation indicator ξ_i. We have $Var\left(\frac{\xi_i}{\pi_i} - 1 \mid X_i, \widehat{X}_i\right) = \frac{1}{\pi_i} - 1$. Therefore,

$$Var\left[\left(\frac{\xi_i}{\pi_i} - 1\right) M d_i(\theta_0) \mid X_i, \widehat{X}_i\right] = (M d_i(\theta_0))^2 \left(\frac{1}{\pi_i} - 1\right).$$

The expected additional sampling variance is therefore $E\left[(M d_i(\theta_0))^2 \left(\frac{1}{\pi_i} - 1\right)\right]$. Since $E[(M d_i(\theta_0))^2]$ is unrelated to π_i, choosing π_i only requires minimizing $E\left[\frac{(M d_i(\theta_0))^2}{\pi_i}\right]$. Thus, the active validation objective can be written as

$$\min_{\pi(\cdot)} E\left[\frac{h(\widehat{X}_i)}{\pi(\widehat{X}_i)}\right], \qquad \text{s.t. } E[\pi(\widehat{X}_i)] \le \rho,$$

where the constraint is the budget constraint and $\rho = n_b/n$ denotes the average validation rate. The Lagrange first-order condition gives, in the interior case ignoring the upper-bound constraint $\pi(\widehat{X}_i) \le 1$,

$$\pi^{opt}(\widehat{X}_i) \propto \sqrt{h(\widehat{X}_i)} = \sqrt{E[(M d_i(\theta_0))^2 \mid \widehat{X}_i]}.$$

Below we provide the formal proof of this oracle rule.

Proposition 2: Oracle active rule. Ignoring the upper-bound constraint $\pi(\widehat{X}_i) \le 1$, under the validation budget $E[\pi(\widehat{X}_i)] \le \rho$, the validation probability that minimizes $E\left[\frac{h(\widehat{X}_i)}{\pi(\widehat{X}_i)}\right]$ satisfies

$$\pi^{opt}(\widehat{X}_i) \propto \sqrt{h(\widehat{X}_i)} = \sqrt{E[(M d_i)^2 \mid \widehat{X}_i]}.$$

If the budget constraint binds, then

$$\pi^{opt}(\widehat{X}_i) = \rho \frac{\sqrt{h(\widehat{X}_i)}}{E[\sqrt{h(\widehat{X}_i)}]}.$$

Proof. Consider the optimization problem $\min_{\pi(\cdot)} E\left[\frac{h(\widehat{X}_i)}{\pi(\widehat{X}_i)}\right]$ subject to $E[\pi(\widehat{X}_i)] = \rho$. Construct the Lagrangian $L = E\left[\frac{h(\widehat{X}_i)}{\pi(\widehat{X}_i)}\right] + \lambda\left(E[\pi(\widehat{X}_i)] - \rho\right)$. Taking the first-order condition with respect to $\pi(\widehat{X}_i)$ at each value of $\widehat{X}_i$ gives $-\frac{h(\widehat{X}_i)}{\pi(\widehat{X}_i)^2} + \lambda = 0$. Therefore, $\pi(\widehat{X}_i)^2 = \frac{h(\widehat{X}_i)}{\lambda}$. Since $\pi(\widehat{X}_i) > 0$, $\pi^{opt}(\widehat{X}_i) = \sqrt{\frac{1}{\lambda}}\sqrt{h(\widehat{X}_i)}$. Let $c = \sqrt{1/\lambda}$. Then $\pi^{opt}(\widehat{X}_i) = c\sqrt{h(\widehat{X}_i)}$. By the budget constraint, $E[\pi^{opt}(\widehat{X}_i)] = cE[\sqrt{h(\widehat{X}_i)}] = \rho$. Therefore, $c = \frac{\rho}{E[\sqrt{h(\widehat{X}_i)}]}$. Thus, $\pi^{opt}(\widehat{X}_i) = \rho\frac{\sqrt{h(\widehat{X}_i)}}{E[\sqrt{h(\widehat{X}_i)}]}$. This completes the proof.

The derivation above ignores the constraint that the validation probability cannot exceed one. If the constraint $0 < \pi(\widehat{X}_i) \leq 1$ is imposed, the optimal rule becomes the truncated form

$$\pi^{opt}(\widehat{X}_i) = \min\left\{1,\, c\sqrt{h(\widehat{X}_i)}\right\} = \min\left\{1,\, c\sqrt{E[(Md_i)^2 \mid \widehat{X}_i]}\right\}.$$

Since the true value of M is unknown in practice, this paper can estimate M using historical validation samples and construct $\widehat{B} = \widehat{M}'\widehat{M}$. The conditional contamination risk can then be written as $\widehat{h}(\widehat{X}_i) = E[d_i'\widehat{B}d_i \mid \widehat{X}_i] = E[(\widehat{M}d_i)^2 \mid \widehat{X}_i]$. The corresponding plug-in active validation rule is

$$\pi_i^{plug-in} \propto \sqrt{E[(\widehat{M}d_i)^2 \mid \widehat{X}_i]}.$$

If $\widehat{M}$ is a consistent estimator of M, this rule can be viewed as a feasible approximation to the oracle active rule.

4.2 Efficiency Comparison with Uniform Validation

The uniform validation rule is $\pi^{unif}(\widehat{X}_i) = \rho$. For a scalar target, the objective value under uniform validation is $E\left[\frac{h(\widehat{X}_i)}{\rho}\right] = \frac{E[h(\widehat{X}_i)]}{\rho}$. The objective value under the oracle active rule is $E\left[\frac{h(\widehat{X}_i)}{\pi^{opt}(\widehat{X}_i)}\right]$. Using $\pi^{opt}(\widehat{X}_i) = \rho\frac{\sqrt{h(\widehat{X}_i)}}{E[\sqrt{h(\widehat{X}_i)}]}$, we obtain $E\left[\frac{h(\widehat{X}_i)}{\pi^{opt}(\widehat{X}_i)}\right] = E\left[\frac{h(\widehat{X}_i)E[\sqrt{h(\widehat{X}_i)}]}{\rho\sqrt{h(\widehat{X}_i)}}\right]$. Rearranging gives $E\left[\frac{h(\widehat{X}_i)}{\pi^{opt}(\widehat{X}_i)}\right] = \frac{E[\sqrt{h(\widehat{X}_i)}]^2}{\rho}$. By Jensen's inequality or the Cauchy–Schwarz inequality, $E[\sqrt{h(\widehat{X}_i)}]^2 \leq E[h(\widehat{X}_i)]$. Therefore, $\frac{E[\sqrt{h(\widehat{X}_i)}]^2}{\rho} \leq \frac{E[h(\widehat{X}_i)]}{\rho}$. This shows that the variance objective value of the oracle active rule is no greater than that of uniform validation, and the two are the same only when $h(\widehat{X}_i)$ is almost surely constant. In other words, if moment-condition contamination is heterogeneous across samples, active validation can strictly improve on uniform validation.

