Two-Stage Machine Learning for Nonparametric Instrumental Variable Regression

David Bruns-Smith*

Stanford University

October 2, 2025

Abstract

The growing access to large administrative datasets with rich covariates presents an opportunity to revisit classic two-stage least squares (2SLS) applications with machine learning (ML). We develop Two-Stage Machine Learning, a simple and efficient estimator for nonparametric instrumental variables (NPIV) regression. Our method uses ML models to flexibly estimate nonparametric treatment effects while avoiding the computational complexity and statistical instability of existing machine learning NPIV approaches. Our procedure has two steps: first, we predict the outcomes given instruments and covariates (the reduced form) and extract a basis from this predictor; second, we predict the outcomes using the treatment and covariates, but where the predictions are projected onto the learned basis of instruments. We prove that under a testable condition, our estimation error depends entirely on the reduced-form prediction task, where ML methods excel. We also develop a bias correction procedure that provides valid confidence intervals for scalar summaries like average derivatives. In an empirical application to California supermarket data featuring bunching at 99-ending price points, we find our machine learning approach is crucial for modeling discontinuities in demand at the dollar boundary: we reduce NPIV estimation error nearly seven-fold compared to previous estimators and estimate a price elasticity that is 2.5-6 times larger than prior estimates.

*I thank Kirill Borusyak, Kevin Chen, Giovanni Compiani, Avi Feller, Arthur Gretton, Guido Imbens, Evan Munro, Emi Nakamura, Jesse Rothstein, Jann Spiess, Jón Steinsson, Vasilis Syrgkanis, participants at the American Causal Inference Conference, and seminars at UC Berkeley and Stanford for helpful comments and discussions.
Researcher(s)' own analyses calculated (or derived) based in part on data from Nielsen Consumer LLC and marketing databases provided through the NielsenIQ Datasets at the Kilts Center for Marketing Data Center at The University of Chicago Booth School of Business. The conclusions drawn from the NielsenIQ data are those of the researcher(s) and do not reflect the views of NielsenIQ. NielsenIQ is not responsible for, had no role in, and was not involved in analyzing and preparing the results reported herein. All NielsenIQ data and results are redacted in this draft pending approval for public release. Contact the author at causal@stanford.edu for more details.

1 Introduction

Instrumental variable (IV) regression with continuous treatments and instruments underlies applications in empirical economics spanning returns to schooling (Card, 2001), the price elasticity of demand (Hausman, 1996), effects of monetary policy (Stock and Watson, 2018), and the marginal propensity to consume (Blundell et al., 2008). These applications have traditionally employed linear specifications solved with two-stage least squares (2SLS). The increasing availability of high-quality administrative datasets with large sample sizes and rich covariate information now makes it feasible to estimate machine learning models that can flexibly learn complex nonlinearities, discontinuities, and interaction effects. In settings with even moderate dimensionality and sample size, machine learning methods like gradient boosted trees or neural networks achieve substantially higher predictive accuracy than traditional nonparametric estimators such as sieve and kernel regression. These parallel developments in data availability and flexible modeling tools present an opportunity to revisit classic 2SLS applications in economics.

In this paper, we use machine learning models to estimate the nonparametric effects (also known as the structural function), $f_0(D, X)$, of an endogenous continuous treatment D on an outcome Y given covariates X, in a setting with instruments Z. For example, in a demand estimation setting where D is price and X contains product and market characteristics, then $f_0(D, X)$ would be the demand function. While heterogeneity in f_0 is interesting in its own right, flexibly estimating nonlinearities and interactions in $f_0(D, X)$ is also crucial for accurately estimating summaries like the average price elasticity. We adopt the nonparametric instrumental variables (NPIV) framework of Newey and Powell (2003), where the structural function $f_0(D, X)$ is the solution to the conditional moment equality $\mathbb{E}[Y|Z, X] = \mathbb{E}[f_0(D, X)|Z, X]$.[1] Our goal is to solve this moment equality using machine learning models to represent both $f_0(D, X)$ and $\mathbb{E}[f_0(D, X)|Z, X]$. This allows researchers to leverage the excellent out-of-performance prediction capabilities of modern machine learning methods to capture nonlinearities and heterogeneity in causal effects.

