VARIANCE-EFFICIENT ADJUSTMENT FOR HIGH-DIMENSIONAL CAUSAL INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Variable selection for causal inference typically prioritizes unbiasedness by identifying a valid adjustment set. In high-dimensional settings, where there are more covariates than samples, valid adjustment sets can be suboptimal in terms of treatment effect MSE, as they ignore the bias-variance trade-off. To address this, we propose VERA (Variance-Efficient Relaxed Adjustment), a variable selection component designed to wrap around any differentiable causal estimator. VERA selects variables via a differentiable gating mechanism, learning inclusion probabilities for each covariate. The selection is optimized to reduce the variance of the Efficient Influence Function, which prioritizes variables that explain more variation in the outcome than variation in the treatment. In our experiments, VERA improves the finite-sample estimation error of various base estimators in most settings, particularly in high-dimensional and small-sample regimes.

1 INTRODUCTION

To make causal claims from observational data, researchers must account for confounding bias, which is most commonly achieved by controlling for specific covariates. For example, when estimating the causal effect of a college degree on future earnings, it is necessary to control for socioeconomic background, since this is a confounder that drives the correlation between getting a college degree and future earnings away from the true causal effect. Rosenbaum & Rubin (1983) formalized this requirement into the condition of *ignorability*, which states that conditional on the control variables, the treatment assignment must be independent of the potential outcomes. Pearl (1993) developed graphical criteria that determine which covariates to select based on a known causal graph, such that the adjustment set is *valid*, i.e., the estimated causal effect with this set is unbiased.

If we do not know the causal graph, we can employ causal discovery to identify it from data, but most methods are computationally infeasible for larger numbers of variables (Chickering, 1996). Moreover, in high-dimensional settings, in which the number of covariates p is higher than the number of datapoints N, this estimation can be noisy. In these cases, even if we have a known causal graph, including all of the variables in a valid adjustment set might be suboptimal in terms of the MSE of the treatment effect in finite samples (Rutsch et al., 2025). Most current approaches for high-dimensional data tend to rely on sparsity assumptions or focus on achieving zero bias as the sample size grows to infinity (Kalisch & Bühlman, 2007; Chernozhukov et al., 2018; Belloni et al., 2014a). These methods do not give us a clear strategy for selecting variables when our goal is to minimize the finite-sample estimation error in high-dimensional or small-sample datasets.

Rather than striving for unbiasedness, we focus on variable selection that improves the finite-sample error of existing treatment effect estimators and shift the focus from eliminating bias to *minimizing the variance* of the estimator. We do this by learning a variable selection mechanism that minimizes a component of the variance of the Efficient Influence Function (EIF), as shown in Fig. 1. Our approach, VERA (Variance-Efficient Relaxed Adjustment), encourages dropping variables that inflate variance while keeping variables that explain outcome variation. Because VERA relies on a differentiable gating mechanism, it can be applied to any differentiable causal model, but our empirical evaluation focuses on neural treatment effect estimators. Our main contributions are:

- We propose VERA (Variance-Efficient Relaxed Adjustment), a variable selection component that uses a differentiable gating mechanism and can be combined with any differentiable causal estimator, aiming at reducing the finite sample MSE of the treatment effect.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

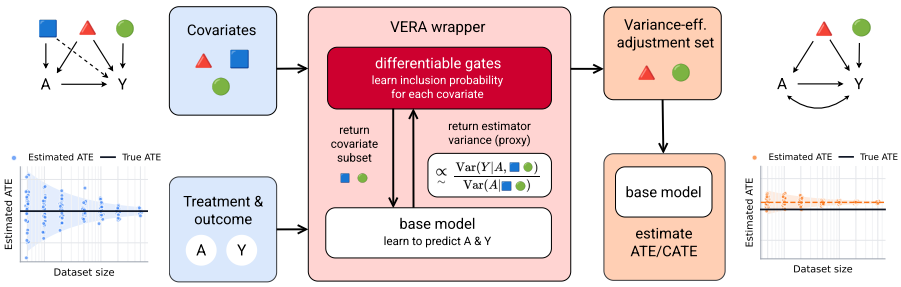


Figure 1: Illustration of VERA with toy example. VERA augments a base treatment-effect estimator with differentiable covariate gates for variable selection. By alternating updates of the base model and the gates, VERA aims to learn a variance-efficient adjustment set. The gates are optimized using a proxy for estimator variance derived from the Efficient Influence Function (EIF), which selects variables that reduce variation in the outcome while keeping variation in the treatment.

- We show theoretically that the empirical gate objective is a plug-in estimator of a residual EIF variance component under fixed masks and sample splitting.
- In our experiments, VERA improves the MSE of the treatment effect for various base estimators in most settings, especially in high-dimensional and small-sample regimes.

2 Preliminaries

We consider the problem of estimating the Average Treatment Effect (ATE) and Conditional Average Treatment Effect (CATE) from observational data, consisting of p-dimensional covariates $X = (X_1, \ldots, X_p) \in \mathbb{R}^p$, a binary treatment variable A, and a continuous outcome Y. These variables can be represented with a causal graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V} = \{X_1, \ldots, X_p, A, Y\}$ and causal relations are represented as directed edges $(V_k, V_l) \in \mathcal{E}$. A *path* is a sequence of edges connecting a sequence of distinct nodes. A path is *directed* if all edges point in the same direction, and *backdoor* if it starts with an arrow pointing into the treatment A. If there is a directed path from V_k to V_l, then V_l is a *descendant* of V_k, denoted $V_l \in \mathrm{de}(V_k)$. We assume that all covariates X are pre-treatment:

Assumption 2.1. All covariates X are not descendants of the treatment A, i.e., $\mathrm{de}(A) \cap X = \emptyset$.

Based on the causal structure, covariates fall into three distinct types: (1) instrumental variables (IVs) that affect the treatment but not directly the outcome, (2) confounders that affect both the treatment and the outcome, and (3) precision variables that affect only the outcome (Brookhart et al., 2006). For an example of a causal graph with all three types, see Fig. 2. Variables that strongly affect the treatment but only weakly affect the outcome are *Near-IVs* (Myers et al., 2011).

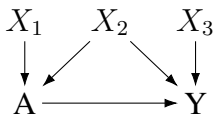


Figure 2: Causal graph with X_1 (IV), X_2 (confounder) and X_3 (precision variable).

In the potential outcomes framework (Rubin, 1974), a unit i with covariates $X_i = (X_{i1}, \ldots, X_{ip})$ has a potential outcome for each treatment assignment, denoted as $Y_i(0)$ for control and $Y_i(1)$ for treatment. For any given unit i, only the potential outcome corresponding to its treatment A_i is observed, such that $Y_i = A_i Y_i(1) + (1 - A_i) Y_i(0)$. The true CATE and ATE are defined as $\tau(x) = \mathbb{E}[Y(1) - Y(0) \mid X = x]$ and $\tau = \mathbb{E}[Y(1) - Y(0)]$. Estimands for CATE and ATE are identifiable from purely observational data under the standard assumptions of positivity and conditional ignorability. Throughout this paper, we assume that strong positivity, finite second moments of the outcome and conditional ignorability hold:

Assumption 2.2 (Strong Positivity)**.** There exists a constant $\epsilon > 0$ such that for all x in the support of X, the true propensity score is strictly bounded away from 0 and 1: $\epsilon < \Pr(A = 1 \mid X = x) < 1 - \epsilon$.

Assumption 2.3 (Finite second moments of the outcome)**.** We assume that $\mathbb{E}[Y^2] < \infty$.

Assumption 2.4 (Conditional ignorability)**.** All confounders are observed, i.e., potential outcomes are independent of the treatment assignment given the covariates: $\{Y(0), Y(1)\} \perp\!\!\!\perp A \mid X$.

Conditional ignorability corresponds to the absence of latent confounding and selection bias. A sufficient graphical condition for a covariate set X to satisfy conditional ignorability is the backdoor criterion, which requires that X blocks all backdoor paths between A and Y without blocking any directed paths from A to Y. In the example graph in Fig. 2, any set X that includes the confounder X_2 satisfies the backdoor criterion. When a covariate set satisfies this criterion, the CATE and the ATE can be identified via covariate adjustment as follows:

$$\tau(x) = \mathbb{E}[Y \mid A = 1, X = x] - \mathbb{E}[Y \mid A = 0, X = x], \qquad \tau = \mathbb{E}[\tau(X)].$$

When an adjustment set does not satisfy conditional ignorability (e.g. due to omitted confounders), estimating the treatment effect by conditioning on this set yields a biased result. To formalize this, we use the following notation. Let $m \in \{0,1\}^p$ be a binary mask indicating a chosen subset of covariates, where 1 represents that a covariate is included and 0 represents that a covariate is not included. Let X_m denote those selected variables. We define the conditional outcome mean for this subset as $\mu_{a,m}(x_m) = \mathbb{E}[Y \mid A = a, X_m = x_m]$ and the induced CATE and ATE estimands as:

$$\psi_m(x_m) = \mu_{1,m}(x_m) - \mu_{0,m}(x_m), \quad \psi(m) = \mathbb{E}\left[\mu_{1,m}(X_m) - \mu_{0,m}(X_m)\right].$$

If the subset X_m satisfies conditional ignorability, the induced CATE estimand equals the true CATE, i.e., $\psi_m(x_m) = \mathbb{E}[Y(1) - Y(0) \mid X_m = x_m]$, and, similarly, the induced ATE estimand equals the ATE, i.e., $\psi(m) = \tau$. Following Hahn (1998) and Tsiatis (2006), the Efficient Influence Function (EIF) variance for the induced ATE estimand $\psi(m)$ can be written as

$$V_{\text{EIF}}(m) = V_{\text{res}}(m) + \text{Var}\left(\psi_m(X_m)\right), \tag{1}$$

where $\text{Var}\left(\psi_m(X_m)\right)$ is the effect heterogeneity and $V_{\text{res}}(m)$ the *residual variance component*:

$$V_{\text{res}}(m) := \mathbb{E}\left[\frac{\sigma^2_{1,m}(X_m)}{e_m(X_m)} + \frac{\sigma^2_{0,m}(X_m)}{1 - e_m(X_m)}\right], \tag{2}$$

where $e_m(X_m) := \Pr(A = 1 \mid X_m)$ and $\sigma^2_{a,m}(X_m) := \text{Var}(Y \mid A = a, X_m)$. Equivalently, we can also write this residual variance component with the Riesz representer α:

$$V_{\text{res}}(m) = \mathbb{E}\left[(Y - \mu_{A,m}(X_m))^2 \alpha_m^2\right], \qquad \text{with} \quad \alpha_m := \frac{A}{e_m(X_m)} - \frac{1 - A}{1 - e_m(X_m)}. \tag{3}$$

The full EIF variance contains the treatment-effect heterogeneity term in Eq. (1). Under homogeneous treatment effects, $\text{Var}\left(\psi_m(X_m)\right) = 0$, so minimizing $V_{\text{res}}(m)$ minimizes the EIF variance of the ATE under valid adjustment. For the induced CATE estimand $\psi_m(x_m)$, the EIF variance is the conditional equivalent of $V_{\text{res}}(m)$ for $X_m = x_m$, which implies that in the conditional version of Eq. (1), the heterogeneity term is $\text{Var}(\psi_m(X_m) \mid X_m = x_m) = 0$. Hence, minimizing $V_{\text{res}}(m)$ minimizes the average CATE EIF variance under valid adjustment. We target this residual variance component V_{res}. While valid adjustment sets guarantee that the estimands are unbiased w.r.t. the true CATE and ATE, they may be suboptimal for finite-sample MSE when variance inflation outweighs bias reduction. Because of this, we aim for variance-efficient adjustment, which we introduce next.

3 VARIANCE-EFFICIENT ADJUSTMENT

In high-dimensional and small-sample regimes, variance inflation from adjusting for many covariates can contribute substantially to finite-sample estimation error. To address this, we introduce the concept of *variance-efficient* adjustment sets. We represent this set with a binary mask m, where 1 indicates inclusion of a variable. Specifically, we define the variance-efficient mask m^* as the one that minimizes the residual variance component $V_{\text{res}}(m^*)$ of the EIF.

Definition 3.1 (Variance-efficient adjustment set)**.** Let $\mathcal{M} = \{0,1\}^p$ be the space of all possible covariate masks. The variance-efficient set $m^* \in \mathcal{M}$ is the binary mask that includes the variables that minimize the residual variance component of the EIF in Eq. (2):

$$m^* = \arg\min_{m \in \mathcal{M}} V_{\text{res}}(m).$$

Note that m^* is not necessarily a valid adjustment set, since it is only optimized for variance reduction, and hence the treatment effect estimation might be biased. We now provide a worst-case bound for the omitted variable bias (OVB) in the ATE estimand that m^* could introduce.

Bias of the variance-efficient adjustment set. While m^* minimizes the residual EIF variance component $V_{\text{res}}(m)$, omitting covariates can introduce omitted-variable bias (OVB), denoted by $\text{Bias}(m^*)$. Let m^c denote a mask of omitted confounders such that the long mask $m^\ell := m^* \vee m^c$ is a valid adjustment mask, representing the union of the variance-efficient adjustment set and the dropped confounders. Note that while m^* may not be a valid adjustment, there always exists a valid extension m^ℓ of m^*, due to conditional ignorability (Ass. 2.4) and the fact that all covariates are pre-treatment (Ass. 2.1). Let the overlap constant ϵ_ℓ denote the largest constant such that $\epsilon_\ell \leq e_\ell(X_{m^\ell}) \leq 1 - \epsilon_\ell$ a.s., where $e_\ell(X_{m^\ell})$ is the propensity score. Under the strong positivity assumption (Ass. 2.2), such a constant exists and satisfies $\epsilon_\ell \geq \epsilon$. Following Chernozhukov et al. (2021), we bound the squared bias of $\psi(m^*)$ relative to the ATE by $V_{\text{res}}(m^*)$ and ϵ_ℓ:

Corollary 3.2 (Worst-case OVB bound)**.** *Under Ass. 2.1, 2.2 and 2.4, let $m^\ell = m^* \vee m^c$ be the valid long mask obtained by adding the omitted confounders m^c to m^*, such that $\psi(m^\ell) = \tau$. Let ϵ_ℓ be the largest constant satisfying $\epsilon_\ell \leq e_\ell(X_{m^\ell}) \leq 1 - \epsilon_\ell$ a.s. Then,*

$$\text{Bias}^2(m^*) \leq V_{\text{res}}(m^*) \frac{(1 - 2\epsilon_\ell)^2}{\epsilon_\ell(1 - \epsilon_\ell)}. \tag{4}$$

The proof is given in App. A. While the bound is conservative, it provides intuition about the scenarios in which variance-efficient adjustment may be appropriate. The bound contains two components. The first component, V_{res}, is minimized by variance-efficient adjustment. The second component, $((1 - 2\epsilon_\ell)^2)/(\epsilon_\ell(1 - \epsilon_\ell))$, depends on the overlap constant ϵ_ℓ under the valid long mask. When treatment assignment is sufficiently random under the valid long mask, ϵ_ℓ is not too small and the penalty is moderate. In the limiting case of complete randomization with $\epsilon_\ell = 1/2$, like in certain randomized controlled trials, the bound implies that variance-efficient adjustment is unbiased. However, when treatment is nearly deterministic for some units, ϵ_ℓ is close to zero and the factor becomes large. This result can provide an indication of whether a variance-efficient adjustment set provides a reasonable trade-off in terms of bias in a given dataset, if we know or can estimate the value of ϵ_ℓ reliably. However, this estimation might be difficult in practice, especially in high dimensions. In these cases, we propose a more flexible constrained variance-efficient adjustment notion, where the inclusion of important confounders is forced based on expert knowledge:

Constrained variance-efficient adjustment. Let $C \subseteq \{1, \ldots, p\}$ denote required covariates. We define the constrained variance-efficient mask as

$$m_C^* = \underset{m \in \mathcal{M}:\, C \subseteq \text{supp}(m)}{\arg\min} V_{\text{res}}(m),$$

where $\text{supp}(m) = \{j : m_j = 1\}$. $V_{\text{res}}(m)$ is minimized only over masks containing C. We next describe how to select covariates in practice when using differentiable base learners for treatment effect estimation, with the goal of improving finite-sample estimation error by learning to select variables that approximate the variance-efficient adjustment mask m^*, or the constrained mask m_C^*.