4.3 Practical Active Validation Rule

From the derivation in the previous section, the oracle rule depends on $h(\widehat{X}_i) = E[(Md_i)^2 \mid \widehat{X}_i]$, but $d_i = g_i(\theta; X_i) - g_i(\theta; \widehat{X}_i)$ is unobserved in unvalidated samples. Therefore, in practice, one first uses the true X_i's in the existing validation set to compute d_i, then trains a parameter-contamination prediction model $\widehat{h}(\widehat{X}_i)$, constructs $u_i = \sqrt{\widehat{h}(\widehat{X}_i)}$, and finally sets $\pi_i^{active} = \eta u_i$, where η is chosen to satisfy the budget constraint, thereby ensuring the practical feasibility of the method. The specific steps are as follows.

The first step is to compute $d_i(\widehat{\theta}) = g_i(\widehat{\theta}; X_i) - g_i(\widehat{\theta}; \widehat{X}_i)$ in the pilot sample, where $\widehat{\theta}$ can be an estimator obtained using the true X_i's in the existing validation set. Since $\widehat{\theta}$ is used in place of θ, cross-fitting is needed in practical computation.

The second step is to train a parameter-contamination prediction model $\widehat{h}(\widehat{X}_i) \approx E[(Md_i)^2 \mid \widehat{X}_i]$. Alternatively, one can first estimate $r(\widehat{X}_i) \approx E[(X_i - \widehat{X}_i)^2 \mid \widehat{X}_i]$, and then multiply by the corresponding moment weight according to the specific model structure.

The third step is to construct the active score $u_i = \sqrt{\widehat{h}(\widehat{X}_i)}$, and then convert it into a validation probability: $\pi_i^{active} = \eta u_i$, where $\widehat{\eta} = \max\left\{\eta \in H : \eta\sum_{i=1}^n u(X_i) \leq n_b\right\}$. Since the expected number of validation samples is $E\left[\sum_{i=1}^n \xi_i\right] = \sum_{i=1}^n E(\xi_i) = \sum_{i=1}^n \pi_i$, choosing $\widehat{\eta}$ according to the rule above can be used to satisfy the budget constraint $\sum_{i=1}^n \pi_i^{active} \approx n_b$.

4.4 A More Robust Sampling Rule

To maximize validation efficiency, the active validation probability should satisfy $\pi_i^{act} \propto u(\widehat{X}_i)$. However, $u(\widehat{X}_i)$ itself is estimated and cannot be exactly equal to the true contamination intensity. If the contamination-intensity predictor mistakenly assigns a predicted contamination

Algorithm 1 Feasible Active Validation Rule

Require: Unvalidated data $\{Y_i, \widehat{X}_i\}_{i=1}^n$, validation budget n_b, pilot sample $I_p \subset \{1, \ldots, n\}$, $|I_p| = n_p$, initial estimator $\widehat{\theta}$, candidate scaling-parameter set H

Ensure: Active validation probabilities $\{\pi_i^{active}\}_{i=1}^n$

1: **for** $i \in I_p$ **do**

2: Compute the sample's moment-condition contamination: $d_i(\widehat{\theta}) = g_i(\widehat{\theta}; X_i) - g_i(\widehat{\theta}; \widehat{X}_i)$.

3: **end for**

4: Estimate the conditional parameter-contamination prediction model using the pilot validation sample: $\widehat{h}(\widehat{X}_i) \approx E[(Md_i)^2 \mid \widehat{X}_i]$.

5: Construct the active validation score: $u_i = \sqrt{\widehat{h}(\widehat{X}_i)}$.

6: Choose the scaling parameter: $\widehat{\eta} = \max\{\eta \in H : \eta \sum_{i=1}^n u_i \leq n_b\}$.

7: **for** $i = 1, \ldots, n$ **do**

8: Set the active validation probability: $\pi_i^{active} = \widehat{\eta} u_i$.

9: **end for**

10: Output active validation probabilities $\{\pi_i^{active}\}_{i=1}^n$.

close to zero to some high-contamination samples, that is, $u(\widehat{X}_i) \approx 0$, then the active validation probability π_i^{act} of these samples will be very small. Once such samples are validated, the inverse probability weight $1/\pi_i^{act}$ in the correction term will be extremely amplified, leading to a highly unstable estimator with large variance. This issue becomes even more concerning when $\widehat{h}(\widehat{X}_i)$ is highly inaccurate.

To mitigate this problem, this paper mixes the active validation rule with the uniform validation rule. Let the uniform validation probability be $\pi_i^{unif} = \frac{n_b}{n}$. For a given mixing parameter $\tau \in [0, 1]$, the final validation probability is defined as

$$\pi_i^{(\tau)} = (1 - \tau)\pi_i^{act} + \tau \pi_i^{unif}.$$

When $\tau = 0$, the rule reduces to pure active validation; when $\tau = 1$, the rule reduces to uniform validation. As long as $\tau > 0$, all samples have a positive lower bound on the validation probability: $\pi_i^{(\tau)} \geq \tau \frac{n_b}{n}$. Therefore, the mixed rule can avoid extreme inverse probability weights caused by excessively small $u(\widehat{X}_i)$, and improve the stability of the active corrected estimator. We can use the existing validation set to choose the mixing parameter τ. Suppose the existing validation data are $\left\{Y_i^h, \widehat{X}_i^h, X_i^h\right\}_{i=1}^{n_h}$, where X_i^h denotes the true variable obtained by manual validation in the historical sample. For each candidate τ, first calculate the corresponding mixed validation probability in the historical sample: $\pi_i^{(\tau),h} = (1 - \tau)\pi_i^{act,h} + \tau \pi_i^{unif,h}$. Use this probability as the active validation probability, and then adjust τ to minimize the corresponding variance proxy on the historical data:

$$\widehat{\tau} = \arg \min_{\tau \in [0,1]} \sum_{i=1}^{n_h} \frac{\left(\widehat{M} d_i^h(\widehat{\theta})\right)^2}{\pi_i^{(\tau),h}}.$$

This objective function corresponds to the part of the asymptotic variance of the active corrected estimator that is related to the sampling rule. If the contamination-intensity predictor $u(\widehat{X}_i)$ performs well on the historical data, a smaller τ will be selected, so that the procedure relies more on active validation. If $u(\widehat{X}_i)$ is unstable on the historical data, a larger τ will be selected, increasing the share of uniform validation probability.

5 Applications of Active Validation in Econometric Regression Models

This section presents the forms of moment contamination in commonly used econometric models and analyzes the optimal validation rules in these models.

5.1 OLS

In Section 3.1, we used a univariate linear regression model to illustrate the influence of machine-learning-generated variables on OLS estimation. We now analyze the form of moment contamination in OLS more generally in a multivariate regression problem, and derive the oracle active rule under OLS.