However, NPIV is both computationally and statistically difficult to solve with arbitrary machine learning models. The main challenge is nonlinearity. Under a linear model for f_0, the moment equality can be solved by two-stage least squares. In the first stage we estimate $\mathbb{E}[D|Z, X]$ — i.e. we predict the treatment given the instruments and covariates. Then, in the second stage we predict Y using the first-stage fitted values $\mathbb{E}[D|Z, X]$ and the covariates. When $f_0(D, X)$ is nonlinear, the traditional 2SLS approach fails because $\mathbb{E}[f_0(D, X)|Z, X] \neq f_0(\mathbb{E}[D|Z, X], X)$ — the

[1] Equivalently, we can write the GMM-type condition, $\mathbb{E}[Y - f_0(D, X)|Z, X] = 0$.

"forbidden regression" problem (Hausman, 1983). Instead, for every candidate solution $f(D, X)$, we must solve an additional prediction task to estimate $\mathbb{E}[f(D, X)|Z, X]$ — the best predictor of $f(D, X)$ given Z and X. Previous work bypasses this challenge by assuming that $f_0(D, X)$ is linear in a transformation of D and X, such as a sieve or kernel basis (Newey and Powell, 2003; Ai and Chen, 2003; Singh et al., 2019), but these methods have limited ability to model complicated real-world data, as we show later in our empirical application. In an important extension, Chen and Ludvigson (2009); Chen et al. (2023) introduce a computationally-tractable procedure that models f_0 with an arbitrary machine learning algorithm by making the strong restriction that $\mathbb{E}[f(D, X)|Z, X]$ is linear in a fixed sieve of the instruments uniformly across all $f(D, X)$ — the linearity assumption is moved from f_0 onto the instruments.

A recent and growing literature uses machine learning methods like trees or neural networks to model both $f_0(D, X)$ and $\mathbb{E}[f_0(D, X)|Z, X]$, but incorporated into statistically difficult and computationally intensive procedures. For example, DeepIV (Hartford et al., 2017; Li et al., 2024) replaces the traditional first stage with conditional density estimation, a statistically-intractable problem in even moderately-high dimensions. Other estimators solve the conditional moment equality directly using adversarial training (Bennett et al., 2019; Dikkala et al., 2020; Muandet et al., 2020; Liao et al., 2020), or by iterating between first and second stages (Xu et al., 2020; Bakhitov and Singh, 2022). In addition to high computational costs, the instability of these adversarial/iterative procedures can result in large estimation errors, even relative to simpler linear methods.

We introduce a simple and efficient two-stage procedure for NPIV that supports arbitrary machine learning models in both stages — we call our procedure *Two-Stage Machine Learning*. As in Chen et al. (2023), we can efficiently solve NPIV using machine learning to model f_0, provided we represent the instruments linearly in some basis like a sieve. Call this basis $\phi(Z, X)$. We show that the NPIV estimation error with such an approach is ultimately limited by how well $\phi(Z, X)$ linearly predicts the outcomes Y. This suggests a natural solution: learn $\phi(Z, X)$ by using machine learning to predict Y given Z and X, and then extract a basis from the fitted predictor. This prediction task estimates $\mathbb{E}[Y|Z, X]$ — called the "reduced-form" in linear IV. Since $\mathbb{E}[Y|Z, X] = \mathbb{E}[f_0(D, X)|Z, X]$, constructing $\phi(Z, X)$ from the reduced form guarantees that $\phi(Z, X)$ are strong instruments for $f_0(D, X)$.