4 Implementation

We propose VERA (**V**ariance-**E**fficient **R**elaxed **A**djustment), a component that can be added to any differentiable causal effect estimator. VERA augments a base model with a differentiable mechanism that targets the variance-efficient adjustment set in Def. 3.1. Since we optimize locally, VERA learns an approximation of this set. To make the search space tractable in high dimensions, we assign independent inclusion probabilities to each covariate in a differentiable gating layer.

Differentiable gating layer and training procedure. Let $\hat{\mu}_a(X; \theta)$ be any differentiable outcome prediction model used in a causal estimator (e.g. prediction heads in Shalit et al., 2017), approximating the conditional outcome mean $\mu_a(x)$. VERA learns a covariate mask $m \in \{0, 1\}^p$ so that only the selected subset X_m (implemented as $m \odot X$) is passed to the base model, yielding $\hat{\mu}_{a,m}(X_m; \theta)$ that approximates $\mu_{a,m}(x_m)$. The mask is generated by a differentiable gating layer with one learnable logit ϕ_j per covariate $j \in [p]$, which we collect in the gate parameter $\phi = (\phi_1, \ldots, \phi_p)$. The gate activations which can be interpreted as inclusion probabilities are defined as $\pi_j = \sigma(\phi_j)$, where σ denotes the hard sigmoid function. Because the hard sigmoid saturates at exactly 0 and 1, it provides true variable selection. During early training cycles, the logits ϕ_j may lie in the linear region

of the hard sigmoid, yielding fractional values that can be interpreted as inclusion probabilities. To learn the base models and inclusion probabilities, we train by cycling through three stages:

1. Mask derivation: we consider the gate parameters ϕ from the previous iteration and compute the binary mask by thresholding the current gate probabilities, $m_j = \mathbb{1}\{\pi_j \geq 0.5\}$. For the initial cycle, we start with all covariates included ($m = \mathbf{1}$)
2. Model updates: using a training split of the data, we then fit the outcome model $\hat{\mu}_{a,m}(X_m; \theta)$ on the selected subset of covariates based on the mask m. In this phase, we also fit an auxiliary propensity model $\hat{e}_m(X_m; \eta)$, which is required for the gate objective.
3. Gate update: we freeze the parameters θ and η. On a separate validation split, we update the gate logits ϕ using the continuous hard sigmoid outputs π. In this stage, the gate parameters minimize an objective that targets $V_{\text{res}}(m)$, which we describe in the next paragraph.

The cycle loop terminates early when the gate probabilities stabilize. Using the mask derived from the converged probabilities, the prediction models of the base learner are trained one final time.

Gate update objective. Let $\hat{Y}_i = \hat{\mu}_{A_i,m}(X_{m,i}; \theta)$ be the outcome prediction for unit i, and $\hat{r}_i = Y_i - \hat{Y}_i$ the residuals. We clip the predicted propensities to $[\delta, 1-\delta]$, for a small δ, e.g. 10^{-6}, and denote it as $\hat{e}_{\text{clip}}(X_{m,i})$. We denote the clipped inverse probability weights as

$$\hat{w}_i = A_i / \hat{e}_{\text{clip}}(X_{m,i})^2 + (1 - A_i)/(1 - \hat{e}_{\text{clip}}(X_{m,i}))^2.$$

This weight is the empirical counterpart of $\alpha_m^2 = A/e^2 + (1-A)/(1-e)^2$ for binary treatment, so $\widehat{V}_n := \frac{1}{n}\sum_{i=1}^n \hat{w}_i \hat{r}_i^2$ is the plug-in analogue of $V_{\text{res}}(m)$ with clipped propensities for numerical stability. The gates are trained by minimizing the empirical plug-in of V_{res}:

$$\mathcal{L}_{\text{VERA}} = \log\left(\widehat{V}_n + \varepsilon_{\text{num}}\right) = \log\left(\frac{1}{n}\sum_{i=1}^n \hat{w}_i \hat{r}_i^2 + \varepsilon_{\text{num}}\right), \tag{5}$$

where $\varepsilon_{\text{num}} = 10^{-6}$ is a small constant added for numerical stability. By minimizing this objective, the gating layer learns to drop covariates that inflate the propensity score denominators (such as Near-IVs). VERA's objective targets the $V_{\text{res}}(m)$ via plug-ins from the base model. This aligns VERA with the goal of finding the variance-efficient adjustment set as introduced in Sec. 3.

Proposition 4.1 (Plug-in convergence of $\mathcal{L}_{\text{VERA}}$ at a fixed mask)**.** *Fix a binary mask $m \in \{0,1\}^p$. Let $(X_i, A_i, Y_i)_{i=1}^n$ be an i.i.d. validation sample of size n, independent of the training sample of size N used to fit the nuisance plug-ins $(\hat{\mu}_{0,m}, \hat{\mu}_{1,m}, \hat{e}_m)$. Assume Ass. 2.2 and 2.3 and that the clipping level is chosen outside the positivity margin (Ass. B.1), the nuisance plug-ins are $L_2(P)$-consistent (Ass. B.2) and there is sample splitting for the gate objective (Ass. B.3). As the validation sample size $n \to \infty$ and the training sample size $N \to \infty$, the plug-in analogue $\widehat{V}_n$ converges to the residual variance component $V_{\text{res}}(m)$ in probability, i.e., $\widehat{V}_n \xrightarrow{P} V_{\text{res}}(m)$. Moreover, if $\varepsilon_{\text{num}} > 0$, then by the continuous mapping theorem,*

$$\mathcal{L}_{\text{VERA}} = \log\left(\widehat{V}_n + \varepsilon_{\text{num}}\right) \xrightarrow{P} \log\left(V_{\text{res}}(m) + \varepsilon_{\text{num}}\right).$$

The proof is in App. B. Intuitively, for a fixed mask and an independent validation split, the gate loss converges in probability to V_{res}, up to the small constant ε_{num}. We further show in App. C that the criterion evaluated on a held-out validation split reflects both the residual EIF variance and finite-sample outcome estimation error, with the propensity estimation error reweighting both terms.

Final estimation. Once the VERA cycles are complete and the gate probabilities have converged, we extract the final binary mask $\hat{m} = \mathbb{1}\{\pi \geq 0.5\}$. The outcome and propensity models are trained one final time from scratch using the fixed subset $X_{\hat{m}}$. The final CATE and ATE are estimated as:

$$\widehat{\tau}_{\text{VERA}}(X_i) = \hat{\mu}_{1,\hat{m}}(X_{\hat{m},i}) - \hat{\mu}_{0,\hat{m}}(X_{\hat{m},i}), \quad \widehat{\tau}_{\text{VERA}} = \frac{1}{N}\sum_{i=1}^N \left[\hat{\mu}_{1,\hat{m}}(X_{\hat{m},i}) - \hat{\mu}_{0,\hat{m}}(X_{\hat{m},i})\right].$$

Adding background knowledge. VERA can implement constrained variance-efficient adjustment by forcing certain covariates to be included. Specifically, if some variables are known to be important confounders, the search space of masks can be restricted to masks that contain these variables, which prevents VERA from dropping them. In practice, this is implemented by fixing the corresponding gate probabilities to one and optimizing only the remaining gate parameters.

5 Related work

With the advancements of deep learning, highly flexible methods for causal effect estimation using neural networks have been developed. Double Machine Learning (DML) (Chernozhukov et al., 2018) provides consistency guarantees under specific convergence rates for the underlying neural networks. Several neural models have been proposed for estimating the CATE, e.g., TARNet and Counterfactual Regression (CFR) (Shalit et al., 2017), which combine an S-Learner to learn a shared covariate representation with T-structured prediction heads, as well as numerous follow-up works (e.g. Shi et al., 2019; Yao et al., 2018). These methods usually use all covariates for adjustment, whereas VERA selects a subset based on the residual EIF variance component V_{res}.

The literature on adjustment set selection has a rich history, rooted in concepts such as ignorability (Rosenbaum & Rubin, 1983) and the back-door criterion (Pearl, 1993). While this has historically led to the practice of adjusting for all available pre-treatment covariates, including pre-treatment colliders can bias the effect estimate (Sjölander, 2009; Liu et al., 2012). For valid adjustment set selection based on the causal graph, Shpitser et al. (2010) and Perković et al. (2018) introduced complete graphical criteria. Multiple methods also optimize the variance of treatment effect estimators within valid adjustment sets. It is a well-established guideline to include pre-treatment variables that are highly predictive of the outcome, while avoiding instrumental variables (Bhattacharya & Vogt, 2007; Wooldridge, 2016). In high-dimensional settings, outcome-adaptive Lasso (OAL) (Shortreed & Ertefaie, 2017) and post-double-selection Lasso (PDS) (Belloni et al., 2014b) use regularization to select covariates for causal effect estimation. Henckel et al. (2022) proposed a graphical criterion for identifying the asymptotically optimal adjustment set $\mathbf{O}$ among all valid sets in linear models. Rotnitzky & Smucler (2020) showed this criterion also works in nonparametric settings and that it gives the lowest asymptotic variance of the Efficient Influence Function (EIF). Recently, Rutsch et al. (2025) demonstrated that in finite-sample settings, theoretically invalid or biased adjustment sets can outperform the optimal adjustment set $\mathbf{O}$. However, their results are limited to linear Gaussian causal models, and their strategies for adjustment set selection fail in high-dimensional settings, as they rely on iterating over a large number of candidate adjustment sets. Here we propose a solution to this by replacing covariate selection with a continuous inclusion probability.

Semiparametric efficiency (Hahn, 1998) characterizes the lowest asymptotic variance of regular estimators through Efficient Influence Functions (EIF). DML (Chernozhukov et al., 2018) and TMLE (van der Laan & Rubin, 2006) use the EIF for orthogonal estimation. In our work, werepurpose an EIF variance term as a pragmatic, differentiable optimization objective for covariate selection.

6 Experiments

We evaluate VERA across synthetic, semi-synthetic, and real-world benchmarks. to assess its impact on finite-sample estimation error across varying sample sizes and dimensionalities.

Synthetic data. We design a nonlinear Data Generating Process (DGP) to evaluate VERA's ability to exclude variance-inflating variables. We consider $N = 50$ and $p \in \{10, 100\}$ across 1,000 random seeds. Using a polynomial basis $B_j = X_j + X_j^2$, we generate: $A_i \sim \text{Bern}(\text{expit}(\gamma \tilde{a}_i))$ and $Y_i = \sum_{j=1}^{p} \beta_j^{(Y)} B_{ij} + \tau A_i + \varepsilon_i^{(Y)}$, where $\tilde{a}_i$ is a centered and shifted version of $\sum_j \beta_j^{(A)} B_{ij}$, $\tau = 3$ is the ATE, $\gamma = 0.1$ is a scaling factor and $\varepsilon^{(Y)}$ is a standard Gaussian noise variable. We consider two settings: one with independent and one with correlated covariates. We calibrate coefficients β such that $\beta_j^{(A)} \beta_j^{(Y)} = 1$ remains constant for all covariates, but vary the ratio $q_j = \beta_j^{(A)} / \beta_j^{(Y)} \in [0.05, 20]$. This ensures the confounders range from near-IVs (high q_j) to precision variables (low q_j) with roughly equal OVB, as illustrated in App. D.1, where we also describe the detailed experimental settings.

Here the full adjustment set coincides with the optimal adjustment set **O**, since all covariates are parents of the outcome. We report ATE RMSE against the population τ.

In the synthetic setting with $p = 10$ independent covariates, we also run an Oracle OLS estimator using the nonlinear basis B_j as features, instead of the covariates X_j, which means the model is now linear. We evaluate all possible adjustment sets m in terms of their MSE and the estimated EIF variance term $V_{\text{res}}(m)$. As shown in Fig. 3, $V_{\text{res}}(m)$ is positively correlated with the MSE of the ATE, suggesting that VERA's objective of minimizing a plug-in analogue for $V_{\text{res}}(m)$ aligns well with minimizing finite-sample error.

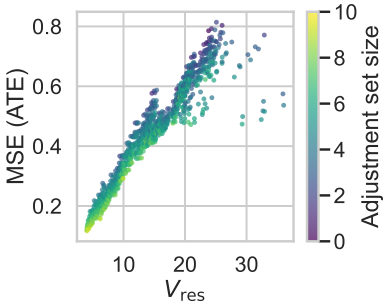


Figure 3: MSE of the ATE depends on the value of the V_{res} for each adjustment set using the OLS oracle.

Semi-synthetic data. We evaluate VERA on IHDP (Hill, 2011) (1,000 replications, $p = 25, N = 672$). To evaluate performance where variable selection is important, we subsample the training sets at $N \in \{50, 100, 200, 300, 400, 500, 600, 672\}$. We report the CATE RMSE ($\sqrt{\epsilon_{\text{PEHE}}}$) and the ATE MAE ($\epsilon_{\text{ATE}}$). To test VERA under complex non-linearities and poor overlap, we use the ACIC 2018 scaling track (Shimoni et al., 2018) ($p = 177$). We use 432 datasets with $N = 1000$ (80/20 split) and a high-dimensional subsampled version with $N = 50$ per replication.

Real-world data. We use the Jobs dataset (LaLonde, 1986; Shalit et al., 2017) ($p = 8, N = 2935$) as a low-dimensional stress test, since this is a setting where full adjustment is generally preferable. We use the version with a binary outcome (employment status) and report ATT MAE (ϵ_{ATT}) and policy values (R_{pol}) as in Shalit et al. (2017) (see App. D.2).

Base models and baselines. To evaluate the effectiveness and applicability of VERA, we apply it to four standard meta-learning architectures implemented as neural networks: (1) the *S-Learner*, which uses a single network to model the outcome with the treatment as an additional feature; (2) the *T-Learner* (Künzel et al., 2019), which fits two independent networks for each treatment group; (3) the *R-Learner* (Nie & Wager, 2021), which uses Robinson residuals to target the CATE directly through a decomposition of the objective; and (4) *TARNet* (Shalit et al., 2017), which uses a shared representation (S-structure) followed by treatment-specific heads (T-structure).

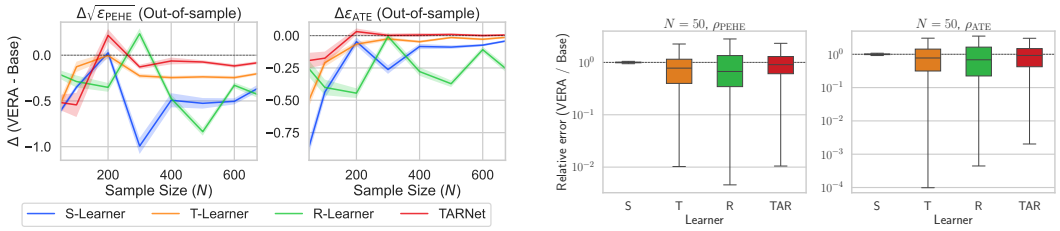
We compare these VERA versions to the corresponding base learners and to additional baselines: BART (Chipman et al., 2010), causal forest (Wager & Athey, 2018), IPW (Rosenbaum & Rubin, 1983; Hirano et al., 2003), and DML (with XGBoost) (Chernozhukov et al., 2018; Chen & Guestrin, 2016), as well as OLS and Lasso with both S- and T-structures, outcome-adaptive Lasso (OAL) (Shortreed & Ertefaie, 2017), and post-double-selection Lasso (PDS) (Belloni et al., 2014b) with both the Belloni-Chernozhukov-Hansen plug-in penalty and a cross-validated penalty. DML and IPW target the ATE only and do not estimate CATE. Therefore, we assign the estimated ATE as the predicted CATE for all units. Since Jobs is low-dimensional and has a binary outcome, we use logistic regression instead of the OLS and Lasso baselines. Implementation details are in App. E and code is available at `https://anonymous.4open.science/r/vera-iclr-7B76/`.