First consider an ordinary linear regression. The true model is

$$Y_i = \beta_0 X_i + W_i'\gamma_0 + \varepsilon_i,$$

where X_i is the true explanatory variable, and W_i includes control variables and an intercept. To simplify notation, first residualize Y_i, X_i, and $\widehat{X}_i$ with respect to W_i: $\widetilde{Y}_i = Y_i - \Pi_W Y_i$, $\widetilde{X}_i = X_i - \Pi_W X_i$, $\widetilde{\widehat{X}}_i = \widehat{X}_i - \Pi_W \widehat{X}_i$. Here, Π_W denotes the linear projection on the control variables W_i. After residualization, the true OLS moment can be written as $g_i^{OLS}(\beta; X_i) = \widetilde{X}_i(\widetilde{Y}_i - \beta\widetilde{X}_i)$. If the machine-learning-generated variable $\widehat{X}_i$ is directly used, the naive OLS moment is $g_i^{OLS}(\beta; \widehat{X}_i) = \widetilde{\widehat{X}}_i(\widetilde{Y}_i - \beta\widetilde{\widehat{X}}_i)$. Therefore, the OLS moment contamination is

$$d_i^{OLS}(\beta) = g_i^{OLS}(\beta; X_i) - g_i^{OLS}(\beta; \widehat{X}_i),$$

that is,

$$d_i^{OLS}(\beta) = \widetilde{X}_i(\widetilde{Y}_i - \beta\widetilde{X}_i) - \widetilde{\widehat{X}}_i(\widetilde{Y}_i - \beta\widetilde{\widehat{X}}_i).$$

Let $\widetilde{e}_i = \widetilde{\widehat{X}}_i - \widetilde{X}_i$. Then $\widetilde{\widehat{X}}_i = \widetilde{X}_i + \widetilde{e}_i$. At the true parameter β_0, if $\widetilde{Y}_i = \beta_0\widetilde{X}_i + \widetilde{\varepsilon}_i$, then the naive OLS moment is $g_i^{OLS}(\beta_0; \widehat{X}_i) = (\widetilde{X}_i + \widetilde{e}_i)(\widetilde{\varepsilon}_i - \beta_0\widetilde{e}_i)$. The true OLS moment is $g_i^{OLS}(\beta_0; X_i) = \widetilde{X}_i\widetilde{\varepsilon}_i$. Therefore, $g_i^{OLS}(\beta_0; \widehat{X}_i) - g_i^{OLS}(\beta_0; X_i) = \widetilde{e}_i\widetilde{\varepsilon}_i - \beta_0\widetilde{X}_i\widetilde{e}_i - \beta_0\widetilde{e}_i^2$. Hence,

$$d_i^{OLS}(\beta_0) = -\widetilde{e}_i\widetilde{\varepsilon}_i + \beta_0\widetilde{X}_i\widetilde{e}_i + \beta_0\widetilde{e}_i^2.$$

This expression further shows that, in OLS, the contamination of the target moment condition by machine learning prediction error is not simply $\widetilde{e}_i = \widetilde{\widehat{X}}_i - \widetilde{X}_i$, but includes terms such as $\widetilde{e}_i\widetilde{\varepsilon}_i$, $\widetilde{X}_i\widetilde{e}_i$, and $\widetilde{e}_i^2$. Moreover, since the contamination of the parameter can be written as the contamination of the moment condition multiplied by the first-order influence matrix, under OLS the most valuable samples to validate are not simply those with the largest prediction errors, but those that cause the largest contamination to $\widetilde{X}_i(\widetilde{Y}_i - \beta\widetilde{X}_i)$.

The corresponding oracle active rule can be written as

$$\pi_i^{OLS,opt} \propto \sqrt{E[(d_i^{OLS}(\beta_0))^2 \mid \widehat{X}_i]}.$$

5.2 IV

First consider the case with a single instrumental variable. Let the instrumental variable be Z_i, and denote its residualized version by $\widetilde{Z}_i$. The true IV moment is $g_i^{IV}(\beta; X_i) = \widetilde{Z}_i(\widetilde{Y}_i - \beta\widetilde{X}_i)$. When using the machine-learning-generated variable, the naive IV moment is $g_i^{IV}(\beta; \widehat{X}_i) = \widetilde{Z}_i(\widetilde{Y}_i - \beta\widetilde{\widehat{X}}_i)$. Therefore, the moment contamination in IV is $d_i^{IV}(\beta) = g_i^{IV}(\beta; X_i) - g_i^{IV}(\beta; \widehat{X}_i)$. Expanding gives $d_i^{IV}(\beta) = \widetilde{Z}_i(\widetilde{Y}_i - \beta\widetilde{X}_i) - \widetilde{Z}_i(\widetilde{Y}_i - \beta\widetilde{\widehat{X}}_i)$. Rearranging yields $d_i^{IV}(\beta) = \beta\widetilde{Z}_i(\widetilde{\widehat{X}}_i -$

$\widetilde{X}_i$). Let $\widetilde{e}_i = \widetilde{\widehat{X}}_i - \widetilde{X}_i$. Then $d_i^{IV}(\beta) = \beta\widetilde{Z}_i\widetilde{e}_i$. At the true parameter β_0, $d_i^{IV}(\beta_0) = \beta_0\widetilde{Z}_i\widetilde{e}_i$. Therefore, under scalar IV, the oracle active rule is

$$\pi_i^{IV,opt} \propto \sqrt{E[\beta_0^2\widetilde{Z}_i^2\widetilde{e}_i^2 \mid \widehat{X}_i]}.$$

Since β_0 is a constant and $\widetilde{Z}_i$ is observable before validation, the optimal validation rule in the IV setting is

$$\pi_i^{IV,opt} \propto |\widetilde{Z}_i|\sqrt{E[\widetilde{e}_i^2 \mid \widehat{X}_i]}.$$

If there are multiple instrumental variables, $\widetilde{Z}_i \in \mathbb{R}^q$, then $g_i^{IV}(\beta; X_i) = \widetilde{Z}_i(\widetilde{Y}_i - \beta\widetilde{X}_i)$, where $g_i^{IV}(\beta; X_i) \in \mathbb{R}^q$. The contamination term is $d_i^{IV}(\beta) = \beta\widetilde{Z}_i\widetilde{e}_i$. If the goal is to reduce GMM-weighted moment contamination, we can take the weighting matrix W and obtain $\pi_i^{IV-GMM,opt} \propto \sqrt{E[d_i^{IV}(\beta_0)'Wd_i^{IV}(\beta_0) \mid \widehat{X}_i]}$. Substituting $d_i^{IV}(\beta_0) = \beta_0\widetilde{Z}_i\widetilde{e}_i$ gives $\pi_i^{IV-GMM,opt} \propto \sqrt{E[\beta_0^2\widetilde{e}_i^2\widetilde{Z}_i'W\widetilde{Z}_i \mid \widehat{X}_i]}$. Thus, $\pi_i^{IV-GMM,opt} \propto \|W^{1/2}\widetilde{Z}_i\|\sqrt{E[\widetilde{e}_i^2 \mid \widehat{X}_i]}$. It should be noted that our ultimate goal is to reduce the effect of contamination on the coefficient β. Therefore, the oracle rule should be adjusted accordingly as

$$\pi_i^{IV-\beta,opt} \propto |a_\beta'\widetilde{Z}_i|\sqrt{E[\widetilde{e}_i^2 \mid \widehat{X}_i]}.$$

5.3 DiD

Next consider difference-in-differences. The core idea of DiD is to compare changes before and after treatment between the treatment group and the control group. If the treatment variable is generated from unstructured data by machine learning, then machine learning errors contaminate the DiD identifying moment condition.