Accordingly, Two-Stage Machine Learning works as follows: in our first stage, we predict Y given Z and X using machine learning, and construct a basis $\phi(Z, X)$ from the predictor. For example, we fit the reduced form with gradient-boosted trees, and then take $\phi(Z, X)$ to be the output of

each individual tree in the ensemble. In our second stage, we estimate the structural function with machine learning as in Chen et al. (2023) using the $\phi(Z, X)$ learned in the first stage. To our knowledge, this is the first NPIV estimator with an easy-to-run two stage structure that incorporates off-the-shelf ML predictors for both f_0 and $\mathbb{E}[f_0(D, X)|Z, X]$. Beyond helping to choose the basis $\phi(Z, X)$, the reduced form also serves as a convenient specification test for existing methods like Newey and Powell (2003); Singh et al. (2019); Chen et al. (2023): if sieve or kernel methods are not the best mean-squared error predictor for the reduced form, then they cannot be well-specified for $\mathbb{E}[f_0(D, X)|Z, X]$ in finite samples.

Additionally, we provide valid inference and asymptotic normal confidence intervals for scalar summaries of the structural function like the average derivative — in applied work, this may correspond to the average price elasticity or marginal propensity to consume. Because machine learning models typically introduce bias to reduce variance, naively using our estimated structural function to compute a point estimate will not yield an asymptotically normal estimator. Therefore, we develop a *bias correction* procedure based on double/debiased machine learning (Chernozhukov et al., 2023) that yields an unbiased point estimate for estimands like the average derivative with valid confidence intervals. The bias correction procedure requires solving a second conditional moment equation, facing the same statistical and computational challenges as NPIV. We show that an analogous Two-Stage Machine Learning procedure applied to a different loss function also solves the debiasing problem.

We prove finite-sample-valid error bounds and convergence guarantees for Two-Stage Machine Learning under a general loss function, subsuming both NPIV and the debiasing step as special cases. Our main theoretical contribution is to show that under a testable condition, the estimation error is entirely driven by the difficulty of the reduced form prediction task. We show that this condition holds in all datasets we consider when we construct $\phi(Z, X)$ using gradient-boosted trees. Importantly, we avoid requiring that $\phi(Z, X)$ can uniformly approximately $\mathbb{E}[f(D, X)|Z, X]$ for all f. Existing methods like the Sieve Minimum Distance estimator from (Chen et al., 2023) and Kernel IV (Singh et al., 2019) can be written as a special case of our procedure with (fixed) sieve and kernel bases for $\phi(Z, X)$ respectively, and so they inherit our analysis. This partly explains why our Two-Stage Machine Learning method outperforms these previous estimators in our empirical application — gradient-boosted trees achieve an out-of-sample R^2 for the reduced form prediction task that is 0.33 better than sieve regression, and 0.2 better than kernel ridge regression.

Because our convergence guarantee applies to both estimating the structural function and the de-

biasing step, we can derive conditions under which debiased Two-stage Machine Learning satisfies the convergence rates for asymptotic normality from Chernozhukov et al. (2023). In particular, our debiasing procedure enjoys "double robustness to ill-posedness", meaning that as long as we achieve fast rates for the reduced-form problem — for example, using standard analyses from statistical learning theory — then we can obtain a valid confidence interval if one or even (in some cases) both inverse problems are severely ill-posed. We provide examples of concrete rates when our algorithm is instantiated using tree ensembles.

We evaluate two stage machine learning using synthetic and semi-synthetic data. First, we assess how well our estimate approximates the true structural function as measured in mean squared error — this validates our L_2-convergence guarantees and demonstrates how well we can capture rich heterogeneity in the structural function. Our evaluation uses two semi-synthetic benchmarks constructed by adding correlated noise to large datasets on taxi fares and house prices. Compared to existing NPIV estimators, we improve R^2 for the true structural function on the two benchmarks by at least 0.1 and 0.15 respectively. Second, we assess the coverage of our debiased confidence intervals on a synthetic average derivative estimation task. We improve coverage for the 95% confidence interval from 70% without debiasing to 94.4% with debiasing, demonstrating that bias correction can be quite important for valid inference in practice.