6.1 Results

Treatment effect estimation. As shown in Table 1, VERA improves upon all base learners in all synthetic settings. For the high-dimensional settings with $p = 100$, VERA TARNet outperforms all baselines. T-structured learners (including TARNet) seem to outperform R- and S-Learners in these settings. This is likely because the S-Learner fails to isolate the treatment effect from the covariate effects in high dimensions, and the R-Learner suffers from compounding errors when estimating high-dimensional nuisance models. For the low-dimensional settings with $p = 10$, other methods like DML, BART and PDS (CV) outperform VERA methods. We additionally evaluate a VERA TARNet variant with added background knowledge, denoted *VERA TAR.+BK*, in which the last covariate is forced to remain included. In this DGP, this covariate is known to be strongly outcome-predictive and weakly treatment-predictive, which makes it a good adjustment variable.

Table 1: ATE RMSE (with standard errors) and ratio between absolute bias $|\text{Bias}| = |\mathbb{E}[\hat{\tau} - \tau]|$, and variance $\text{Var}(\hat{\tau})$ for synthetic data. The lowest RMSE is bolded, the second lowest is underlined.

	Independent, p=10		Independent, p=100		Correlated, p=10		Correlated, p=100	
Method	RMSE	$\lvert\text{Bias}\rvert/\text{Var}$	RMSE	$\lvert\text{Bias}\rvert/\text{Var}$	RMSE	$\lvert\text{Bias}\rvert/\text{Var}$	RMSE	$\lvert\text{Bias}\rvert/\text{Var}$
OLS/S	0.76 ± 0.02	$0.32/0.48$	2.30 ± 0.01	$2.29/0.04$	0.76 ± 0.02	$0.38/0.43$	2.22 ± 0.01	$2.21/0.06$
OLS/T	0.89 ± 0.02	$0.40/0.62$	1.11 ± 0.02	$0.88/0.45$	0.84 ± 0.02	$0.44/0.52$	1.08 ± 0.02	$0.86/0.44$
Lasso/S	0.76 ± 0.02	$0.39/0.42$	1.14 ± 0.02	$0.90/0.49$	0.78 ± 0.02	$0.47/0.38$	1.14 ± 0.02	$0.90/0.51$
Lasso/T	0.81 ± 0.02	$0.45/0.46$	1.15 ± 0.02	$0.92/0.46$	0.83 ± 0.02	$0.54/0.39$	1.16 ± 0.02	$0.94/0.45$
IPW	0.79 ± 0.02	$0.38/0.48$	1.13 ± 0.02	$0.91/0.44$	0.82 ± 0.02	$0.49/0.44$	1.19 ± 0.02	$0.99/0.44$
DML	$\mathbf{0.64} \pm 0.01$	$0.02/0.41$	0.85 ± 0.02	$0.39/0.58$	$\mathbf{0.61} \pm 0.01$	$0.04/0.37$	0.89 ± 0.02	$0.45/0.60$
BART	$\underline{0.72} \pm 0.02$	$0.39/0.36$	1.12 ± 0.02	$0.90/0.43$	$\underline{0.69} \pm 0.01$	$0.42/0.29$	1.14 ± 0.02	$0.95/0.40$
CausalFor.	0.78 ± 0.02	$0.43/0.42$	1.12 ± 0.02	$0.91/0.44$	0.81 ± 0.02	$0.54/0.37$	1.18 ± 0.02	$0.98/0.43$
OAL	0.80 ± 0.02	$0.40/0.48$	0.82 ± 0.02	$0.32/0.57$	0.81 ± 0.02	$0.47/0.43$	0.81 ± 0.02	$0.40/0.50$
PDS (BCH)	0.83 ± 0.02	$0.45/0.49$	1.13 ± 0.02	$0.92/0.44$	0.93 ± 0.02	$0.61/0.51$	1.20 ± 0.02	$1.00/0.45$
PDS (CV)	0.75 ± 0.02	$0.30/0.48$	4.27 ± 0.64	$0.39/18.11$	0.74 ± 0.02	$0.36/0.42$	5.14 ± 1.29	$0.81/25.76$
R-Learner	1.46 ± 0.02	$1.18/0.74$	1.80 ± 0.02	$1.66/0.48$	1.39 ± 0.02	$1.08/0.76$	1.99 ± 0.02	$1.89/0.41$
VERA R.	0.92 ± 0.02	$0.38/0.71$	1.35 ± 0.02	$1.07/0.66$	0.88 ± 0.02	$0.36/0.64$	1.56 ± 0.02	$1.40/0.49$
S-Learner	1.51 ± 0.02	$1.00/1.28$	2.79 ± 0.00	$2.79/0.02$	1.34 ± 0.03	$0.84/1.09$	2.77 ± 0.00	$2.77/0.02$
VERA S.	0.86 ± 0.02	$0.05/0.73$	2.49 ± 0.00	$2.48/0.03$	0.73 ± 0.02	$0.09/0.52$	2.51 ± 0.01	$2.50/0.03$
T-Learner	1.09 ± 0.02	$0.29/1.11$	1.40 ± 0.03	$1.09/0.78$	1.03 ± 0.02	$0.15/1.03$	1.38 ± 0.02	$1.15/0.59$
VERA T.	0.85 ± 0.02	$0.25/0.66$	0.80 ± 0.02	$0.54/0.34$	0.76 ± 0.02	$0.24/0.53$	0.91 ± 0.02	$0.72/0.31$
TARNet	1.08 ± 0.02	$0.20/1.13$	1.19 ± 0.02	$0.65/1.00$	0.98 ± 0.02	$0.07/0.96$	1.15 ± 0.02	$0.76/0.73$
VERA TAR.	0.86 ± 0.02	$0.27/0.66$	$\underline{0.74} \pm 0.02$	$0.10/0.54$	0.76 ± 0.02	$0.27/0.51$	$\underline{0.75} \pm 0.02$	$0.28/0.49$
+BK	0.83 ± 0.02	$0.26/0.63$	$\mathbf{0.74} \pm 0.02$	$0.10/0.53$	0.74 ± 0.02	$0.26/0.49$	$\mathbf{0.74} \pm 0.02$	$0.27/0.48$



(a) IHDP absolute error differences.

(b) ACIC 2018 relative out-of-sample error.

Figure 4: Error comparison of VERA against base learners on semi-synthetic benchmarks. (a) Averaged per-experiment differences in error on the IHDP dataset across sample sizes; values below zero indicate VERA outperforms the base learner. (b) Ratio of VERA error to baseline error for PEHE and ATE on ACIC 2018, denoted as ρ_{PEHE} and ρ_{ATE}. Values below 10^0 indicate a lower error for VERA. Outliers are omitted for clarity but included in quartile calculations.

For IHDP, we report the averaged per-experiment differences of $\sqrt{\epsilon_{\text{PEHE}}}$ and ϵ_{ATE} across different sample sizes in Fig. 4a. VERA improves ATE and CATE estimation performance across almost all settings, with a tendency to make larger improvements at smaller sample sizes. For the full results (for $N = 50$ and $N = 672$) including all baselines, see Table 3 in App. F.

The ACIC 2018 benchmark is a highly challenging setting with large heterogeneity, poor overlap and complex nonlinear functions. Since errors vary in scale, we report the relative out-of-sample errors for $N = 50$ in Fig. 4b. For relative out-of-sample errors at $N = 1000$, and for versions that show the full range of ratios (including extreme values), see App. F. Because many ACIC replications have small treatment effects relative to noise, the S-Learner can do well with near-zero CATEs, so relative metrics for S-Learner are very noisy. In Fig. 4b, the median relative out-of-sample error is below 1 for the T- and R-Learner as well as TARNet, indicating that VERA typically improves upon these base learners. However, some replications show large relative worsening. When stratifying the ACIC datasets by overlap, VERA's advantage is strongest in the best-overlap tercile and smallest in the worst-overlap tercile (see App. G.2). Similarly, the ACIC 2018 results as suggest a typical relative improvement upon base models, but with non-negligible tail risk.

Table 2 shows the results for VERA and the corresponding base models on the Jobs dataset. Full baseline results are provided in App. F. As expected, VERA does not improve performance here,

Table 2: Performance comparison between base learners and VERA variants on Jobs. Results are reported as mean ± standard error across 100 replications.

	R-Learner		S-Learner		T-Learner		TARNet	
Metric	Base	VERA	Base	VERA	Base	VERA	Base	VERA
Within-sample								
ϵ_{ATT}	$.07 \pm .00$	$.09 \pm .00$	$.07 \pm .00$	$.06 \pm .01$	$.07 \pm .01$	$.06 \pm .01$	$.04 \pm .00$	$.05 \pm .00$
R_{pol}	$.23 \pm .00$	$.26 \pm .00$	$.26 \pm .00$	$.27 \pm .00$	$.22 \pm .00$	$.26 \pm .00$	$.20 \pm .00$	$.24 \pm .00$
Out-of-sample								
ϵ_{ATT}	$.11 \pm .01$	$.13 \pm .01$	$.12 \pm .01$	$.11 \pm .01$	$.12 \pm .01$	$.11 \pm .01$	$.11 \pm .01$	$.11 \pm .01$
R_{pol}	$.25 \pm .01$	$.29 \pm .01$	$.26 \pm .01$	$.28 \pm .01$	$.25 \pm .01$	$.28 \pm .01$	$.24 \pm .01$	$.29 \pm .01$

since the sample size is large relative to the number of covariates. On the other hand, VERA remains competitive with each base learner for estimating the ATT with a small increase in policy risk.

Selection behavior. In Fig. 5, we show the averages of VERA's learned inclusion probabilities for each covariate in the synthetic setting with independent covariates and $p = 10$. As an oracle, we show the probability of each covariate being in the estimated variance-efficient adjustment set when we compare all possible adjustment sets using OLS with the oracle features. We see that the learned inclusion probabilities follow a similar curve as the oracle probabilities. This shows that VERA works as intended. In a sparse DGP with pure IVs, confounders, precision variables, and noise variables, VERA qualitatively shows the intended selection behavior, as it keeps precision variables most often, followed by confounders and then IVs and noise variables (see App. G.3). For an analysis of the learned inclusion probabilities on the Jobs dataset, see App. F.

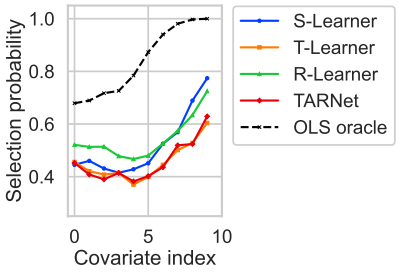


Figure 5: VERA's selection probabilities of each variable on synthetic data with $p = 10$ independent covariates.

Additional analyses. We investigate the regimes in which VERA is most beneficial, as well as the effects of omitting confounders in Apps. G.1 and G.3. In the synthetic DGP, VERA improves upon its T-Learner and TARNet base models for N/p up to roughly 2, with the largest gains at $N/p \leq 1$. In an adversarial DGP, where the population V_{res} criterion prefers dropping a strong confounder although keeping it is MSE-optimal, at $p = 100$ VERA with the T-Learner remains competitive with the full-adjustment base learner, matches or outperforms OAL and PDS in terms of RMSE, and drops the confounder less often than OAL and PDS with the BCH penalty. Additionally, we show that a single strong confounder is dropped less often when it is strongly outcome-predictive. We also examine VERA's optimization in App. G.4, including selection stability, variable re-entry, and the mismatch between soft and hard gates.

7 Conclusions and Discussion

We proposed VERA, a wrapper that selects covariates to reduce a variance term derived from the Efficient Influence Function, with the goal of improving finite-sample error. While unbiasedness of the estimator is a standard property in causal inference, our results show that relaxing this requirement can improve the MSE of treatment effect estimators in high-dimensional and small-sample regimes.

On the other hand, our work is a first step towards improving MSE in these settings and as such it still has several limitations. For example, minimizing V_{res} can drop crucial confounders even when keeping them is better in terms of MSE. Notably, this also occurs for OAL and PDS in our adversarial experiment (see App. G.3). Moreover, the gate objective is evaluated using nuisance and outcome models fit *under the current mask*, so the estimated residual EIF variance might not coincide with the value one would obtain after fully retraining at every alternative mask. Accordingly, we should expect only local improvement of the estimate (and no guarantee of reaching a globally variance-minimizing mask). Mitigating these limitations is an interesting direction for future work.

AI USE STATEMENT

In this work, we used generative AI tools to derive the finite-sample decomposition in App. C and its proof, to assist with the proof of Prop. 4.1, to get feedback on the methodology and theory, and to help design the additional analyses in App. G. On top of a manual literature review, we used generative AI tools to find relevant literature. Some parts of the text were drafted by generative AI tools, such as App. E and App. G, and some author-drafted parts were edited with generative AI tools. The methods and experiments were implemented by an AI coding agent.

We have not used generative AI tools to generate the main research idea, including the VERA method, to interpret results or to generate data beyond implementing the synthetic DGPs. Translation and qualitative data analysis are not applicable to this work. All method implementations that do not rely on widely used standard implementations, or that deviate from them, were checked manually against their descriptions or reference implementations, and all implementations were additionally cross-checked with several generative AI tools.

All proofs were checked manually by at least two authors and edited for clarity. The AI-drafted parts of the text were also checked for correctness and edited for clarity. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work aims to advance the field of Causal Machine Learning. The data, methodology, and results in this paper do not pose safety risks and do not raise questions or concerns regarding the Code of Ethics.

REPRODUCIBILITY STATEMENT

All experimental results presented in this work can be reproduced using the code available at `https://anonymous.4open.science/r/vera-iclr-7B76/`, following the instructions given in the repository (including data access and preprocessing). Details on the implementation are also described in App. E. All theoretical results presented in this work are supported by proofs in the appendix (see Apps. A to C, with all necessary assumptions stated.

REFERENCES

Jason Ansel, Edward Yang, Horace He, Natalia Gimelshein, Animesh Jain, Michael Voznesensky, Bin Bao, Peter Bell, David Berard, Evgeni Burovski, Geeta Chauhan, Anjali Chourdia, Will Constable, Alban Desmaison, Zachary DeVito, Elias Ellison, Will Feng, Jiong Gong, Michael Gschwind, Brian Hirsh, Sherlock Huang, Kshiteej Kalambarkar, Laurent Kirsch, Michael Lazos, Mario Lezcano, Yanbo Liang, Jason Liang, Yinghai Lu, CK Luk, Bert Maher, Yunjie Pan, Christian Puhrsch, Matthias Reso, Mark Saroufim, Marcos Yukio Siraichi, Helen Suk, Michael Suo, Phil Tillet, Eikan Wang, Xiaodong Wang, William Wen, Shunting Zhang, Xu Zhao, Keren Zhou, Richard Zou, Ajit Mathews, Gregory Chanan, Peng Wu, and Soumith Chintala. PyTorch 2: Faster machine learning through dynamic python bytecode transformation and graph compilation. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, ASPLOS '24. ACM, April 2024. doi: 10.1145/3620665.3640366. URL `https://docs.pytorch.org/assets/pytorch2-2.pdf`.

Alexandre Belloni, Victor Chernozhukov, and Christian Hansen. High-dimensional methods and inference on structural and treatment effects. *Journal of Economic Perspectives*, 28(2):29–50, 2014a.

Alexandre Belloni, Victor Chernozhukov, and Christian Hansen. Inference on treatment effects after selection among high-dimensional controls. *The Review of Economic Studies*, 81(2):608–650, 04 2014b. ISSN 0034-6527. doi: 10.1093/restud/rdt044. URL `https://doi.org/10.1093/restud/rdt044`.

Jay Bhattacharya and William B. Vogt. Do instrumental variables belong in propensity scores? NBER Technical Working Papers 0343, National Bureau of Economic Research, Inc, Sep 2007. URL https://ideas.repec.org/p/nbr/nberte/0343.html.

M Alan Brookhart, Sebastian Schneeweiss, Kenneth J Rothman, Robert J Glynn, Jerry Avorn, and Til Stürmer. Variable selection for propensity score models. *American journal of epidemiology*, 163(12):1149–1156, 2006.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 02 2018. ISSN 1368-4221. doi: 10.1111/ectj.12097. URL https://doi.org/10.1111/ectj.12097.