For simplicity, first consider two-period two-group DiD. Let $\Delta Y_i = Y_{i1} - Y_{i0}$ denote the change in outcome for individual i, and let the true treatment status be $X_i \in \{0, 1\}$. The classical DiD can be written as $\Delta Y_i = \alpha_0 + \tau_0 X_i + \eta_i$, where α_0 is the average change in outcomes for the control group, and τ_0 is the ATT. Note that if the regression includes an intercept, we cannot directly use $X_i(\Delta Y_i - \tau X_i)$ as the corresponding moment. Instead, the constant term should first be residualized out. Let $\overline{\Delta Y} = n^{-1}\sum_{i=1}^n \Delta Y_i$, $\bar{D} = n^{-1}\sum_{i=1}^n X_i$, and define the demeaned variables $\widetilde{\Delta Y}_i = \Delta Y_i - \overline{\Delta Y}$, $\widetilde{X}_i = X_i - \bar{D}$. Then, the moment for τ in the DiD regression with an intercept can be written as

$$g_i^{DiD}(\tau; X_i) = \widetilde{X}_i\left(\widetilde{\Delta Y}_i - \tau\widetilde{X}_i\right).$$

If the machine-learning-generated treatment status is $\widehat{X}_i = X_i + e_i$, it also needs to be demeaned. Let $\overline{\widehat{D}} = n^{-1}\sum_{i=1}^n \widehat{X}_i$, and define $\widetilde{\widehat{X}}_i = \widehat{X}_i - \overline{\widehat{D}}$. The corresponding naive DiD moment is

$$g_i^{DiD}(\tau; \widehat{X}_i) = \widetilde{\widehat{X}}_i\left(\widetilde{\Delta Y}_i - \tau\widetilde{\widehat{X}}_i\right).$$

Therefore, the moment contamination in DiD is $d_i^{DiD}(\tau) = g_i^{DiD}(\tau; X_i) - g_i^{DiD}(\tau; \widehat{X}_i)$, that is,

$$d_i^{DiD}(\tau) = \widetilde{X}_i\left(\widetilde{\Delta Y}_i - \tau\widetilde{X}_i\right) - \widetilde{\widehat{X}}_i\left(\widetilde{\Delta Y}_i - \tau\widetilde{\widehat{X}}_i\right).$$

To see the structure of the contamination term clearly, let $\widetilde{e}_i = \widetilde{\widehat{X}}_i - \widetilde{X}_i$. At the true parameter τ_0, if the demeaned model satisfies $\widetilde{\Delta Y}_i = \tau_0\widetilde{X}_i + \widetilde{\eta}_i$, then

$$g_i^{DiD}(\tau_0; \widehat{X}_i) = (\widetilde{X}_i + \widetilde{e}_i)(\widetilde{\eta}_i - \tau_0\widetilde{e}_i).$$

The true moment is $g_i^{DiD}(\tau_0; X_i) = \widetilde{X}_i\widetilde{\eta}_i$. Therefore,

$$g_i^{DiD}(\tau_0; \widehat{X}_i) - g_i^{DiD}(\tau_0; X_i) = \widetilde{e}_i\widetilde{\eta}_i - \tau_0\widetilde{X}_i\widetilde{e}_i - \tau_0\widetilde{e}_i^2.$$

Hence,

$$d_i^{DiD}(\tau_0) = -\widetilde{e}_i\widetilde{\eta}_i + \tau_0\widetilde{X}_i\widetilde{e}_i + \tau_0\widetilde{e}_i^2.$$

The corresponding oracle active rule is

$$\pi_i^* \propto \sqrt{E[(d_i^{DiD}(\tau_0))^2 \mid \widehat{X}_i]},$$

with normalization under the budget constraint $\sum_{i=1}^n \pi_i^* = n_b$ and the probability constraint $0 < \pi_i^* \leq 1$.

In two-period two-group DiD with an intercept, since

$$d_i^{DiD}(\tau_0) = -\widetilde{e}_i\widetilde{\eta}_i + \tau_0\widetilde{X}_i\widetilde{e}_i + \tau_0\widetilde{e}_i^2,$$

the oracle rule is equivalent to allocating more validation budget to observations with larger

$$E\left[\left(-\widetilde{e}_i\widetilde{\eta}_i + \tau_0\widetilde{X}_i\widetilde{e}_i + \tau_0\widetilde{e}_i^2\right)^2 \mid \widehat{X}_i\right].$$

Similarly, the true treatment effect in DiD can be written as the difference in expected changes in outcomes between the treatment and control groups: $\tau_0 = E[\Delta Y_i \mid D_i = 1] - E[\Delta Y_i \mid D_i = 0]$. In the linear regression representation with an intercept, this parameter can also be written as $\tau_0 = \frac{E[\widetilde{D}_i \Delta Y_i]}{E[\widetilde{D}_i^2]}$, where $\widetilde{D}_i = D_i - E[D_i]$. Therefore, whether the machine-learning-generated treatment status seriously affects the DiD estimate does not depend only on the overall error rate of the treatment classifier, but depends on whether these errors substantially change the treatment-group versus control-group difference in outcome changes. In other words, even if the prediction error rate for treatment status is high, if the errors mainly occur among observations that contribute little to $E[\Delta Y_i \mid D_i = 1] - E[\Delta Y_i \mid D_i = 0]$, or if these misclassifications do not substantially change the comparison between treatment and control groups, then their effect on the DiD estimator may remain limited. Conversely, even if the overall error rate is not high, if misclassifications are concentrated among observations that exert a large influence on the treatment-control difference in changes, they may cause large moment-condition contamination. Therefore, under the DiD setting, the active validation rule should not simply rank observations by the treatment classifier's uncertainty or error probability. Instead, it should prioritize validating samples whose misclassification is most likely to change the DiD identifying contrast and thereby exert a large influence on the estimate of τ_0.

5.4 RD

We first consider the sharp RD case. Let the true running variable be R_i, and let the cutoff be c. To simplify notation, let $x_i = R_i - c$. If the running variable is machine-learning-generated, then the researcher actually observes $\widehat{R}_i$, and $\widehat{x}_i = \widehat{R}_i - c$. In sharp RD, treatment status is determined by $D_i^{RD} = 1\{R_i \geq c\} = 1\{x_i \geq 0\}$. The object to be estimated is the conditional expectation limits on both sides of the cutoff:

$$\mu_+(c) = \lim_{r \downarrow c} E[Y_i \mid R_i = r], \qquad \mu_-(c) = \lim_{r \uparrow c} E[Y_i \mid R_i = r].$$

Let $K_h(x_i) = \frac{1}{h}K\left(\frac{x_i}{h}\right)$ be the kernel weight, and let $p(x_i) = (1, x_i)' = (1, R_i - c)'$ be the local linear basis. We need to estimate $\widehat{\alpha}_+ = \arg\min_{\alpha_+} \sum_{i:R_i \geq c} K_h(R_i - c)\left[Y_i - p(R_i - c)'\alpha_+\right]^2$. The resulting estimator $\widehat{\alpha}_{+,0}$ is the estimate of the right-side conditional expectation at the cutoff,

that is, $\widehat{\mu}_+(c) = \widehat{\alpha}_{+,0}$. Similarly, the estimate of the left-side conditional expectation at the cutoff is $\widehat{\mu}_-(c) = \widehat{\alpha}_{-,0}$. Thus, the RD treatment effect estimator is

$$\tau_{RD} = \widehat{\mu}_+(c) - \widehat{\mu}_-(c) = \widehat{\alpha}_{+,0} - \widehat{\alpha}_{-,0}.$$