Finally, we apply our debiased Two-Stage Machine Learning procedure to demand estimation using the California supermarket data from Compiani (2022). This dataset features extensive bunching at 9-ending price points. For example, [redacted] of our [redacted] observations on organic strawberries have a price ending in 0.99, and [redacted] have a price of exactly [redacted]. The tendency of prices to bunch at 9-endings has been widely observed (Anderson and Simester, 2003; Snir and Levy, 2021), contributing to uniform pricing (DellaVigna and Gentzkow, 2019) and asymmetric price rigidity (Levy et al., 2020). Our Two-Stage Machine Learning approach using tree ensembles excels at flexibly modeling the discontinuities in demand at the dollar boundary, resulting in a nearly sevenfold reduction in NPIV estimation error compared to the best previous estimator. Our debiased estimate of the average own-price elasticity is [redacted], between 2.5 and 6 times larger than previous estimates reported using this same dataset (Compiani, 2022; Chen et al., 2023). We find that our estimate is driven by large responses at the dollar boundary; for example, our estimated average price elasticity among the observations with price exactly [redacted] is around [redacted]. Our results have immediate implications for price rigidity: a supermarket using 99-ending prices cannot pass small increases in cost onto the consumer without realizing potentially large decreases in demand.

The paper proceeds as follows. Section 2 describes the NPIV framework and reviews previous estimators. Section 3 introduces our new estimator. Section 4 describes our debiasing procedure to obtain standard errors. Section 5 presents our theoretical results. Section 6 evaluates our method with synthetic data. Section 7 presents our empirical application to demand estimation. Section 8 concludes.

2 Problem Setup

We adopt the nonparametric instrumental variables (NPIV) framework of Newey and Powell (2003). Let $D \in \mathcal{D}$ denote the treatment variable, $Z \in \mathcal{Z}$ the instruments, $X \in \mathcal{X}$ the covariates, and $Y \in \mathbb{R}$ the outcome. Our object of interest is the structural function f_0 satisfying

$$Y = f_0(D, X) + \epsilon, \quad \mathbb{E}[\epsilon | Z, X] = 0. \tag{1}$$

Notice that the NPIV framework already imposes a substantive restriction: the structural function exhibits heterogeneity only in (D, X) with unobserved variables entering additively through ϵ. Including a very rich covariate set X weakens this heterogeneity restriction and also possibly helps secure the exogeneity restriction, $\mathbb{E}[\epsilon | Z, X] = 0$.

We are often interested in scalar summaries of f_0, like an average derivative. However, the full structural function itself can also be of independent interest, e.g. for predicting counterfactual outcomes, or flexibly assessing heterogeneity in treatment effects. We now provide some examples:

Example 1 (Demand Estimation)**.** *For a demand estimation problem, Y might be market share for a good, D endogenous prices, and X other market-level covariates. For example, in the strawberry demand setting of Compiani (2022), Y is market share for organic/non-organic strawberries, D is prices of organic/non-organic strawberries, and X includes a measure of taste for organic products, availability of other fruit, and market-level income. Common instruments Z are the price of the same product in nearby markets (Hausman, 1996), and wholesale prices faced by retailers. In this setting, $f_0(D, X)$ is the demand function, and the average derivative with respect to D is the average price elasticity of demand.*

Example 2 (Consumption out of Permanent Income)**.** *Instrumental variables have been widely used to disentangle transitory and permanent components of income (Dynan et al., 2004; Blundell et al., 2008; Straub, 2019). In this setting, Y is household consumption, D is household income, and Z is an instrument for permanent income such as lagged or future income. Relevant covariates X include, age, family size, education, and asset position. The marginal propensity to consume out of permanent income is the average*

derivative of the structural function $f_0(D, X)$ with respect to D. The heterogeneity in X is of particular interest, including how strongly the spending response depends on liquid assets and debt.

While the covariates X are important in many applications, without loss of generality, we shorten notation by writing D for (D, X) and Z for (Z, X).[2]

There are two equivalent ways to express f_0 as the solution to an optimization problem. First, (1) implies that $\mathbb{E}[Y - f_0(D)|Z] = 0$, giving rise to the GMM-style problem:

$$f_0 = \underset{f}{\operatorname{argmin}} \left\{ \max_g \mathbb{E}[g(Z)(Y - f_0(D))] \right\}. \tag{2}$$

Alternatively, the moment condition can be written as:

$$\mathbb{E}[Y|Z] = \mathbb{E}[f_0(D)|Z]. \tag{3}$$

Applying the characterization of the conditional expectation as the best mean squared error predictor, we get the nested regression problem:

$$f_0 = \underset{f}{\operatorname{argmin}} \mathbb{E}\left[(Y - \mathbb{E}[f(D)|Z])^2 \right]. \tag{4}$$

Unfortunately, minimizing the (equivalent) optimization problems (2) and (4) directly over flexible function classes like gradient-boosting or neural networks presents substantial computational challenges. In either case, evaluation of the objective function at each candidate f requires solving a nested optimization problem — for (2), the optimization over g; for (4) to estimate $\mathbb{E}[f(D)|Z]$.