Victor Chernozhukov, Carlos Cinelli, Whitney K Newey, Amit Sharma, and Vasilis Syrgkanis. Long story short: Omitted variable bias in causal machine learning. *The Review of Economics and Statistics*, pp. 1–45, 2021.

David Maxwell Chickering. Learning bayesian networks is np-complete. In *Learning from data: Artificial intelligence and statistics V*, pp. 121–130. Springer, 1996.

Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. Bart: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298, 2010. ISSN 19326157, 19417330. URL http://www.jstor.org/stable/27801587.

Rajeev H Dehejia and Sadek Wahba. Causal effects in nonexperimental studies: Reevaluating the evaluation of training programs. *Journal of the American statistical Association*, 94(448):1053–1062, 1999.

Vincent Dorie, Hugh Chipman, and Robert McCulloch. *dbarts: Discrete Bayesian Additive Regression Trees Sampler*, 2026. URL https://CRAN.R-project.org/package=dbarts. R package version 0.9-33.

Laurent Gautier. rpy2, 2023. URL https://github.com/rpy2/rpy2/releases/tag/RELEASE_3_6_6.

Jinyong Hahn. On the role of the propensity score in efficient semiparametric estimation of average treatment effects. *Econometrica*, pp. 315–331, 1998.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi: 10.1038/s41586-020-2649-2. URL https://doi.org/10.1038/s41586-020-2649-2.

Leonard Henckel, Emilija Perković, and Marloes H Maathuis. Graphical criteria for efficient total effect estimation via adjustment in causal linear models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(2):579–599, 2022.

Jennifer L Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011.

Keisuke Hirano, Guido W Imbens, and Geert Ridder. Efficient estimation of average treatment effects using the estimated propensity score. *Econometrica*, 71(4):1161–1189, 2003.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3): 90–95, 2007. doi: 10.1109/MCSE.2007.55.

Markus Kalisch and Peter Bühlman. Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *Journal of Machine Learning Research*, 8(3), 2007.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017. URL https://arxiv.org/abs/1412.6980.

Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the national academy of sciences*, 116(10):4156–4165, 2019.

Robert J LaLonde. Evaluating the econometric evaluations of training programs with experimental data. *The American economic review*, pp. 604–620, 1986.

Wei Liu, M Alan Brookhart, Sebastian Schneeweiss, Xiaojuan Mi, and Soko Setoguchi. Implications of m bias in epidemiologic studies: a simulation study. *American journal of epidemiology*, 176 (10):938–948, 2012.

Jessica A Myers, Jeremy A Rassen, Joshua J Gagne, Krista F Huybrechts, Sebastian Schneeweiss, Kenneth J Rothman, Marshall M Joffe, and Robert J Glynn. Effects of adjusting for instrumental variables on bias and precision of effect estimates. *American journal of epidemiology*, 174(11): 1213–1222, 2011.

Xinkun Nie and Stefan Wager. Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, 108(2):299–319, 2021.

The pandas development team. pandas-dev/pandas: Pandas, February 2020. URL https://doi.org/10.5281/zenodo.3509134.

Judea Pearl. Comment: Graphical models, causality and intervention. *Statistical Science*, 8(3): 266–269, 1993. ISSN 08834237. URL http://www.jstor.org/stable/2245965.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Emilija Perković, Johannes Textor, Markus Kalisch, and Marloes H Maathuis. Complete graphical characterization and construction of adjustment sets in markov equivalence classes of ancestral graphs. *Journal of Machine Learning Research*, 18(220):1–62, 2018.

Paul R Rosenbaum and Donald B Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

Andrea Rotnitzky and Ezequiel Smucler. Efficient adjustment sets for population average causal treatment effect estimation in graphical models. *Journal of Machine Learning Research*, 21(188): 1–86, 2020.

Donald B Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688, 1974.

Nadja Rutsch, Sara Magliacane, and Stéphanie van der Pas. Finite sample-optimal adjustment sets in linear gaussian causal models, 2025. URL https://arxiv.org/abs/2506.12869.

Uri Shalit, Fredrik D Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference on machine learning*, pp. 3076–3085. PMLR, 2017.

Claudia Shi, David Blei, and Victor Veitch. Adapting neural networks for the estimation of treatment effects. *Advances in neural information processing systems*, 32, 2019.

Yishai Shimoni, Chen Yanover, Ehud Karavani, and Yaara Goldschmnidt. Benchmarking framework for performance-evaluation of causal inference analysis, 2018. URL https://arxiv.org/abs/1802.05046.

Yishai Shimoni, Ehud Karavani, Sivan Ravid, Peter Bak, Tan Hung Ng, Sharon Hensley Alford, Denise Meade, and Yaara Goldschmidt. An evaluation toolkit to guide model selection and cohort definition in causal inference, 2019. URL https://arxiv.org/abs/1906.00442.

Susan M. Shortreed and Ashkan Ertefaie. Outcome-adaptive lasso: Variable selection for causal inference. *Biometrics*, 73(4):1111–1122, 2017. doi: 10.1111/biom.12679.

Ilya Shpitser, Tyler VanderWeele, and James M. Robins. On the validity of covariate adjustment for estimating causal effects. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, UAI'10, pp. 527–536, Arlington, Virginia, USA, 2010. AUAI Press. ISBN 9780974903965.

Arvid Sjölander. Propensity scores and m-structures. *Statistics in medicine*, 28(9):1416–1420, 2009.

Julie Tibshirani, Susan Athey, Erik Sverdrup, and Stefan Wager. *grf: Generalized Random Forests*, 2026. URL https://CRAN.R-project.org/package=grf. R package version 2.6.1.

Anastasios A Tsiatis. *Semiparametric theory and missing data*. Springer, 2006.

Mark J. van der Laan and Daniel Rubin. Targeted maximum likelihood learning. *The International Journal of Biostatistics*, 2(1):1–40, 2006. doi: 10.2202/1557-4679.1043. URL https://www.ovid.com/journals/tijbio/abstract/10.2202/1557-4679.1043~targeted-maximum-likelihood-learning.

Stefan Wager and Susan Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 113(523):1228–1242, 2018.

Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6 (60):3021, 2021. doi: 10.21105/joss.03021. URL https://doi.org/10.21105/joss.03021.

Wes McKinney. Data Structures for Statistical Computing in Python. In Stéfan van der Walt and Jarrod Millman (eds.), *Proceedings of the 9th Python in Science Conference*, pp. 56 – 61, 2010. doi: 10.25080/Majora-92bf1922-00a.

Jeffrey M Wooldridge. Should instrumental variables be used as matching variables? *Research in Economics*, 70(2):232–237, 2016.

Omry Yadan. Hydra - a framework for elegantly configuring complex applications. Github, 2019. URL https://github.com/facebookresearch/hydra.

Omry Yadan, Jasha Sommer-Simpson, and Olivier Delalleau. Omegaconf, 2019. URL https://github.com/omry/omegaconf.

Liuyi Yao, Sheng Li, Yaliang Li, Mengdi Huai, Jing Gao, and Aidong Zhang. Representation learning for treatment effect estimation from observational data. *Advances in neural information processing systems*, 31, 2018.

A PROOF OF COROLLARY 3.2

Corollary 3.2 (Worst-case OVB bound)**.** *Under Ass. 2.1, 2.2 and 2.4, let $m^\ell = m^* \vee m^c$ be the valid long mask obtained by adding the omitted confounders m^c to m^*, such that $\psi(m^\ell) = \tau$. Let ϵ_ℓ be the largest constant satisfying $\epsilon_\ell \le e_\ell(X_{m^\ell}) \le 1 - \epsilon_\ell$ a.s. Then,*

$$\mathrm{Bias}^2(m^*) \le V_{\mathrm{res}}(m^*)\frac{(1-2\epsilon_\ell)^2}{\epsilon_\ell(1-\epsilon_\ell)}. \tag{4}$$

Proof. For a mask m, we define $\mu_{a,m}(x_m) = \mathbb{E}[Y \mid A = a, X_m = x_m]$ and the Riesz representer α_m as in Eq. (3):

$$\alpha_m := \frac{A}{e_m(X_m)} - \frac{1-A}{1-e_m(X_m)}. \tag{6}$$

Let

$$\mu_* := \mu_{A,m^*}(X_{m^*}), \qquad \mu_\ell := \mu_{A,m^\ell}(X_{m^\ell}),$$
$$\alpha_* := \alpha_{m^*}, \qquad \alpha_\ell := \alpha_{m^\ell}.$$

Recall that $m^\ell = m^* \vee m^c$ is a valid adjustment mask, so $\psi(m^\ell) = \tau$, and the omitted-variable bias is $\mathrm{Bias}(m^*) = \psi(m^*) - \psi(m^\ell)$, up to the sign. Using Theorem 2 in Chernozhukov et al. (2021) adapted to our notation, the squared omitted variable bias can be written as:

$$\mathrm{Bias}^2(m^*) = \left\{\mathbb{E}\left[(\mu_* - \mu_\ell)(\alpha_* - \alpha_\ell)\right]\right\}^2 = \left\{\mathbb{E}\left[(\mu_* - \mu_\ell)\alpha_* \cdot \frac{\alpha_* - \alpha_\ell}{\alpha_*}\right]\right\}^2.$$

Here the long and short functionals correspond to $\psi(m^\ell)$ and $\psi(m^*)$, respectively. By the Cauchy-Schwarz inequality,

$$\mathrm{Bias}^2(m^*) \le \mathbb{E}\left[(\mu_* - \mu_\ell)^2\alpha_*^2\right]\mathbb{E}\left[\left(\frac{\alpha_* - \alpha_\ell}{\alpha_*}\right)^2\right]. \tag{7}$$

Additionally, we can decompose the weighted residual variance:

$$\begin{aligned}\mathbb{E}\left[(Y-\mu_*)^2\alpha_*^2\right] &= \mathbb{E}\left[\left((Y-\mu_\ell) - (\mu_* - \mu_\ell)\right)^2\alpha_*^2\right] \\ &= \mathbb{E}\left[(Y-\mu_\ell)^2\alpha_*^2\right] + \mathbb{E}\left[(\mu_* - \mu_\ell)^2\alpha_*^2\right] - 2\mathbb{E}\left[(Y-\mu_\ell)(\mu_* - \mu_\ell)\alpha_*^2\right].\end{aligned}$$

The cross-term vanishes. Since

$$\mu_\ell = \mathbb{E}[Y \mid A, X_{m^\ell}],$$

we have

$$\mathbb{E}[Y - \mu_\ell \mid A, X_{m^\ell}] = 0.$$

Hence, by iterated expectation,

$$\begin{aligned}\mathbb{E}\left[(Y-\mu_\ell)(\mu_* - \mu_\ell)\alpha_*^2\right] &= \mathbb{E}\left[(\mu_* - \mu_\ell)\alpha_*^2\mathbb{E}[Y - \mu_\ell \mid A, X_{m^\ell}]\right] \\ &= 0.\end{aligned}$$

It follows that we can bound the squared difference of the conditional expectations:

$$\mathbb{E}\left[(\mu_* - \mu_\ell)^2\alpha_*^2\right] \le \mathbb{E}\left[(Y-\mu_*)^2\alpha_*^2\right]. \tag{8}$$

By the definition of the residual variance component in Eq. (3), we have $\mathbb{E}\left[(Y-\mu_*)^2\alpha_*^2\right] = V_{\mathrm{res}}(m^*)$. It remains to bound the relative Riesz shift. Writing $e_\ell = e_\ell(X_{m^\ell})$ and $e_* = e_*(X_{m^*})$, and conditioning on X_{m^ℓ}, we have $A \sim \mathrm{Bern}(e_\ell)$. Hence

$$\begin{aligned}\mathbb{E}\left[\left(\frac{\alpha_* - \alpha_\ell}{\alpha_*}\right)^2 \middle| X_{m^\ell}\right] &= e_\ell\left(\frac{e_\ell - e_*}{e_\ell}\right)^2 + (1-e_\ell)\left(\frac{e_* - e_\ell}{1-e_\ell}\right)^2 \\ &= \frac{(e_* - e_\ell)^2}{e_\ell(1-e_\ell)}.\end{aligned}$$

Since $\mathrm{supp}(m^*) \subseteq \mathrm{supp}(m^\ell)$, with $\mathrm{supp}(m) = \{j : m_j = 1\}$, we have $e_* = \mathbb{E}[e_\ell \mid X_{m^*}]$, so the long-mask positivity bound implies $e_* \in [\epsilon_\ell, 1 - \epsilon_\ell]$ a.s. Therefore,

$$|e_* - e_\ell| \le 1 - 2\epsilon_\ell$$

and

$$e_\ell(1 - e_\ell) \ge \epsilon_\ell(1 - \epsilon_\ell).$$

Hence,

$$\mathbb{E}\left[\left(\frac{\alpha_* - \alpha_\ell}{\alpha_*}\right)^2\right] \le \frac{(1 - 2\epsilon_\ell)^2}{\epsilon_\ell(1 - \epsilon_\ell)}. \tag{9}$$

Substituting Eq. (9), Eq. (8) and $\mathbb{E}\left[(Y - \mu_*)^2\alpha_*^2\right] = V_{\mathrm{res}}(m^*)$ into Eq. (7) gives Eq. (4). □

B Proof of Proposition 4.1 (plug-in convergence for the gate objective)

First, we state additional assumptions on top of the assumptions in Sec. 2 formally:

Assumption B.1 (Strong positivity and clipping margin)**.** As an implication of strong positivity for all x in the support of X, we assume strong positivity for all masks, i.e. $\epsilon \le e_m(X_m) \le 1 - \epsilon$. Let δ denote the numerical clipping level used in $\mathrm{clip}(\hat{e}_m, \delta, 1 - \delta)$. We assume $\delta < \epsilon$, so the true propensity lies inside the clipping interval.

Assumption B.2 ($L_2(P)$ consistency of nuisances on validation draws)**.** Let P denote the probability distribution of a validation draw (X, A, Y). As the training sample size $N \to \infty$, the plug-in nuisance estimators are mean-square consistent:

$$\|\hat{\mu}_{a,m} - \mu_{a,m}\|_{L_2(P)} \xrightarrow{P} 0 \quad \text{for } a \in \{0, 1\}, \qquad \|\hat{e}_m - e_m\|_{L_2(P)} \xrightarrow{P} 0.$$

Assumption B.3 (Sample splitting for the gate objective)**.** The validation sample used to evaluate the gate objective is independent of the training procedure used to fit $(\hat{\mu}_{0,m}, \hat{\mu}_{1,m}, \hat{e}_m)$, including the training sample and any algorithmic randomness.

We can now restate the proposition and prove it.

Proposition 4.1 (Plug-in convergence of $\mathcal{L}_{\mathrm{VERA}}$ at a fixed mask)**.** *Fix a binary mask $m \in \{0, 1\}^p$. Let $(X_i, A_i, Y_i)_{i=1}^n$ be an i.i.d. validation sample of size n, independent of the training sample of size N used to fit the nuisance plug-ins $(\hat{\mu}_{0,m}, \hat{\mu}_{1,m}, \hat{e}_m)$. Assume Ass. 2.2 and 2.3 and that the clipping level is chosen outside the positivity margin (Ass. B.1), the nuisance plug-ins are $L_2(P)$-consistent (Ass. B.2) and there is sample splitting for the gate objective (Ass. B.3). As the validation sample size $n \to \infty$ and the training sample size $N \to \infty$, the plug-in analogue $\widehat{V}_n$ converges to the residual variance component $V_{\mathrm{res}}(m)$ in probability, i.e., $\widehat{V}_n \xrightarrow{P} V_{\mathrm{res}}(m)$. Moreover, if $\varepsilon_{\mathrm{num}} > 0$, then by the continuous mapping theorem,*

$$\mathcal{L}_{\mathrm{VERA}} = \log\left(\widehat{V}_n + \varepsilon_{\mathrm{num}}\right) \xrightarrow{P} \log\left(V_{\mathrm{res}}(m) + \varepsilon_{\mathrm{num}}\right).$$

Proof. Define the oracle summand

$$Z_i^* := A_i \frac{(Y_i - \mu_{1,m}(X_{m,i}))^2}{e_m(X_{m,i})^2} + (1 - A_i)\frac{(Y_i - \mu_{0,m}(X_{m,i}))^2}{(1 - e_m(X_{m,i}))^2},$$

and note that $\mathbb{E}[Z_i^*] = V_{\mathrm{res}}(m)$ by conditioning on X_m,

$$\begin{aligned}\mathbb{E}\left[A\frac{(Y - \mu_{1,m})^2}{e_m^2}\middle| X_m\right] &= \frac{1}{e_m^2}\mathbb{E}[A(Y - \mu_{1,m})^2 \mid X_m] = \frac{1}{e_m^2}e_m\mathbb{E}[(Y - \mu_{1,m})^2 \mid A = 1, X_m] \\ &= \frac{\sigma_{1,m}^2(X_m)}{e_m(X_m)},\end{aligned}$$

and similarly for the $(1 - A)$ term.