To write local linear RD regression in the form of moment conditions, take the first-order condition with respect to α_+ in $\widehat{\alpha}_+ = \arg\min_{\alpha_+} \sum_{i=1}^n 1\{R_i \geq c\} K_h(R_i - c) \left[Y_i - p(R_i - c)'\alpha_+\right]^2$. The solution $\widehat{\alpha}_+$ satisfies $\sum_{i=1}^n 1\{R_i \geq c\} K_h(R_i - c) p(R_i - c) \left[Y_i - p(R_i - c)'\widehat{\alpha}_+\right] = 0$. Therefore, we can define the contribution of each sample as

$$g_{i,+}^{RD}(\alpha_+; R_i) = 1\{R_i \geq c\} K_h(R_i - c) p(R_i - c) \left[Y_i - p(R_i - c)'\alpha_+\right],$$

and write the sample moment as $\frac{1}{n}\sum_{i=1}^n g_{i,+}^{RD}(\widehat{\alpha}_+; R_i) = 0$. If the researcher uses the machine-learning-generated running variable $\widehat{R}_i$, the right-side naive moment is $g_{i,+}^{RD}(\alpha_+; \widehat{R}_i) = 1\{\widehat{R}_i \geq c\} K_h(\widehat{R}_i - c) p(\widehat{R}_i - c) \left[Y_i - p(\widehat{R}_i - c)'\alpha_+\right]$. Then define the right-side moment contamination term as $d_{i,+}^{RD}(\alpha_+) = g_{i,+}^{RD}(\alpha_+; R_i) - g_{i,+}^{RD}(\alpha_+; \widehat{R}_i)$. That is,

$$d_{i,+}^{RD}(\alpha_+) = 1\{R_i \geq c\} K_h(R_i - c) p(R_i - c) \left[Y_i - p(R_i - c)'\alpha_+\right]$$

$$-1\{\widehat{R}_i \geq c\} K_h(\widehat{R}_i - c) p(\widehat{R}_i - c) \left[Y_i - p(\widehat{R}_i - c)'\alpha_+\right].$$

The left-side moment contamination term is defined analogously. Combining the left and right sides, the RD moment contamination can be written as

$$d_i^{RD}(\theta) = \begin{pmatrix} d_{i,+}^{RD}(\alpha_+) \\ d_{i,-}^{RD}(\alpha_-) \end{pmatrix},$$

where $\theta = (\alpha_+', \alpha_-')'$.

The special feature of RD is that samples far from the cutoff may not matter even if the prediction error of the running variable is large; whereas samples near the cutoff may seriously contaminate the RD moment if they cross the cutoff, even when the prediction error is not large. That is, when $R_i < c$ but $\widehat{R}_i \geq c$, or when $R_i \geq c$ but $\widehat{R}_i < c$, the sample is incorrectly assigned to the other side of the cutoff, directly contaminating RD identification. For this reason, active validation in RD should not simply validate samples with the largest $|\widehat{R}_i - R_i|$, but should validate those samples that have the largest influence on the local RD moment. The corresponding oracle rule is

$$\pi_i^{RD,opt} \propto \sqrt{E[(a_\tau' d_i^{RD}(\theta_0))^2 \mid \widehat{x}_i]},$$

where $\tau = \alpha_{+,0} - \alpha_{-,0}$, and a_τ is the corresponding parameter influence direction. In practice, the RD active score can be approximated as

$$u_i^{RD} = K_h(\widehat{x}_i) \cdot r_R(\widehat{x}_i) \cdot s_i^{switch},$$

where $r_R(\widehat{x}_i) \approx \sqrt{E[(\widehat{R}_i - R_i)^2 \mid \widehat{x}_i]}$ represents the prediction-error risk of the running variable, $K_h(\widehat{x}_i)$ represents the local weight of the sample near the cutoff, and s_i^{switch} represents the risk that the sample crosses the cutoff. A simple choice is to use the historical validation set to compute $s_i^{switch} = P(x_i \widehat{x}_i < 0 \mid \widehat{x}_i)$, that is, $s_i^{switch} = P((R_i - c)(\widehat{R}_i - c) < 0 \mid \widehat{x}_i)$. This is the probability that the true running variable and the predicted running variable lie on opposite sides of the cutoff. This score captures the core feature of RD: the most valuable samples to validate are those near the cutoff, with high side-switching risk, and with large local estimation weights.

If the regression method is fuzzy RD, the treatment status T_i is not fully determined by the cutoff; instead, the treatment probability jumps near the cutoff. In this case, it can be written as a local IV moment condition:

$$g_i^{FRD}(\beta; T_i, R_i) = K_h(R_i - c)Z_i^{RD}(Y_i - \beta T_i),$$

where $Z_i^{RD} = 1\{R_i \geq c\}$. If T_i is a machine-learning-generated variable, and the researcher actually uses $\widehat{T}_i$, then the naive fuzzy RD moment is $g_i^{FRD}(\beta; \widehat{T}_i, R_i) = K_h(R_i - c)Z_i^{RD}(Y_i - \beta\widehat{T}_i)$. Therefore, the contamination term is $d_i^{FRD}(\beta) = g_i^{FRD}(\beta; T_i, R_i) - g_i^{FRD}(\beta; \widehat{T}_i, R_i)$, that is,

$$d_i^{FRD}(\beta) = K_h(R_i - c)Z_i^{RD}(Y_i - \beta T_i) - K_h(R_i - c)Z_i^{RD}(Y_i - \beta\widehat{T}_i).$$

Thus,

$$d_i^{FRD}(\beta) = \beta K_h(R_i - c)Z_i^{RD}(\widehat{T}_i - T_i).$$

The validation rule for fuzzy RD is therefore

$$\pi_i^{FRD,opt} \propto K_h(R_i - c)|Z_i^{RD}|\sqrt{E[(\widehat{T}_i - T_i)^2 \mid \widehat{T}_i]}.$$

6 Performance on a Real-World Dataset

6.1 Data Description and Task Description

The data used in this section are basically the same as those used in the real-data analysis of Allon et al. (2023). Therefore, the data description and task description can be found in Section 6.1 of that paper. For ease of reading, the relevant data description and task description from Allon et al. (2023) are excerpted below.

The Civil Comments platform was an online comment plug-in for independent news websites, operating from 2015 to 2017.[1] The dataset contains the text of 1.8 million comments, as well as metadata such as article ID and timestamp, along with crowdsourced ratings of each comment's content and quality. The key variables are *Approval*, which indicates whether a comment was initially approved by other commenters, and *Toxicity*, which indicates whether a comment was rated as toxic by at least half of the human annotators. For example, a comment saying "Wow, that sounds great." was initially approved and later rated as toxic by 0 of 4 annotators ($Approval = 1, Toxicity = 0$). Another comment saying "haha you guys are a bunch of losers" was initially rejected and rated as toxic by 42 of 47 annotators ($Approval = 0, Toxicity = 1$). Additional variables capture the demographic identities most commonly mentioned in comments. For example, a comment saying "The article says Sarah would be the first female candidate in THIS race|the 2016 one." was labeled by 8 of 10 annotators as mentioning women ($female = 1$). Identity mentions are relatively sparse and are available for a subset of about 400,000 comments.