First, we apply the weak law of large numbers to the average of the oracle summand. For this, we show that $\mathbb{E}\left[|Z^*|\right]$ is finite. By Ass. B.1, $e_m(X_m) \in [\epsilon, 1-\epsilon]$, so

$$|Z_i^*| \leq \epsilon^{-2}\left((Y_i - \mu_{1,m}(X_{m,i}))^2 + (Y_i - \mu_{0,m}(X_{m,i}))^2\right). \tag{10}$$

Hence,

$$\mathbb{E}\left[|Z^*|\right] \leq \epsilon^{-2} \sum_{a\in\{0,1\}} \mathbb{E}\left[(Y - \mu_{a,m}(X_m))^2\right]. \tag{11}$$

Using $(u-v)^2 \leq 2u^2 + 2v^2$, Ass. 2.3, and $\mathbb{E}[\mathbb{1}\{A=a\}\mu_{a,m}(X_m)^2] \leq \mathbb{E}[\mathbb{1}\{A=a\}Y^2] \leq \mathbb{E}[Y^2]$, we get $\mathbb{E}\left[|Z^*|\right] < \infty$. Therefore, by the weak law of large numbers,

$$\frac{1}{n}\sum_{i=1}^{n} Z_i^* - \mathbb{E}[Z^*] \xrightarrow{P} 0. \tag{12}$$

Next, we show that clipping of the propensity scores $\hat{e}_m$ is irrelevant in the limit. Define

$$g(t) := \min\{\max(t,\delta), 1-\delta\}, \qquad \hat{e}_{\text{clip}}(X_m) = g(\hat{e}_m(X_m)).$$

Since the clipping map g is 1-Lipschitz and $g(e_m(X_m)) = e_m(X_m)$ under $\delta < \epsilon$,

$$\|\hat{e}_{\text{clip}} - e_m\|_{L_2(P)} = \|g(\hat{e}_m) - g(e_m)\|_{L_2(P)} \leq \|\hat{e}_m - e_m\|_{L_2(P)} \xrightarrow{P} 0. \tag{13}$$

Using the convergence of the clipped propensity scores, we can now show that the difference between the empirical summand and the oracle summand $|\hat{w}_i\hat{r}_i^2 - Z_i^*|$ is asymptotically negligible. Let

$$w_i^* := \frac{A_i}{e_m(X_{m,i})^2} + \frac{1-A_i}{(1-e_m(X_{m,i}))^2}, \qquad r_i^* := Y_i - \mu_{A_i,m}(X_{m,i}).$$

We decompose the error $\left|\hat{w}_i\hat{r}_i^2 - Z_i^*\right|$ into a weight component and a squared-residual component:

$$\begin{aligned}\left|\hat{w}_i\hat{r}_i^2 - Z_i^*\right| &= \left|\hat{w}_i\hat{r}_i^2 - w_i^*(r_i^*)^2\right| = \left|(\hat{w}_i - w_i^*)\hat{r}_i^2 + w_i^*(\hat{r}_i^2 - (r_i^*)^2)\right| \\ &\leq |\hat{w}_i - w_i^*|\hat{r}_i^2 + |w_i^*|\left|\hat{r}_i^2 - (r_i^*)^2\right|. \end{aligned} \tag{14}$$

We now show that both terms in Eq. (14) are asymptotically negligible. Let $\mathcal{T}$ denote the training procedure, including the training sample and any algorithmic randomness used to fit $(\hat{\mu}_{0,m}, \hat{\mu}_{1,m}, \hat{e}_m)$. Conditional on $\mathcal{T}$, the fitted nuisances are fixed functions, while the validation observations remain i.i.d. We start with the weight difference component. Using

$$\hat{r}_i = r_i^* - \left(\hat{\mu}_{A_i,m}(X_{m,i}) - \mu_{A_i,m}(X_{m,i})\right),$$

we have

$$\hat{r}_i^2 \leq 2(r_i^*)^2 + 2\left(\hat{\mu}_{A_i,m}(X_{m,i}) - \mu_{A_i,m}(X_{m,i})\right)^2.$$

Moreover, the map $t \mapsto A/t^2 + (1-A)/(1-t)^2$ is Lipschitz on $[\delta, 1-\delta]$, so

$$|\hat{w}_i - w_i^*| \leq L(\delta)\left|\hat{e}_{\text{clip}}(X_{m,i}) - e_m(X_{m,i})\right|.$$

Hence,

$$\begin{aligned}\mathbb{E}\left[|\hat{w}_i - w_i^*|\hat{r}_i^2 \mid \mathcal{T}\right] &\leq 2L(\delta)\mathbb{E}\left[|\hat{e}_{\text{clip}}(X_m) - e_m(X_m)|\,(r^*)^2 \mid \mathcal{T}\right] \\ &\quad + 2L(\delta)\mathbb{E}\left[|\hat{e}_{\text{clip}}(X_m) - e_m(X_m)|\left(\hat{\mu}_{A,m}(X_m) - \mu_{A,m}(X_m)\right)^2 \mid \mathcal{T}\right].\end{aligned}$$

The second term converges to zero in probability because $|\hat{e}_{\text{clip}} - e_m| \leq 1$ and

$$\mathbb{E}\left[\left(\hat{\mu}_{A,m}(X_m) - \mu_{A,m}(X_m)\right)^2 \mid \mathcal{T}\right] = \|\hat{\mu}_{A,m} - \mu_{A,m}\|_{L_2(P)}^2 \xrightarrow{P} 0.$$

For the first term, we can use a truncation argument. Since $\mathbb{E}[(r^*)^2] < \infty$, for any $M > 0$,

$$\begin{aligned}&\mathbb{E}\left[|\hat{e}_{\text{clip}}(X_m) - e_m(X_m)|\,(r^*)^2 \mid \mathcal{T}\right] \\ &= \mathbb{E}\left[\mathbb{1}\{(r^*)^2 \leq M\}\,|\hat{e}_{\text{clip}}(X_m) - e_m(X_m)|\,(r^*)^2 \mid \mathcal{T}\right] \\ &\quad + \mathbb{E}\left[\mathbb{1}\{(r^*)^2 > M\}\,|\hat{e}_{\text{clip}}(X_m) - e_m(X_m)|\,(r^*)^2 \mid \mathcal{T}\right] \\ &\leq M\mathbb{E}\left[|\hat{e}_{\text{clip}}(X_m) - e_m(X_m)| \mid \mathcal{T}\right] + \mathbb{E}\left[(r^*)^2\mathbb{1}\{(r^*)^2 > M\}\right] \\ &\leq M\|\hat{e}_{\text{clip}} - e_m\|_{L_2(P)} + \mathbb{E}\left[(r^*)^2\mathbb{1}\{(r^*)^2 > M\}\right].\end{aligned}$$

First choose M large so that the tail term is arbitrarily small, and then use $\|\hat{e}_{\text{clip}} - e_m\|_{L_2(P)} \xrightarrow{P} 0$. Therefore,

$$\mathbb{E}\left[|\hat{w}_i - w_i^*|\hat{r}_i^2 \mid \mathcal{T}\right] \xrightarrow{P} 0.$$

Now we consider the squared-residual difference component. Using $(a^2 - b^2) = (a-b)(a+b)$, we have

$$\begin{aligned}\left|\hat{r}_i^2 - (r_i^*)^2\right| &= |\hat{r}_i - r_i^*|\,|\hat{r}_i + r_i^*| \\ &= |\hat{\mu}_{A_i,m}(X_{m,i}) - \mu_{A_i,m}(X_{m,i})|\,|\hat{\mu}_{A_i,m}(X_{m,i}) + \mu_{A_i,m}(X_{m,i}) - 2Y_i|\,.\end{aligned}$$

Therefore, by Cauchy-Schwarz,

$$\mathbb{E}\left[\left|\hat{r}_i^2 - (r_i^*)^2\right| \mid \mathcal{T}\right] \le \|\hat{\mu}_{A,m} - \mu_{A,m}\|_{L_2(P)}\|\hat{\mu}_{A,m} + \mu_{A,m} - 2Y\|_{L_2(P)} \xrightarrow{P} 0.$$

The first factor converges to zero by Ass. B.2, while the second is bounded in probability since

$$\|\hat{\mu}_{A,m} + \mu_{A,m} - 2Y\|_{L_2(P)} \le \|\hat{\mu}_{A,m} - \mu_{A,m}\|_{L_2(P)} + 2\|\mu_{A,m} - Y\|_{L_2(P)}$$

and $\mathbb{E}[Y^2] < \infty$ by Ass. 2.3. Since w_i^* is bounded by a constant depending only on ϵ, it follows that the full squared-residual component satisfies

$$\mathbb{E}\left[|w_i^*|\left|\hat{r}_i^2 - (r_i^*)^2\right| \mid \mathcal{T}\right] \xrightarrow{P} 0.$$

Together with the weight-component bound and Eq. (14), this gives

$$\mathbb{E}\left[\left|\hat{w}_i\hat{r}_i^2 - Z_i^*\right|\middle|\mathcal{T}\right] \xrightarrow{P} 0.$$

Then,

$$\begin{aligned}\mathbb{E}\left[\left|\frac{1}{n}\sum_{i=1}^n\left(\hat{w}_i\hat{r}_i^2 - Z_i^*\right)\right|\middle|\mathcal{T}\right] &\le \frac{1}{n}\sum_{i=1}^n \mathbb{E}\left[\left|\hat{w}_i\hat{r}_i^2 - Z_i^*\right|\middle|\mathcal{T}\right] \\ &= \mathbb{E}\left[\left|\hat{w}_1\hat{r}_1^2 - Z_1^*\right|\middle|\mathcal{T}\right] \xrightarrow{P} 0.\end{aligned}$$

By Markov's inequality,

$$\frac{1}{n}\sum_{i=1}^n\left(\hat{w}_i\hat{r}_i^2 - Z_i^*\right) \xrightarrow{P} 0.$$

Putting everything together, we use

$$\widehat{V}_n - \mathbb{E}[Z^*] = \frac{1}{n}\sum_{i=1}^n(\hat{w}_i\hat{r}_i^2 - Z_i^*) + \left(\frac{1}{n}\sum_{i=1}^n Z_i^* - \mathbb{E}[Z^*]\right).$$

The first term converges to 0 in probability by the plug-in error bound above, and the second term converges to 0 in probability by Eq. (12). Therefore,

$$\widehat{V}_n \xrightarrow{P} \mathbb{E}[Z^*] = V_{\text{res}}(m).$$

Finally, since $\varepsilon_{\text{num}} > 0$ is fixed, the map $x \mapsto \log(x + \varepsilon_{\text{num}})$ is continuous on $[0, \infty)$. By the continuous mapping theorem,

$$\mathcal{L}_{\text{VERA}} = \log\left(\widehat{V}_n + \varepsilon_{\text{num}}\right) \xrightarrow{P} \log\left(V_{\text{res}}(m) + \varepsilon_{\text{num}}\right).$$

□

C FINITE-SAMPLE DECOMPOSITION OF THE GATE OBJECTIVE

In Prop. 4.1, we show that under consistency of the nuisance estimators and strict sample splitting, the empirical gate objective converges to the residual EIF variance component $V_{\mathrm{res}}(m)$ for a fixed mask. We next characterize its finite-sample conditional expectation on an independent validation sample when the nuisance models are estimated on the training sample. This shows how estimation error in the outcome and propensity models affects the criterion.

To recall, we have the following nuisance functions, given a mask $m \in \{0,1\}^p$ and masked covariates X_m:

$$\begin{aligned} e_m(x_m) &= \Pr(A{=}1 \mid X_m{=}x_m), \\ \mu_{a,m}(x_m) &= \mathbb{E}[Y \mid A{=}a,\, X_m{=}x_m], \\ \sigma^2_{a,m}(x_m) &= \mathrm{Var}(Y \mid A{=}a,\, X_m{=}x_m), \end{aligned}$$

for $a \in \{0,1\}$, and the targeted EIF variance term

$$V_{\mathrm{res}}(m) = \mathbb{E}\left[\frac{\sigma^2_{1,m}(X_m)}{e_m(X_m)} + \frac{\sigma^2_{0,m}(X_m)}{1-e_m(X_m)}\right].$$

Let $\hat{\mu}_{a,m}$ and $\hat{e}_m$ denote the outcome models and the propensity fitted on the training half D_{train}, and let the plug-in of Eq. (5) be evaluated on a validation set of n_v points (denoted n in Prop. 4.1) $(X_i^{\mathrm{val}}, A_i^{\mathrm{val}}, Y_i^{\mathrm{val}})$, where the superscript marks membership in the validation split:

$$\widehat{V}_n(m) = \frac{1}{n_v}\sum_{i\in\mathrm{val}}\left[\frac{A_i^{\mathrm{val}}\big(Y_i^{\mathrm{val}} - \hat{\mu}_{1,m}(X_{m,i}^{\mathrm{val}})\big)^2}{\hat{e}_m(X_{m,i}^{\mathrm{val}})^2} + \frac{(1-A_i^{\mathrm{val}})\big(Y_i^{\mathrm{val}} - \hat{\mu}_{0,m}(X_{m,i}^{\mathrm{val}})\big)^2}{\big(1-\hat{e}_m(X_{m,i}^{\mathrm{val}})\big)^2}\right].$$

Throughout, hatted quantities are fitted on D_{train} and are therefore fixed once we condition on D_{train}, and $\hat{e}_m$ denotes the clipped propensity used in the gate objective.

Assumption C.1 (Sample splitting)**.** The validation points $(X_i^{\mathrm{val}}, A_i^{\mathrm{val}}, Y_i^{\mathrm{val}})$ are i.i.d. draws from P, independent of D_{train}.

Assumption C.2 (Gate inputs)**.** $\hat{\mu}_{a,m}$ and $\hat{e}_m$ are functions of x_m and D_{train} only.

Assumption C.3 (Moments and clipping)**.** We assume $\mathbb{E}[Y^2] < \infty$. Also, for $a \in \{0,1\}$ the fitted outcome models satisfy $\mathbb{E}\big[\hat{\mu}_{a,m}(X_m)^2 \,\big|\, D_{\mathrm{train}}\big] < \infty$ almost surely, and the fitted propensity satisfies $\hat{e}_m(x_m) \in (\delta, 1-\delta)$ for some $\delta > 0$ (guaranteed by the clipping in Eq. (5)).

Assumption C.1 would be satisfied for the fixed-mask gate objective if the validation split was not used for early stopping (or otherwise involved in fitting the nuisance models). In our implementation, the validation split is used for early stopping, so the strict independence assumption does not hold exactly. Assumption C.2 holds by construction, since the gated models see only the masked inputs.

Proposition C.4 (The held-out plug-in estimates V_{res} plus a weighted outcome model error)**.** *Let $\Delta_a(x_m) = \hat{\mu}_{a,m}(x_m) - \mu_{a,m}(x_m)$ be the estimation error in the outcome model, and $\rho_1(x_m) = e_m(x_m)/\hat{e}_m(x_m)$, $\rho_0(x_m) = (1-e_m(x_m))/(1-\hat{e}_m(x_m))$, which express the propensity score estimation error as multiplicative weights.*

Under Assumptions C.1–C.3,

$$\begin{aligned} \mathbb{E}\big[\widehat{V}_n(m)\big|D_{\mathrm{train}}\big] = \mathbb{E}\Bigg[&\rho_1(X_m^{\mathrm{val}})^2\frac{\sigma^2_{1,m}(X_m^{\mathrm{val}}) + \Delta_1(X_m^{\mathrm{val}})^2}{e_m(X_m^{\mathrm{val}})} \\ &+ \rho_0(X_m^{\mathrm{val}})^2\frac{\sigma^2_{0,m}(X_m^{\mathrm{val}}) + \Delta_0(X_m^{\mathrm{val}})^2}{1-e_m(X_m^{\mathrm{val}})} \;\Bigg|\; D_{\mathrm{train}}\Bigg], \end{aligned}$$

where the expectation on the right is over a generic validation draw X_m^{val}. When the propensity score is correctly specified ($\hat{e}_m = e_m$, so $\rho_a = 1$),

$$\mathbb{E}\big[\widehat{V}_n(m)\big|D_{\mathrm{train}}\big] = V_{\mathrm{res}}(m) + \mathbb{E}\left[\frac{\Delta_1(X_m^{\mathrm{val}})^2}{e_m(X_m^{\mathrm{val}})} + \frac{\Delta_0(X_m^{\mathrm{val}})^2}{1-e_m(X_m^{\mathrm{val}})}\Big|D_{\mathrm{train}}\right].$$

Remark C.5 (Sensitivity to the two nuisance errors). The decomposition separates into a V_{res} part and an error part:

$$\begin{aligned}\mathbb{E}\big[\widehat{V}_n(m)\big|D_{\mathrm{train}}\big] &= \mathbb{E}\left[\rho_1^2\frac{\sigma_{1,m}^2}{e_m}+\rho_0^2\frac{\sigma_{0,m}^2}{1-e_m}\Big|D_{\mathrm{train}}\right]+\mathbb{E}\left[\rho_1^2\frac{\Delta_1^2}{e_m}+\rho_0^2\frac{\Delta_0^2}{1-e_m}\Big|D_{\mathrm{train}}\right]\\ &= \big(\rho^2\text{-reweighted } V_{\mathrm{res}}(m)\big)+\big(\text{reweighted outcome model error}\big),\end{aligned}$$

where all functions are evaluated at X_m^{val}. This shows that the two nuisance errors enter asymmetrically. The outcome model error enters additively and only through its square Δ_a^2, so a small outcome model error changes the target at second order. The propensity error enters multiplicatively and at first order: writing $\hat{e}_m = e_m+\xi$ for a small perturbation ξ, we have $\rho_1^2 = (1+\xi/e_m)^{-2} \approx 1-2\,\xi/e_m$ (and similarly for ρ_0^2). The plug-in criterion is not an orthogonalized (debiased) estimate of V_{res}, and therefore does not inherit the second-order robustness to nuisance errors that doubly robust ATE estimators have. However, the criterion is only used to compare masks, so an error that multiplies every mask's target by the same constant would leave the selection unchanged. What matters for selection is how the effect of the propensity error differs across masks. A debiased estimate of $V_{\mathrm{res}}(m)$ could be constructed in principle. We regard this as future work.

Proof. By Assumption C.1, the validation points are i.i.d. and independent of D_{train}, so all n_v summands of $\widehat{V}_n(m)$ have the same conditional expectation given D_{train}. It is therefore sufficient to compute this expectation for a single generic validation point $(X^{\mathrm{val}}, A^{\mathrm{val}}, Y^{\mathrm{val}})$, whose masked covariates we write as X_m^{val}.

We first consider only the treatment arm of $\widehat{V}_n$, and condition on the training data D_{train}. Then, we additionally condition on X_m^{val}. Using Assumption C.2, $\hat{e}_m(X_m^{\mathrm{val}})$ is fixed given $(X_m^{\mathrm{val}}, D_{\mathrm{train}})$, so it can be pulled out of the inner expectation:

$$\begin{aligned}&\mathbb{E}\left[\frac{A^{\mathrm{val}}\,(Y^{\mathrm{val}}-\hat{\mu}_{1,m}(X_m^{\mathrm{val}}))^2}{\hat{e}_m(X_m^{\mathrm{val}})^2}\Big|D_{\mathrm{train}}\right]\\ &\quad=\mathbb{E}\left[\frac{1}{\hat{e}_m(X_m^{\mathrm{val}})^2}\mathbb{E}\big[A^{\mathrm{val}}(Y^{\mathrm{val}}-\hat{\mu}_{1,m}(X_m^{\mathrm{val}}))^2\big|X_m^{\mathrm{val}},D_{\mathrm{train}}\big]\Big|D_{\mathrm{train}}\right].\end{aligned}$$

Since the validation point is independent of D_{train} (Assumption C.1), conditioning on D_{train} does not change the distribution of $(A^{\mathrm{val}}, Y^{\mathrm{val}})$ given X_m^{val}. It follows that $\Pr(A^{\mathrm{val}}{=}1 \mid X_m^{\mathrm{val}}, D_{\mathrm{train}}) = e_m(X_m^{\mathrm{val}})$. Since $\hat{\mu}_{1,m}(X_m^{\mathrm{val}})$ is a constant given $(X_m^{\mathrm{val}}, D_{\mathrm{train}})$ (Assumption C.2),

$$\mathbb{E}\big[A^{\mathrm{val}}(Y^{\mathrm{val}}-\hat{\mu}_{1,m})^2\big|X_m^{\mathrm{val}},D_{\mathrm{train}}\big]=e_m(X_m^{\mathrm{val}})\mathbb{E}\big[(Y^{\mathrm{val}}-\hat{\mu}_{1,m})^2\big|A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big].$$

Then, we can decompose the expectation on the right with $Y^{\mathrm{val}}-\hat{\mu}_{1,m} = (Y^{\mathrm{val}}-\mu_{1,m})+(\mu_{1,m}-\hat{\mu}_{1,m}) = (Y^{\mathrm{val}}-\mu_{1,m})-\Delta_1$. Note that $\Delta_1 = \Delta_1(X_m^{\mathrm{val}})$ is a constant given $(X_m^{\mathrm{val}}, D_{\mathrm{train}})$, again by Assumption C.2. Hence,

$$\begin{aligned}&\mathbb{E}\big[(Y^{\mathrm{val}}-\hat{\mu}_{1,m})^2 \mid A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big]\\ &\quad=\mathbb{E}\big[(Y^{\mathrm{val}}-\mu_{1,m})^2 \mid A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big]\\ &\qquad-2\mathbb{E}\big[(\hat{\mu}_{1,m}-\mu_{1,m})(Y^{\mathrm{val}}-\mu_{1,m}) \mid A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big]\\ &\qquad+\mathbb{E}\big[(\hat{\mu}_{1,m}-\mu_{1,m})^2 \mid A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big]\\ &\quad=\mathbb{E}\big[(Y^{\mathrm{val}}-\mu_{1,m})^2 \mid A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big]\\ &\qquad-2\Delta_1\mathbb{E}\big[Y^{\mathrm{val}}-\mu_{1,m} \mid A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big]+\Delta_1^2\\ &\quad=\sigma_{1,m}^2(X_m^{\mathrm{val}})+\Delta_1^2.\end{aligned}$$

The cross term vanishes by Assumption C.1,

$$\mathbb{E}\big[Y^{\mathrm{val}}-\mu_{1,m}\big|A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big]=\mathbb{E}\big[Y^{\mathrm{val}}-\mu_{1,m}\big|A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}}\big]=0,$$

by the definition of $\mu_{1,m}$ as the conditional mean given $(A{=}1, X_m)$. The first term equals $\sigma_{1,m}^2(X_m^{\mathrm{val}})$ by the same argument. It follows that

$$\mathbb{E}\big[(Y^{\mathrm{val}}-\hat{\mu}_{1,m})^2\big|A^{\mathrm{val}}{=}1,X_m^{\mathrm{val}},D_{\mathrm{train}}\big]=\sigma_{1,m}^2(X_m^{\mathrm{val}})+\Delta_1(X_m^{\mathrm{val}})^2,$$

and therefore

$$\mathbb{E}\left[\frac{A^{\text{val}}(Y^{\text{val}}-\hat{\mu}_{1,m})^2}{\hat{e}_m^2}\Big|D_{\text{train}}\right]=\mathbb{E}\left[\frac{e_m}{\hat{e}_m^2}\big(\sigma_{1,m}^2+\Delta_1^2\big)\Big|D_{\text{train}}\right]=\mathbb{E}\left[\rho_1^2\frac{\sigma_{1,m}^2+\Delta_1^2}{e_m}\Big|D_{\text{train}}\right], \tag{15}$$

using $e_m/\hat{e}_m^2=(e_m/\hat{e}_m)^2/e_m=\rho_1^2/e_m$, where all functions are evaluated at X_m^{val}. The same operations can be applied to the term corresponding to the control arm, i.e.

$$\begin{aligned}\mathbb{E}\left[\frac{(1-A^{\text{val}})(Y^{\text{val}}-\hat{\mu}_{0,m})^2}{(1-\hat{e}_m)^2}\Big|D_{\text{train}}\right]&=\mathbb{E}\left[\frac{1-e_m}{(1-\hat{e}_m)^2}\big(\sigma_{0,m}^2+\Delta_0^2\big)\Big|D_{\text{train}}\right]\\&=\mathbb{E}\left[\rho_0^2\frac{\sigma_{0,m}^2+\Delta_0^2}{1-e_m}\Big|D_{\text{train}}\right].\end{aligned} \tag{16}$$

Summing equation 15 and equation 16 gives the conditional expectation of a single validation summand. Since all n_v summands share this expectation (Assumption C.1), averaging over the validation set concludes the proof. □

D Dataset details

D.1 Synthetic data

This section provides the technical specifications for the synthetic Data Generating Process (DGP) used to evaluate the variance-bias trade-off in high-dimensional adjustment.

Covariate generation. The covariates $X\in\mathbb{R}^p$ are sampled from a multivariate Gaussian distribution $\mathcal{N}(\mathbf{0},\Sigma)$. We evaluate two distinct structures for the covariance matrix Σ:

1. Independent: $\Sigma=I_p$, representing an orthogonal covariate space.
2. Correlated: A Toeplitz structure defined by $\Sigma_{jk}=\rho^{|j-k|}$ with $\rho=0.5$, representing decaying local correlations.

To introduce non-linearity, we apply a polynomial transformation $B_{ij}=X_{ij}+X_{ij}^2$ to each covariate. All subsequent treatment and outcome models are linear in this basis B, which makes them non-linear in the original observed feature space X.

Treatment and outcome. The treatment assignment A_i and outcome Y_i are generated as follows:

$$a_i=\sum_{j=1}^{p}\beta_j^{(A)}B_{ij} \tag{17}$$

$$\tilde{a}_i=a_i-\bar{a}+\omega \tag{18}$$

$$A_i\sim\text{Bern}(\text{expit}(\gamma\cdot\tilde{a}_i)) \tag{19}$$

$$Y_i=\sum_{j=1}^{p}\beta_j^{(Y)}B_{ij}+\tau A_i+\varepsilon_i^{(Y)} \tag{20}$$

where $\bar{a}$ is the empirical mean of a across the sample used for centering and $\text{expit}(x)=1/(1+e^{-x})$ denotes the logistic function. We set the population treatment effect to $\tau=3$.

Hyperparameters and scaling. To ensure results are comparable across varying dimensionalities p, we apply the following scaling and parameter settings:

- Signal-to-Noise Ratio (SNR): The outcome's non-linear component $Y_{\text{nl}}=\sum_j\beta_j^{(Y)}B_{ij}$ is centered and scaled to $\text{Var}(Y_{\text{nl}})=5$. The noise term $\varepsilon^{(Y)}$ is sampled from $\mathcal{N}(0,1)$, yielding an outcome SNR of 5:1.

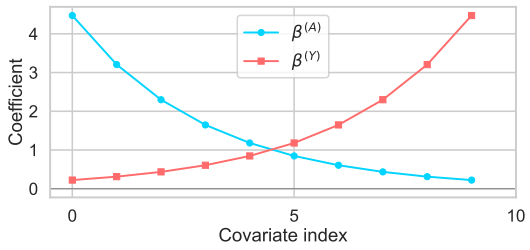


Figure 6: Coefficient values $\beta^{(A)}$ and $\beta^{(Y)}$.

- Overlap and balance: We set $\gamma = 0.1$ to keep the propensity scores within the high-slope region of the sigmoid function, ensuring sufficient overlap. The shift parameter $\omega = 4$ addresses the skew of the quadratic basis, resulting in approximately 57% treated and 43% control units.
- Confounding strength: The coefficients $\beta^{(A)}$ and $\beta^{(Y)}$ are constrained by $\beta_j^{(A)}\beta_j^{(Y)} = 1$. This keeps the product of treatment and outcome influence constant, meaning the potential bias from omitting any single covariate is roughly equal. The specific ratios $q_j = \beta_j^{(A)}/\beta_j^{(Y)}$ are chosen from a log-spaced grid in $[0.05, 20]$ (see Fig. 6).

D.2 Real-world data (Jobs)

On Jobs, we report the MAE of the ATT and the policy risk. The metrics are defined as follows. Let $S_i \in \{0, 1\}$ indicate the NSW randomized subsample. Following Shalit et al. (2017), $\text{ATT} = \frac{1}{\sum_j A_j}\sum_i A_i Y_i - \frac{1}{\sum_j (1-A_j)S_j}\sum_i (1-A_i)S_i Y_i$ and $\epsilon_{\text{ATT}} = \big|\text{ATT} - \frac{1}{\sum_j A_j}\sum_i A_i\big(\hat{\mu}_1(X_i) - \hat{\mu}_0(X_i)\big)\big|$. For $\hat{\tau}(x) = \hat{\mu}_1(x) - \hat{\mu}_0(x)$ and $\pi(x) = \mathbb{1}\{\hat{\tau}(x) > 0\}$, $R_{\text{pol}}(\pi) = 1 - \big(\widehat{\mathbb{E}}[Y_i \mid \pi(X_i) = 1, A_i = 1]\hat{p}(\pi{=}1) + \widehat{\mathbb{E}}[Y_i \mid \pi(X_i) = 0, A_i = 0]\hat{p}(\pi{=}0)\big)$, with all expectations and $\hat{p}(\cdot)$ evaluated on $\{i : S_i = 1\}$ only.

E Implementation details

For methods that use data splitting, we use a stratified 70/30 train-validation split. For binary outcomes (Jobs dataset), stratification is performed jointly over treatment and outcome. The R-Learner additionally uses two-fold treatment-stratified cross-fitting within the training data for its nuisance and effect models. All neural base learners are implemented in PyTorch (Ansel et al., 2024) and use hidden layers of size 200 with ELU activations: S-Learner, T-Learner, and R-Learner use three-layer MLPs, while TARNet is our PyTorch implementation of the architecture in Shalit et al. (2017), with a three-layer shared representation of size 200 followed by separate potential-outcome heads. Unlike the standard TARNet setup, we do not L2-normalize the representation, since this can interfere with gate updates.

Neural networks are trained with Adam (Kingma & Ba, 2017), learning rate 0.001, early stopping with patience 20, and exponential learning-rate decay with gamma 0.5 every 100 epochs, for up to 2000 epochs. For VERA, we draw one stratified 50/50 train-validation split per replication and reuse it across all cycles. We fit the core network and auxiliary propensity network on the train subset, using the validation subset for early stopping. Gate parameters are then updated with Adam using gate learning rate 0.1 for at most 200 epochs per cycle, with batches drawn from the validation subset. The cycle loop terminates early once the gate probabilities stabilize, defined as a maximum change below 0.01 for 5 consecutive cycles. After gate selection, the final core fit uses the 70/30 split.

DML uses XGBoost nuisance learners with 300 estimators (Chen & Guestrin, 2016; Chernozhukov et al., 2018). Causal forest is implemented with `grf`, using default forest settings (Tibshirani et al., 2026). BART is implemented with `dbarts`, using the package defaults (Dorie et al., 2026). We also use the package MCMC defaults for tree depth and posterior draws. For IHDP with $N = 50$ and for the synthetic data with $p = 100$, we additionally use the σ-prior settings `sigest=`

1, `sigdf`$= 3$, and `sigquant`$= 0.9$. For continuous-outcome Lasso, we first residualize both covariates and outcome with respect to treatment, standardize the residualized covariates to zero mean and unit variance so that the ℓ_1 penalty is scale-free, and select the Lasso penalty by cross-validation. Coefficients are mapped back to the original covariate scale for prediction. All penalized baselines (Lasso/S, Lasso/T, OAL, PDS) standardize covariates in this way, with scaling statistics computed on training data only (per treatment arm for Lasso/T). For PDS with the BCH plug-in penalty, the per-covariate penalty weights are computed on the standardized inputs. Neural base learners and VERA receive the covariates without additional scaling.