The Civil Comments dataset contains common features of machine learning data: it contains 1.8 million comments; and it is skewed, with 8% of comments classified as toxic ($Toxicity = 1$) and 92% classified as non-toxic ($Toxicity = 0$). Finally, the comment text provides standard high-dimensional data that machine learning can use.

Unlike typical text comment datasets, the distinctive feature of this dataset is that we observe the "true" labels for all comments. This allows us to obtain true estimates without prediction error, and compare them with situations in which labels are predicted by machine learning models with different degrees of error. This enables us to evaluate the extent of the problem caused by prediction error and to examine the performance of correction methods.

[1] The dataset and its documentation are available at `https://www.tensorflow.org/datasets/catalog/civil_comments`.

Using the Civil Comments dataset, we now follow a two-stage model to evaluate the influence of machine learning prediction error on causal inference. In the machine learning stage, the toxicity estimate is generated from comment text:

$$\widehat{Toxicity}_{it} = \widehat{f}(Text_{it}) = Toxicity_{it} + \eta_{it}.$$

In this model training, the prediction error η_{it} can be precisely calculated because we observe the crowdsourced ratings for each comment and assume that the majority-rule rating is correct. Therefore, the machine learning algorithm generates prediction errors relative to the crowdsourced rating of *Toxicity*, which serves as the true label. In practice, the true prediction error rate is unknown, but it can be estimated by cross-validation on the training data. From the 1.8 million comments, 10% are randomly sampled as our training data, and the remaining 90% are the test set, with the true *Toxicity* values removed. To simulate a standard research procedure, we use a cross-validated convolutional neural network with word embeddings as the baseline model. This model achieves 94% predictive accuracy; details of the model are provided in Appendix B. This improves on the 92% baseline accuracy obtained by always guessing $Toxicity = 0$.

In the causal inference stage, our regression specification is

$$SevereToxicity_{it} = \beta \widehat{Toxicity}_{it} + z_i + \tau_t + \widetilde{\epsilon}_{it}.$$

The parameter of interest is $\widehat{\beta}$, namely the effect of toxicity on whether a comment is labeled as severely toxic. Similar to a standard empirical model, this specification includes fixed effects for the article in which the comment appears, z_i, and fixed effects for the date on which the comment is posted, τ_t. Since we observe the original toxicity rating, we can also estimate the true parameter β, which is not affected by bias introduced by the first-stage machine learning algorithm, by using the above specification but replacing $\widehat{Toxicity}_{it}$ with $Toxicity_{it}$.

Additional note: In Allon et al. (2023), the dependent variable used is $Toxicity_{it}$, but I have some questions about their experimental results. Therefore, I replace the dependent variable with $SevereToxicity_{it}$. This model is easier to interpret: if a comment is labeled as "toxic," then it is more likely to be labeled as "severely toxic." If the dependent variable is $Approval_{it}$, the effect declines slightly, but the conclusion remains valid.

6.2 Experimental Setup

The original dataset contains approximately 1.8 million observations. Since the main purpose of this section is to validate the effectiveness of the method proposed in this paper rather than to conduct large-scale predictive modeling on the complete dataset, this paper prioritizes computational efficiency and experimental reproducibility, and randomly samples 10% of the original dataset as the experimental sample. This experimental sample contains approximately 180,000 observations and is further divided into a training set and a test set at a ratio of $8:2$. The training set, namely the observations with `dataset_role=train`, contains 144,389 observations and is used to train the first-stage machine learning model. The trained model is then applied to the test set to generate the corresponding machine-learning-predicted variable in the test set. The test set contains 36,098 observations. Ultimately, the trained first-stage model achieves a prediction accuracy of 93.37% on the test set.

In the second-stage estimation, this paper uses only the test sample for regression analysis. Therefore, the sample size available for second-stage estimation is $n = 36{,}098$. In this section, we fix the total validation budget at $B = 3000$ and split it into a pilot validation sample, or existing validation set, and an additional sample that needs to be validated, in a ratio of $1:4$: $B_{pilot} = 600, \quad B_{remain} = 2400$. In each Monte Carlo repetition, the parameter-contamination prediction model is trained only using the 600 validated samples in the pilot stage. Specifically, the contamination-direction score is first computed within the pilot sample and used as the

training label of the parameter-contamination prediction model. Then, the observable features $\widehat{X}_i$ in the pilot sample are used to train a random forest parameter-contamination prediction model. Finally, this model is applied to the observable features $\widehat{X}_i$ of the remaining unvalidated samples to generate the second-stage active sampling probabilities.

This paper compares the naive estimator, the classical inference estimator, the uniform corrected estimator, the active validation estimator, and the oracle estimator. The oracle estimator refers to the estimator obtained using the true variable X_i for all observations in the test set. The naive estimator uses only the machine-learning-generated proxy variable $\widehat{X}_i$ for estimation. The validation-only method uses only 3000 validated samples and their true X_i's for OLS estimation. The uniform corrected method and the active corrected method both use proxy variables $\widehat{X}_i$ to construct corrected moment conditions on all 36,098 test samples, but only 3000 samples additionally observe the true X_i. The difference between the two methods lies in the choice of validation-sample probabilities. Therefore, the validation-only, uniform corrected, and active corrected methods have the same cost of obtaining true labels. The difference is that the latter two correction methods further use proxy-variable information in the full sample, thereby improving second-stage estimation precision under the same validation budget. In summary, we have three benchmarks. The first is the naive estimate, which uses only the proxy variable for regression. The second is classical inference, which conducts uniform validation and uses only the validation sample for regression, namely validation-only. The third is the uniform corrected estimator, which uses the active validation method but sets the validation probability to uniform validation, namely uniform corrected.

The experimental parameters are set as follows. The number of Monte Carlo repetitions is $R = 100$. This section treats the 36,098 samples with `dataset_role=validation` in the external prediction file as the regression-analysis population. All second-stage estimations absorb fixed effects for the article in which the comment appears and fixed effects for the date on which the comment is posted. The mixing parameter τ is selected from the candidate set $T_\tau = \{0, 0.05, 0.10, \ldots, 0.50\}$, with a grid interval of 0.05. Machine learning model training parameters are given in the appendix of Allon et al. (2023). The training parameters for the parameter-contamination prediction model are given in the appendix at the end of this paper.

The oracle estimator uses the true X_i on all 36,098 validation subsamples and gives $\beta^{oracle} = 0.044140$. The corresponding Monte Carlo summary results are shown in Table 1.

Table 1: Monte Carlo results using only the 36,098 samples with `dataset_role=validation`.

Method	Bias	SD	MSE	MAE
Naive	-0.013007	0.000000	0.000169	0.013007
Validation-only	-0.000400	0.004320	0.000019	0.003401
Uniform corrected	0.000089	0.003642	0.000013	0.002877
Active corrected	-0.000062	0.003191	0.000010	0.002575

Table 1 presents the following conclusions. First, the naive estimator that directly uses the machine learning proxy variable exhibits substantial bias. Its average bias is -0.013007, and its mean squared error reaches 0.000169, which is significantly worse than the other three methods. This indicates that even when the machine learning prediction accuracy is high, first-stage prediction error still causes substantial contamination to the moment condition. Second, under the fixed total validation budget $B = 3000$, both correction methods outperform the method that simply uses only the validation sample for estimation. The standard deviation of validation-only is 0.004320, whereas that of uniform corrected has already declined to 0.003642, a reduction of approximately 15.69%. This shows that even if validation samples are allocated uniformly, finite-sample efficiency improves as long as full-sample proxy information is reincorporated into estimation. Third, the active corrected method performs best in this setting. Compared with uniform corrected, the MSE of active corrected further declines from 0.000013 to 0.000010, a

reduction of approximately 23.26%; the mean absolute error declines from 0.002877 to 0.002575, a reduction of approximately 10.50%; and the standard deviation also declines from 0.003642 to 0.003191, a reduction of approximately 12.39%. Therefore, under the same true-label budget, actively designing the validation sample can indeed control the variance of the correction term more effectively.

6.3 Validation-Sample Savings Required to Achieve the Same Precision

The previous section fixed the validation budget at $B = 3000$ and compared the performance of different estimators under the same true-label cost. This section further evaluates the active validation method from the perspective of sample savings: if a traditional method achieves a given average confidence interval width under validation budget B, how many validation samples does the active validation method need to achieve the same width? This question more directly corresponds to the practical labeling-cost implication of the proposed method.

Specifically, for each validation budget B, we first compute the average 90% confidence interval width of the validation-only estimator, the uniform corrected estimator, and the active corrected estimator. Let the average interval width of a benchmark method under budget B be $W_{base}(B)$, and let the average interval width of the active corrected estimator under budget B' be $W_{act}(B')$. If there exist two adjacent budget points B_k and B_{k+1} such that

$$W_{act}(B_k) > W_{base}(B) > W_{act}(B_{k+1}),$$

then linear interpolation is used to estimate the budget required for the active method to achieve the same precision:

$$B_{act}(B) = B_k + \frac{W_{act}(B_k) - W_{base}(B)}{W_{act}(B_k) - W_{act}(B_{k+1})}(B_{k+1} - B_k).$$

The corresponding sample-saving ratio is defined as

$$Saving(B) = \frac{B - B_{act}(B)}{B} \times 100\%.$$

Therefore, if $Saving(B) > 0$, it means that the active validation method can achieve the average interval width of the benchmark method under budget B using fewer true labels. If the value is negative, it means that the active method has not yet shown a sample-saving advantage in that budget range.

This section still uses the 36,098 samples with `dataset_role=validation` as the regression-analysis population. The number of Monte Carlo repetitions is $R = 100$, and the budget grid is set to

$$B \in \{500, 750, 1000, 1500, 2000, 2500, 3000, 4000, 5000, 5500, 6000, 6500, 7000, 10000\}.$$

For each budget point, the active method uses 20% of the total budget as the pilot sample, and the remaining budget for second-stage active validation.

Table 2 shows that the active corrected method is still somewhat unstable relative to the uniform corrected method under extremely small budgets. When $B = 500$, because the pilot sample is too small, the parameter-contamination prediction model has difficulty stably learning the sample characteristics associated with large moment contamination. The average interval width of the active corrected estimator is 0.031638, slightly larger than the 0.031022 of the uniform corrected estimator, corresponding to a negative reduction relative to the uniform corrected estimator. However, starting from $B = 750$, the average interval width of the active method is already smaller than that of both benchmark methods. Its interval width is 0.025417, which is 11.23% and 1.94% shorter than those of the validation-only estimator and the uniform

Table 2: Average 90% confidence interval widths under different validation budgets.

Validation budget B	Average 90% CI width			Relative reduction of active validation	
	Validation-only estimator	Uniform validation estimator	Active validation estimator	vs. validation-only (%)	vs. uniform validation (%)
500	0.034147	0.031022	0.031638	7.35	-1.99
750	0.028631	0.025921	0.025417	11.23	1.94
1000	0.025014	0.023116	0.022456	10.22	2.86
1500	0.021118	0.018739	0.017028	19.37	9.13
2000	0.018553	0.016207	0.014667	20.95	9.50
2500	0.016772	0.014595	0.013011	22.43	10.85
3000	0.015236	0.013341	0.011762	22.80	11.84
4000	0.013142	0.011620	0.010009	23.83	13.86
5000	0.011742	0.010375	0.008890	24.29	14.32
5500	0.011203	0.009938	0.008530	23.86	14.17
6000	0.010692	0.009483	0.008141	23.86	14.15
6500	0.010293	0.009146	0.007820	24.02	14.49
7000	0.009920	0.008794	0.007570	23.69	13.92
10000	0.008304	0.007441	0.006351	23.52	14.65

corrected estimator, respectively. As the budget continues to increase, the advantage of the active method gradually expands and stabilizes. For example, when $B = 3000$, the average interval width of the active corrected estimator is 0.011762, clearly smaller than 0.015236 for the validation-only estimator and 0.013341 for the uniform corrected estimator, corresponding to relative reductions of 22.80% and 11.84%, respectively. Under larger budgets, this advantage remains basically stable. For example, when $B = 10000$, the average interval width of the active corrected estimator further declines to 0.006351, which is 23.52% and 14.65% shorter than those of the validation-only estimator and the uniform corrected estimator, respectively.

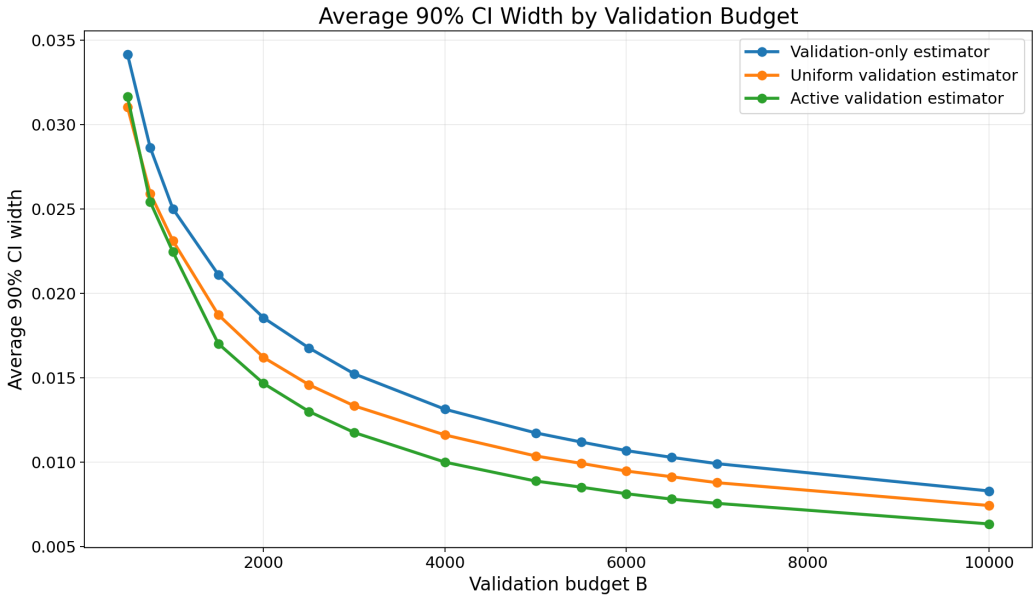

Figure 1: Average 90% confidence interval width as the validation budget changes.