The treatment effect is estimated separately and is therefore not shrunk by the Lasso penalty. For binary outcomes, we instead use L2-penalized logistic regression on treatment and covariates with standard `scikit-learn` settings (Pedregosa et al., 2011). For outcome-adaptive Lasso (OAL) and post-double-selection Lasso (PDS), we follow the implementations described in Shortreed & Ertefaie (2017) and Belloni et al. (2014b), respectively. For PDS, we apply both the Belloni-Chernozhukov-Hansen plug-in penalty and a cross-validated Lasso penalty, followed by an unpenalized second-stage OLS. OAL follows Shortreed & Ertefaie (2017): covariates are standardized, an outcome model $Y \sim T + X$ gives coefficients β, and a weighted-ℓ_1 logistic propensity model is learned with adaptive weights $w_j = |\beta_j|^{-\gamma}$. We search $\lambda_n = n^a$ over $a \in -10, -5, -2, -1, -0.75, -0.5, -0.25, 0.25, 0.49$ with $\gamma = 2(\kappa - a + 1)$ and convergence factor $\kappa = 2$, where we select λ by the weighted absolute mean difference of IPW-weighted covariate means. For estimating the final ATE under the selected propensity model, propensities are clipped at 10^{-3}.

Software and compute. Our neural estimators are implemented in PyTorch (Ansel et al., 2024) (BSD-3-Clause license). For data handling, evaluation and plotting, we use NumPy (Harris et al., 2020) (BSD-3-Clause), pandas (pandas development team, 2020; Wes McKinney, 2010) (BSD-3-Clause), matplotlib (Hunter, 2007) (PSF-based license), seaborn (Waskom, 2021) (BSD-3-Clause), scikit-learn (Pedregosa et al., 2011) (BSD-3-Clause), Hydra (Yadan, 2019) (MIT license) and OmegaConf (Yadan et al., 2019) (BSD-3-Clause). For baselines, we use XGBoost (Chen & Guestrin, 2016) (Apache-2.0), causallib (Shimoni et al., 2019) (Apache-2.0), and the R packages `grf` (Tibshirani et al., 2026) (GPL-3) and `dbarts` (Dorie et al., 2026) (GPL-2/GPL-3), interfaced through `rpy2` (Gautier, 2023) (GPL-2-or-later). Code is available at `https://anonymous.4open.science/r/vera-iclr-7B76`. All reported experiments were run either locally or on AMD Rome CPU nodes. Cluster runs used one AMD Rome CPU node with up to 64 CPU cores and up to 128 GiB of memory. The longest requested wall-time limit was 100 hours.

F Additional results

In this section, we present additional results for the semi-synthetic benchmarks IHDP and ACIC 2018, as well as the learned variable inclusion probabilities on the Jobs dataset. In Table 3, we show the results of all VERA models and baselines on the IHDP dataset in a small-sample regime (within-sample size $N = 50$) and the full sample (within-sample size $N = 672$). VERA substantially improves the estimation of ATE and CATE compared to all base learners, except for TARNet's ATE estimation. However, in the small-sample regime, neural methods are outperformed by simpler methods such as T-structured Lasso and OLS, or BART.

At $N = 50$, one replication contained only one treated observation. DML and both PDS (CV) variants failed to fit on this replication, while causal forest returned a non-finite within-sample ATE error. We therefore report all methods on the 999 replications that were completed by all methods.

Figure 7 reports the mean absolute improvement of VERA over each base model across IHDP sample sizes for within-sample evaluation as well (whereas the main text focuses on out-of-sample errors). The qualitative patterns closely match the main-text plots, i.e. VERA similarly improves performance in both regimes. Figure 8 summarizes relative out-of-sample error of VERA compared to each base model on ACIC 2018, including full-sample runs with $N = 1000$. For readability, extreme ratios are removed from Fig. 8. The untrimmed boxplots including those points appear in Fig. 9. Most of these extreme values lie above 10^0, meaning that they correspond to replications where VERA's error exceeds the base model's. Such points are usually ratio artifacts: they are most

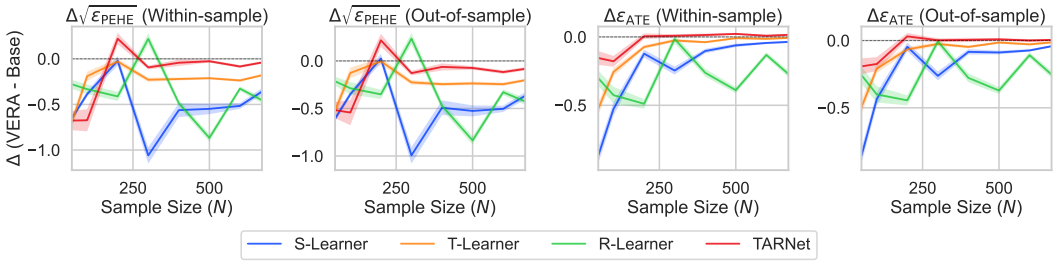


Figure 7: Performance improvements of VERA over base neural learners (within- and out-of-sample results). Averaged per-experiment differences in error on the IHDP dataset across sample sizes. Values below zero indicate VERA outperforms the base learner.

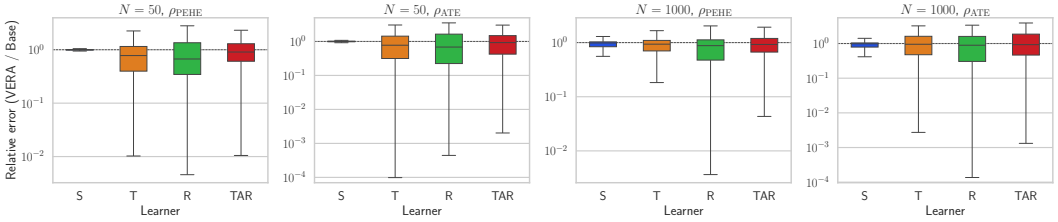


Figure 8: Ratio of VERA error to baseline error for PEHE and ATE on ACIC 2018 with $N = 50$ and $N = 1000$, denoted as ρ_{PEHE} and ρ_{ATE}. Values below 10^0 indicate a lower error for VERA. Outliers are omitted for clarity but included in quartile calculations.

common when the base error is very small (so the ratio can spike even though absolute error may remain small) or when both methods are far from the truth.

Table 3: IHDP results grouped by sample size $N \in \{50, 672\}$. For each N, within-sample then out-of-sample $\sqrt{\epsilon_{\text{PEHE}}}$ and ϵ_{ATE} (mean $\pm$ SE over replications). At $N = 50$ all methods are evaluated on the 999 replications completed by every method; at $N = 672$ all 1000 are used. Lowest per column is bolded, second lowest is underlined.

	$N = 50$				$N = 672$			
	Within-sample		Out-of-sample		Within-sample		Out-of-sample	
Method	$\sqrt{\epsilon_{\text{PEHE}}}$	ϵ_{ATE}	$\sqrt{\epsilon_{\text{PEHE}}}$	ϵ_{ATE}	$\sqrt{\epsilon_{\text{PEHE}}}$	ϵ_{ATE}	$\sqrt{\epsilon_{\text{PEHE}}}$	ϵ_{ATE}
OLS/S	5.94 ± 0.26	1.61 ± 0.08	6.11 ± 0.27	1.86 ± 0.11	5.80 ± 0.26	0.73 ± 0.04	5.78 ± 0.26	0.94 ± 0.05
OLS/T	2.58 ± 0.08	0.51 ± 0.02	4.28 ± 0.16	0.80 ± 0.03	2.26 ± 0.10	0.13 ± 0.00	2.33 ± 0.11	0.29 ± 0.02
Lasso/S	5.80 ± 0.25	1.19 ± 0.06	5.96 ± 0.27	1.51 ± 0.08	5.79 ± 0.26	0.72 ± 0.04	5.78 ± 0.26	0.93 ± 0.05
Lasso/T	$\underline{2.13} \pm 0.08$	$\mathbf{0.42} \pm 0.01$	$\mathbf{3.25} \pm 0.15$	$\mathbf{0.69} \pm 0.03$	2.12 ± 0.10	0.11 ± 0.00	2.18 ± 0.11	0.27 ± 0.02
OAL/S	5.66 ± 0.25	0.50 ± 0.02	5.83 ± 0.26	1.05 ± 0.05	5.74 ± 0.25	0.12 ± 0.00	5.71 ± 0.26	0.60 ± 0.04
OAL/T	2.76 ± 0.11	0.45 ± 0.01	3.78 ± 0.16	0.74 ± 0.03	2.21 ± 0.10	$\underline{0.11} \pm 0.00$	2.26 ± 0.11	0.28 ± 0.02
PDS (BCH)/S	5.70 ± 0.25	0.77 ± 0.04	5.86 ± 0.26	1.15 ± 0.06	5.80 ± 0.26	0.78 ± 0.04	5.79 ± 0.26	0.98 ± 0.05
PDS (BCH)/T	2.85 ± 0.12	0.45 ± 0.01	3.84 ± 0.17	0.77 ± 0.04	2.52 ± 0.12	0.12 ± 0.00	2.57 ± 0.13	0.31 ± 0.02
PDS (CV)/S	5.91 ± 0.25	1.51 ± 0.07	6.07 ± 0.27	1.76 ± 0.09	5.80 ± 0.26	0.74 ± 0.04	5.78 ± 0.26	0.95 ± 0.05
PDS (CV)/T	2.35 ± 0.09	0.44 ± 0.01	3.48 ± 0.15	$\underline{0.71} \pm 0.03$	2.13 ± 0.10	0.11 ± 0.00	2.19 ± 0.11	0.27 ± 0.02
IPW	5.65 ± 0.25	0.47 ± 0.01	5.82 ± 0.26	1.03 ± 0.05	5.74 ± 0.25	0.12 ± 0.00	5.71 ± 0.26	0.61 ± 0.04
DML	5.88 ± 0.25	1.37 ± 0.06	6.03 ± 0.27	1.61 ± 0.08	5.80 ± 0.26	0.71 ± 0.04	5.78 ± 0.26	0.94 ± 0.05
BART	$\mathbf{1.86} \pm 0.07$	$\underline{0.42} \pm 0.01$	$\underline{3.48} \pm 0.16$	0.71 ± 0.03	0.91 ± 0.03	$\mathbf{0.10} \pm 0.00$	1.37 ± 0.06	$\mathbf{0.18} \pm 0.01$
CausalFor.	5.66 ± 0.25	0.55 ± 0.02	5.83 ± 0.26	1.05 ± 0.05	4.16 ± 0.19	0.23 ± 0.01	4.18 ± 0.20	0.60 ± 0.03
R-Learner	5.75 ± 0.22	2.03 ± 0.06	6.02 ± 0.24	2.18 ± 0.07	3.07 ± 0.13	0.81 ± 0.03	3.10 ± 0.14	0.86 ± 0.03
VERA R.	5.48 ± 0.22	1.73 ± 0.06	5.80 ± 0.24	1.92 ± 0.08	2.61 ± 0.12	0.54 ± 0.02	2.67 ± 0.13	0.61 ± 0.03
S-Learner	6.43 ± 0.24	2.50 ± 0.06	6.55 ± 0.25	2.58 ± 0.06	1.52 ± 0.08	0.18 ± 0.01	1.59 ± 0.08	0.24 ± 0.01
VERA S.	5.78 ± 0.24	1.63 ± 0.06	5.94 ± 0.26	1.72 ± 0.06	1.16 ± 0.06	0.15 ± 0.01	1.22 ± 0.06	0.20 ± 0.01
T-Learner	4.50 ± 0.16	1.86 ± 0.07	5.12 ± 0.19	1.98 ± 0.08	1.22 ± 0.03	0.22 ± 0.01	1.35 ± 0.04	0.26 ± 0.01
VERA T.	3.80 ± 0.14	1.33 ± 0.06	4.59 ± 0.18	1.49 ± 0.07	1.04 ± 0.02	0.21 ± 0.01	1.15 ± 0.03	0.24 ± 0.01
TARNet	4.89 ± 0.22	1.42 ± 0.07	5.38 ± 0.24	1.61 ± 0.08	$\underline{0.79} \pm 0.02$	0.17 ± 0.00	$\underline{0.89} \pm 0.03$	$\underline{0.20} \pm 0.01$
VERA TAR.	4.22 ± 0.21	1.27 ± 0.07	4.86 ± 0.22	1.42 ± 0.08	$\mathbf{0.75} \pm 0.02$	0.18 ± 0.01	$\mathbf{0.81} \pm 0.02$	0.20 ± 0.01

Table 4 shows the full results on the Jobs dataset including all baselines. Figure 10 shows VERA's learned inclusion probabilities on the Jobs dataset. Different VERA models disagree on the inclusion of some of the variables. Even within one model class, there is uncertainty about inclusion of some variables, resulting in different inclusion decisions on the same dataset across different runs. Models

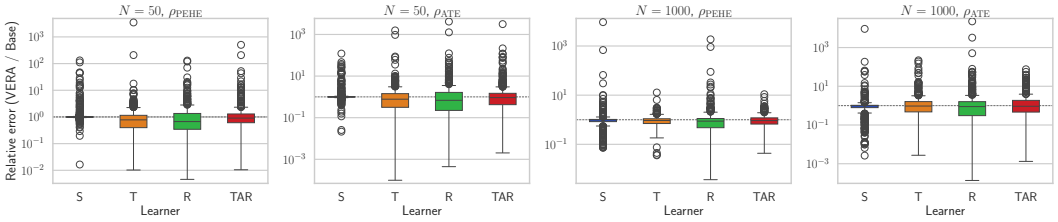


Figure 9: Ratio of VERA error to baseline error for PEHE and ATE on ACIC 2018 with $N = 50$ and $N = 1000$, denoted as ρ_{PEHE} and ρ_{ATE}, including outliers. Values below 10^0 indicate a lower error for VERA.

Table 4: Within-sample and out-of-sample performance on Jobs. Lowest value in each column is bolded, second lowest is underlined.

	within-sample		out-of-sample	
Method	ϵ_{ATT}	R_{pol}	ϵ_{ATT}	R_{pol}
Logistic/S	$.04 \pm .00$	$.24 \pm .00$	$.10 \pm .01$	$\underline{.24} \pm .01$
Logistic/T	$.04 \pm .00$	$.26 \pm .00$	$.11 \pm .01$	$.28 \pm .01$
IPW	$.06 \pm .00$	$.25 \pm .00$	$.11 \pm .01$	$.25 \pm .01$
DML	$.01 \pm .00$	$.24 \pm .00$	$\underline{.10} \pm .01$	$\underline{.24} \pm .01$
BART	$\mathbf{.01} \pm .00$	$.22 \pm .00$	$.10 \pm .01$	$.27 \pm .01$
Caus.For.	$\underline{.01} \pm .00$	$\mathbf{.19} \pm .00$	$\mathbf{.09} \pm .01$	$.25 \pm .01$
R-Learner	$.07 \pm .00$	$.23 \pm .00$	$.11 \pm .01$	$.25 \pm .01$
VERA R.	$.09 \pm .00$	$.26 \pm .00$	$.13 \pm .01$	$.29 \pm .01$
S-Learner	$.07 \pm .00$	$.26 \pm .00$	$.12 \pm .01$	$.26 \pm .01$
VERA S.	$.06 \pm .01$	$.27 \pm .00$	$.11 \pm .01$	$.28 \pm .01$
T-Learner	$.07 \pm .01$	$.22 \pm .00$	$.12 \pm .01$	$.25 \pm .01$
VERA T.	$.06 \pm .01$	$.26 \pm .00$	$.11 \pm .01$	$.28 \pm .01$
TARNet	$.04 \pm .00$	$\underline{.20} \pm .00$	$.11 \pm .01$	$\mathbf{.24} \pm .01$
VERA TAR.	$.05 \pm .00$	$.24 \pm .00$	$.11 \pm .01$	$.29 \pm .01$

tend to agree on the inclusion of the covariates *Earnings in 1975*, *Earnings in 1974* and *Hispanic*. Arguably, past employment status, which is reflected in past earnings, is most predictive of the outcome, making it a variable that improves the EIF variance-term. It seems like the covariates *Years of education*, *Age* and *Black* tend to be excluded by VERA. Potentially, the covariate *No high school degree* is sufficiently predictive of the employment status, while explaining less of the treatment variation compared to *Years of education*, a covariate that may result in poor overlap, as the control sample includes highly educated persons (Dehejia & Wahba, 1999) that may have had a weaker indication for treatment.

G Additional experiments and analyses

This section provides additional experiments and analyses of VERA. Unless otherwise stated, all synthetic experiments in this section use 1,000 random replications.

G.1 The N/p ratio as a rule of thumb

To characterize the regime in which VERA is most useful, we vary the ratio N/p in the independent synthetic DGP. The results in Table 5 suggest the largest improvements at $N/p \leq 1$, with improvements up to roughly $N/p = 2$ in this setting. This boundary is not universal: for the T-Learner the advantage disappears at $N/p = 4$ in this sweep, while TARNet still improves on its base learner.

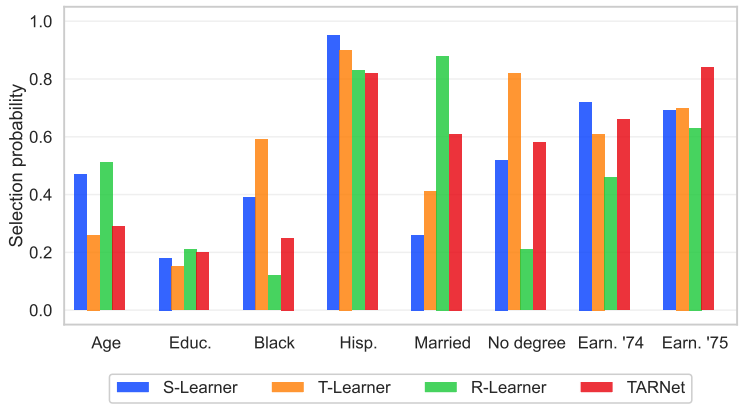


Figure 10: VERA's learned selection probabilities of each variable on the Jobs dataset.

Additionally, we perform experiments with varying N and p while holding N/p fixed. As can be seen in Table 6, the relative error ratio stays similar across scales. This supports the use of N/p as an empirical rule of thumb rather than N or p separately.

Table 5: N/p sweep for $p = 100$. Entries are VERA's mean absolute ATE error divided by that of the corresponding base learner. Values below one favor VERA.

N	N/p	VERA T./base	VERA TAR./base
25	0.25	0.60	0.66
50	0.5	0.56	0.61
100	1	0.51	0.68
200	2	0.62	0.70
400	4	1.46	0.66

Table 6: Fixed-N/p scale experiment. Entries are VERA's mean absolute ATE error divided by that of the corresponding base learner.

N/p	N	p	VERA T./base	VERA TAR./base
0.5	25	50	0.59	0.64
0.5	50	100	0.56	0.61
0.5	100	200	0.55	0.59
2	50	25	0.65	0.73
2	200	100	0.62	0.70
2	400	200	0.71	0.42

G.2 Sensitivity to overlap

We also isolate overlap by varying only the propensity scale in the $p = 100, N = 50$ synthetic DGP (Table 7). VERA's relative advantage remains comparatively stable as the fraction of extreme true propensities increases. A possible explanation is that, as the propensity scale increases, increasingly extreme propensities arise in the saturated region of the assignment function. In this region, the bias cost of omitting treatment-predictive covariates can saturate, while the variance cost of keeping them can continue to increase through more extreme inverse-propensity weights. Consequently, the bias-variance trade-off exploited by VERA can remain favorable even as overlap worsens. This mechanism may arise under commonly used saturating treatment-assignment models, although the exact behavior depends on the underlying DGP.

Table 7: Overlap experiments on the synthetic DGP with $p = 100, N = 50$. The first row is the fraction of true propensities outside $[0.1, 0.9]$. The reported ratio is VERA's ATE RMSE divided by that of the corresponding base learner.

Extreme-propensity fraction	0.16	0.48	0.73	0.86	0.93	0.96	0.98
T-Learner RMSE	1.51	1.40	1.31	1.29	1.31	1.30	1.29
VERA T./base	0.57	0.57	0.58	0.57	0.57	0.56	0.57
TARNet RMSE	1.26	1.19	1.12	1.09	1.09	1.08	1.09
VERA TAR./base	0.60	0.62	0.64	0.66	0.66	0.66	0.66

On ACIC 2018, where assignment, nonlinearity, effect heterogeneity, and noise vary jointly across datasets, the median VERA/base error ratios range from approximately 0.53–0.92 in the best-overlap tercile to 0.89–1.01 in the worst-overlap tercile. Therefore, unlike in the synthetic overlap experiment, the relative benefit of VERA is smaller in the ACIC datasets with the worst overlap.

G.3 Covariate types and confounder omission

Selection of different covariate types. We first construct a sparse linear-Gaussian DGP that includes the variable types in Fig. 2. It contains $p = 100$ independent standard-normal covariates: 10 IVs, 10 confounders spanning a near-IV-to-near-precision range similar to the main synthetic setting (App. D.1), 10 precision variables, and 70 noise variables. Before rescaling the outcome signal, the IV treatment coefficients equal 0.3 and the precision-variable outcome coefficients equal 1. Within the confounder block, we set $q_j = \beta_j^{(A)}/\beta_j^{(Y)}$ on a log-spaced grid from 0.1 to 10, while holding $\beta_j^{(A)}\beta_j^{(Y)} = 0.09$. We use $N = 50$ observations per replication. The combined outcome signal from the confounders and precision variables is rescaled to variance 5, and the variance of the outcome noise is 2, giving an SNR of $5{:}2$. Treatment is generated from a logistic model using the score $X_i^\top \beta^{(A)}$, with no additional scaling. We present the results in Table 8. Precision variables are dropped least often (approximately 35% on average), followed by confounders (43%), and then IVs (51%) and noise variables (53%). This ordering is in line with VERA's objective, which prefers variables that explain outcome variation without strongly explaining treatment assignment, although none of the selection methods perfectly separates the variable types at $N = 50$.

Table 8: Comparison of VERA with base models and baselines on the synthetic DGP with four variable types. The IV, Conf., Prec. and Noise columns report the fraction of covariates of each type that the method drops.

Method	IV	Conf.	Prec.	Noise	RMSE	Bias
VERA T.	0.51	0.43	0.34	0.53	0.562	-0.16
VERA TAR.	0.52	0.43	0.36	0.53	0.681	0.01
T-Learner	–	–	–	–	0.660	-0.40
TARNet	–	–	–	–	0.657	-0.26
OAL	0.93	0.84	0.58	0.93	0.634	0.05
PDS (BCH)	1.00	1.00	1.00	1.00	0.829	0.38
PDS (CV)	0.46	0.36	0.28	0.51	5.857	-0.59
Lasso	0.77	0.67	0.34	0.77	0.775	0.31
DML	–	–	–	–	0.724	0.14

Dominant confounder. We investigate when a single dominant confounder is dropped by VERA. A strong confounder can potentially be dropped when this lowers V_{res}. However, each term in the gate loss is a product of a propensity weight and a squared outcome residual, so when removing a variable makes the propensities less extreme but also increases the outcome residuals, the overall loss can still increase. A strong confounder is therefore more likely to be retained when it explains substantial residual outcome variation.

We modify the synthetic DGP with independent covariates (App. D.1) by making the coefficient product $\beta_{Aj}\beta_{Yj}$ of one confounder $j^\star$ account for half of the sum of these products. Specifically, let

$$q_j = \frac{\beta_{Aj}}{\beta_{Yj}}, \qquad c_j = \beta_{Aj}\beta_{Yj}.$$

As in the standard setting, $\sum_{j=1}^{p} c_j = p$, but with

$$c_{j^\star} = \frac{p}{2}, \qquad c_j = \frac{p}{2(p-1)} \quad (j \neq j^\star).$$

We set $q_{j^\star} = 20$ in the treatment-heavy setting and $q_{j^\star} = 0.05$ in the outcome-heavy setting. The remaining ratios stay on the same geometric grid from 0.05 to 20 as in the standard synthetic DGP. In the corresponding setting with a linear treatment equation, these coefficient products are proportional to the individual covariates' contributions to confounding, so $j^\star$ would account for half

Table 9: Frequency with which VERA drops the dominant confounder $j^\star$ over 1,000 replications.

Setting	Learner	$p = 10$	$p = 100$
Treatment-heavy	T-Learner	47.2%	30.7%
Treatment-heavy	TARNet	43.9%	34.7%
Outcome-heavy	T-Learner	28.6%	22.8%
Outcome-heavy	TARNet	28.0%	23.8%

of the total confounding contribution. In our DGP, treatment is assigned through a logistic function, so this correspondence is only approximate.

Table 9 reports the probability that VERA drops $j^\star$. The treatment-heavy dominant confounder is dropped in 30.7%–47.2% of replications, compared with 22.8%–28.6% for the outcome-heavy confounder. This difference is consistent across dimensions and learners: VERA keeps the dominant confounder in the majority of replications in all settings, and is more likely to keep it when it explains substantial outcome variation compared to treatment variation.

Adversarial confounder. We next consider a setting in which the population V_{res} criterion prefers dropping a strong confounder even though retaining it yields lower finite-sample MSE in an oracle comparison. To align the setting with the Lasso-based variable-selection baselines, we use a linear outcome model with independent Gaussian covariates. The DGP contains a single strong confounder $j^\star$ and $p - 1$ precision variables. Specifically,

$$X_{ij^\star} \sim \mathcal{N}(0, 3), \qquad X_{ij} \sim \mathcal{N}(0, 1), \quad j \neq j^\star,$$

and treatment depends only on the confounder,

$$A_i \mid X_i \sim \text{Bern}\left(\text{expit}(0.35 X_{ij^\star})\right).$$

The outcome is generated from

$$Y_i = A_i + 0.3 X_{ij^\star} + b_p \sum_{j \neq j^\star} X_{ij} + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, 4),$$

so that the treatment effect is $\tau = 1$. We use $N = 50$ and set

$$b_p = \sqrt{\frac{9}{p-1}},$$

such that the overall variance contributed by the precision variables is 9 at both $p = 10$ and $p = 100$. In the high-dimensional setting, the confounder contributes approximately 0.27 to the variance of the outcome signal, compared with approximately 0.09 for each precision variable. Thus, the confounder explains about three times as much outcome variation as any precision variable, which encourages its selection by the Lasso-type baselines.

At $p = 10$, we verify that this setting gives a disagreement between V_{res} and finite-sample MSE. We use the ground-truth contribution of the precision variables to the outcome,

$$S_i = b_p \sum_{j \neq j^\star} X_{ij}.$$

The oracle OLS estimator adjusts for $(X_{ij^\star}, S_i)$ in the "keep" setting and only for S_i in the "drop" setting. The population criterion prefers dropping the confounder,

$$V_{\text{res}}(\text{drop}) = 16.996 < V_{\text{res}}(\text{keep}) = 17.614,$$

whereas at $N = 50$ the oracle OLS estimator has lower MSE when the confounder is kept (0.373 versus 0.440). Dropping $j^\star$ yields an asymptotic bias of approximately 0.29. Hence, in this oracle comparison, minimizing V_{res} results in an adjustment with a larger MSE. We additionally evaluate this in the high-dimensional setting with $p = 100$. The scaling of b_p keeps the variance of the combined precision signal constant across dimensions. Since the combined precision score S_i has

Table 10: Adversarial-confounder results. "$j^\star$-drop" is the fraction of replications in which the strong confounder is dropped.

	$p = 10$			$p = 100$		
Method	$j^\star$-drop	RMSE	Bias	$j^\star$-drop	RMSE	Bias
OAL	0.47	0.78	0.11	0.68	0.90	-0.14
PDS (BCH)	0.95	1.08	0.24	1.00	1.10	0.24
PDS (CV)	0.02	0.66	-0.01	0.29	23.99	-1.14
Lasso	0.05	0.67	0.00	0.66	1.12	0.15
T-Learner	–	0.85	-0.09	–	0.83	-0.44
TARNet	–	0.85	-0.11	–	0.86	-0.33
VERA T.	0.52	0.94	0.03	0.33	0.81	-0.24
VERA TAR.	0.55	1.00	0.06	0.38	1.19	-0.11

the same distribution across dimensions and is independent of $j^\star$, and since the propensity depends only on $j^\star$, the population V_{res} ordering and the asymptotic bias from omitting $j^\star$ are unchanged.

We present the finite-sample results in Table 10.

At $p = 100$, VERA omits $j^\star$ less often than OAL, PDS with the BCH penalty, and Lasso. PDS with the cross-validated penalty retains the confounder more often, but is highly unstable in this setting and has substantially higher RMSE. VERA with the T-Learner remains competitive with full adjustment in RMSE.

G.4 Algorithmic behavior and optimization

We examine several aspects of VERA's optimization procedure.

Selection stability. We analyze the stability of the selected covariates across $1{,}000$ random replications in the independent synthetic setting with $N = 50$. For VERA, OAL, and PDS with the cross-validated penalty, we report the mean size of the selected set and the mean pairwise Jaccard overlap, where a Jaccard value of zero indicates disjoint sets and a value of one identical sets. We omit PDS with the BCH penalty from this comparison because it selects no covariates in this setting and is therefore trivially stable.

Table 11: Selection stability across $1{,}000$ synthetic replications. Jaccard denotes the mean pairwise Jaccard overlap of the selected sets.

Method	p	Mean size	Jaccard
OAL	10	4.3	0.380
PDS (CV)	10	7.6	0.592
VERA TAR.	10	4.6	0.307
VERA T.	10	4.5	0.304
OAL	100	11.1	0.052
PDS (CV)	100	46.3	0.254
VERA TAR.	100	48.2	0.318
VERA T.	100	49.3	0.328

As shown in Table 11, the selected sets vary substantially across replications for the compared methods. The Jaccard values should be interpreted together with the selected-set sizes, since larger sets tend to have greater overlap. For example, two sets that each omit only one of the p covariates must share at least $p-2$ variables, corresponding to a Jaccard overlap of at least 0.98 when $p = 100$. Hence, the comparatively large Jaccard values of methods that select larger sets do not necessarily imply more stable variable selection. In the $p = 10$ setting, OAL seems more stable than VERA at comparable selected-set sizes. In the high-dimensional setting with $p = 100$, VERA seems more stable than PDS (CV) at comparable selected-set sizes.

Soft versus hard gates. The gate objective is optimized using continuous hard-sigmoid values, whereas the final adjustment set is obtained by thresholding the gates. To examine the resulting mismatch, we compare the continuous and corresponding hard-mask objectives in 10 TARNet runs on the independent synthetic setting with $N = 50$ and $p = 100$.

Most gates saturate to exactly zero or one within the first cycle. For cycles after the first one, the mean difference between the soft-gate objective and its value at the corresponding hard mask is approximately 0.002–0.070. During the first cycle, while the gates are still changing, the mismatch can be substantially larger, with gaps up to approximately 0.8. Nevertheless, in all 10 runs the hard-mask objective improved while the hard gates were still changing, and the soft and hard objective values coincided once the gates saturated. Thus, the soft objective can differ substantially from the corresponding hard-mask objective during the initial gate updates, but in these runs this does not appear to misdirect the optimization.

Re-entry. Although a fully saturated hard-sigmoid gate receives no gradient, this does not imply that VERA is a monotone pruning procedure: variables below the binary threshold can re-enter while their gates remain unsaturated. In the $p = 100$ synthetic setting, the selected mask grows between cycles in 65–71% of replications across the T-Learner and TARNet variants. Hence, variables that have been excluded at an intermediate cycle frequently re-enter the selected set at a later cycle.