Table 3 reports the sample-saving ratios obtained using the above linear interpolation method. "Relative to validation-only" indicates the proportion of true labels saved by the active method to achieve the same average interval width as the validation-only estimator under budget B. "Relative to uniform corrected" indicates the saving ratio of the active method relative to the uniform corrected estimator.

Similarly, Table 3 shows that the active validation method does not perform best in all

budget ranges. When $B = 500$, the active method has a negative sample-saving ratio relative to the uniform corrected estimator. This is consistent with the theoretical discussion above: correction is not unconditionally beneficial, and active validation also needs enough information to estimate a reasonable validation probability.

However, once the budget increases slightly, the active validation method begins to deliver substantial sample savings. When the benchmark budget is $B = 3000$, the validation-only estimator needs 3000 true labels to achieve an average interval width of 0.015236, whereas the active corrected estimator is estimated by linear interpolation to need only about 1879 true labels to achieve the same width, corresponding to a saving of 37.35%. Relative to the uniform corrected estimator, the active method requires about 2400 labels to achieve the same width, corresponding to a saving of 19.99%. At the maximum budget point $B = 10000$, the sample saving of the active validation method reaches 42.09% relative to the validation-only estimator and 26.82% relative to the uniform corrected estimator.

Table 3: Validation-sample saving ratios required to achieve the same average interval width.

	Relative to validation-only		Relative to uniform corrected	
Budget B	Active required budget	Saving ratio (%)	Active required budget	Saving ratio (%)
500	500.00	0.00	524.77	-4.95
750	620.84	17.22	729.76	2.70
1000	784.08	21.59	944.26	5.57
1500	1123.25	25.12	1342.41	10.51
2000	1359.56	32.02	1673.87	16.31
2500	1554.09	37.84	2021.76	19.13
3000	1879.36	37.35	2400.16	19.99
4000	2460.48	38.49	3081.20	22.97
5000	3011.17	39.78	3791.19	24.18
5500	3318.79	39.66	4063.66	26.12
6000	3610.53	39.82	4470.32	25.49
6500	3838.03	40.95	4771.23	26.60
7000	4079.48	41.72	5133.27	26.67
10000	5790.55	42.09	7317.95	26.82

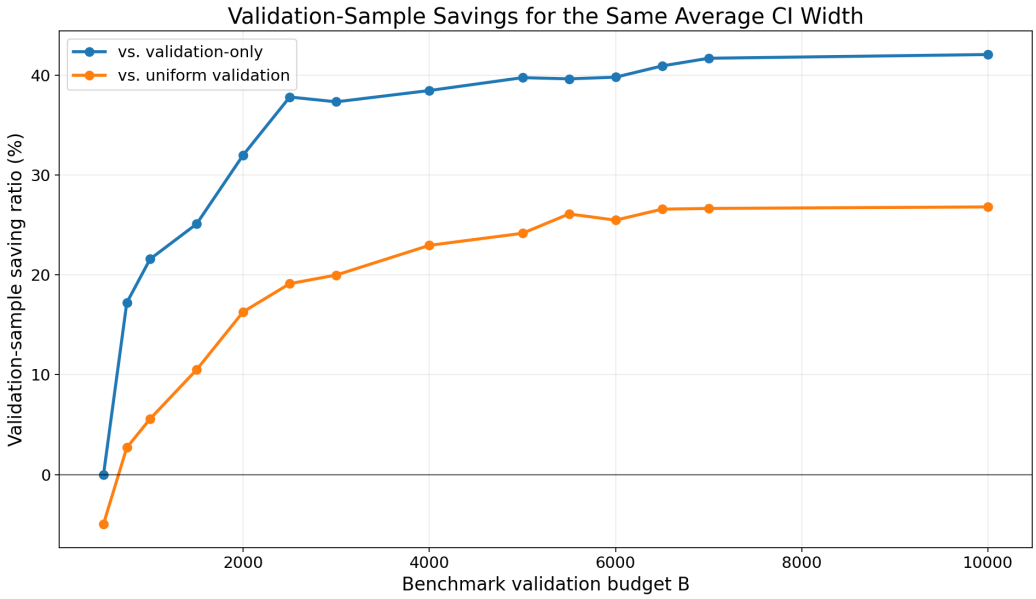

Figure 2: Sample-saving ratio of active validation when achieving the same average interval width.

References

1. Allon, Gad, Daniel Chen, Zhenling Jiang, and Dennis J. Zhang. 2023. Machine Learning and Prediction Errors in Causal Inference. The Wharton School Research Paper, forthcoming. Available at SSRN: `https://ssrn.com/abstract=4480696`.

2. Chen, Xiaohong, Han Hong, and Denis Nekipelov. 2011. Nonlinear Models of Measurement Errors. *Journal of Economic Literature* 49(4): 901–937.

3. Chen, Xiaohong, Han Hong, and Elie Tamer. 2005. Measurement Error Models with Auxiliary Data. *Review of Economic Studies* 72(2): 343–366.

4. Fong, Christian, and Matthew Tyler. 2020. Machine Learning Predictions as Regression Covariates. *Political Analysis* 29(4).

5. Fuller, Wayne A. 2009. *Measurement Error Models*. Hoboken, NJ: John Wiley & Sons.

6. Qiao, Mengke, and Ke-Wei Huang. 2021. Correcting Misclassification Bias in Regression Models with Variables Generated via Data Mining. *Information Systems Research* 32(2): 462–480.

7. Schennach, Susanne M. 2016. Recent Advances in the Measurement Error Literature. *Annual Review of Economics* 8: 341–377.

8. Wei, Max, and Nikhil Malik. 2026. Estimation Bias from Machine-Learned Features in Econometric Models. Working paper, University of Southern California.

9. Yang, Mochen, Gediminas Adomavicius, Gordon Burtch, and Yuqing Ren. 2018. Mind the Gap: Accounting for Measurement Error and Misclassification in Variables Generated via Data Mining. *Information Systems Research* 29(1): 4–24.

10. Zrnic, Tijana, and Emmanuel J. Candès. 2024. Active Statistical Inference. In *Proceedings of the 41st International Conference on Machine Learning*, PMLR 235: 62993–63010.

A Appendix

The parameter-contamination prediction model is trained on the existing validation sample, that is, the pilot sample. The target is to predict the magnitude of the sample's contamination of the estimator: $s_{i,j} = a_j' d_i(\widehat{\theta})$. The training target is set to $\log(s_{i,j}^2 + c)$, where $s_{i,j}^2$ is first winsorized at the 98% percentile, and $c = \max(0.05 \times median(s_{i,j}^2), 10^{-6})$. This is done to reduce the influence of extreme IPW targets on the random forest. The training parameters are as follows:

```
RandomForestRegressor(
  n_estimators=400,
  random_state=seed,
  max_depth=8,
  min_samples_leaf=50,
  max_features="sqrt",
  n_jobs=-1,
  bootstrap=True,
  oob_score=True,
)
```

The input features $\widehat{X}_i$ use only variables visible before validation. In the current version of the experiment, these mainly include control variables and the proxy prediction $\widehat{X}_i$. After training, the model outputs $\log(\widehat{s}^2_{i,j}+c)$, which is then transformed back to the risk scale: $\widehat{h}_i = \exp(\widehat{m}_i)-c$. The active sampling intensity is then $u_i = \sqrt{\widehat{h}_i}$.