2.1 Previous Approaches to NPIV

A large and growing literature studies methods for solving the NPIV problems (2) and (4). Most existing estimators fall into one of two broad categories. First, methods that impose linearity (possibly in a Sieve or RKHS basis) on either $\mathbb{E}[\cdot|Z]$ or $f_0(D)$. Either linearity restriction will result in a simple and computationally-efficient two stage procedure. The second category are methods that use arbitrary machine learning models for both $\mathbb{E}[\cdot|Z]$ and $f_0(D)$, but incorporated into an iterative or adversarial/minimax training procedure. We summarize existing NPIV estimators in Table 1. Our 2SML estimator is the first to support arbitrary machine learning estimators in both the first and second stage, but while maintaining an easy-to-use, computationally efficient, and

[2]This is without loss of generality for our algorithm, but not for all NPIV algorithms. See Appendix F.

non-iterative two-stage structure.

Table 1: Existing NPIV Estimators

Estimator	$\mathbb{E}[\cdot\|Z]$	$f_0(D)$	Iterative?	Reference
2SLS	Linear	Linear	No	
Split Sample ML IV	Any ML	Linear	No	Chen et al. (2020)
Sieve IV	Sieve	Sieve	No	Newey and Powell (2003)
Kernel IV	RKHS	RKHS	No	Singh et al. (2019)
Sieve Minimum Distance	Sieve	Any ML	No	Chen et al. (2023)
SAGD IV	RKHS	Any ML	No	Fonseca et al. (2024)
Deep Feature IV	Any ML	Any ML	Yes	Xu et al. (2020)
Minimax Approaches	Any ML	Any ML	Yes	See Section 2.1.3
Two Stage ML	Any ML ✓	Any ML ✓	No ✓	This Work

Note: Deep IV (Hartford et al., 2017; Li et al., 2024) performs conditional density estimation instead of estimating $\mathbb{E}[\cdot|Z]$. We discuss Deep IV at the end of Section 2.1.

2.1.1 Two Stage Least Squares

2SLS is an important special case of the optimization problem (4) when f_0 is linear. Let $D \in \mathbb{R}^d$. Applying linearity we have:

$$\min_{\beta \in \mathbb{R}^d} \mathbb{E}\left[\left(Y - \mathbb{E}[\beta^\top D|Z]\right)^2\right] = \min_{\beta \in \mathbb{R}^d} \mathbb{E}\left[\left(Y - \beta^\top \mathbb{E}[D|Z]\right)^2\right].$$

In this last expression, $\mathbb{E}[D|Z]$ is the traditional first-stage regression of D on Z. Importantly, we can estimate the first-stage once, and then optimize over β afterward, resulting in a computationally-efficient algorithm. Traditional 2SLS also approximates $\mathbb{E}[D|Z]$ with a linear model, but Chen et al. (2020) shows that with appropriate sample-splitting, we can fit $\mathbb{E}[D|Z]$ using arbitrary machine learning models. However, a linear model for f_0 will usually be high misspecified, especially with high-dimensional covariates. Later, in our empirical application, we find that imposing a misspecified linear model for f_0 leads to very strong attenuation of the average price elasticity.

Previous methods have introduced non-linearity into f_0 by representing the structural function as linear in a fixed transformation of D and X, such as a sieve (Newey and Powell, 2003; Ai and Chen, 2003, 2007) or RKHS basis Singh et al. (2019). These methods still have a simple two-stage procedure, but now the first stage requires predicting every element of the transformation instead of just D. One of our key empirical findings is that these fixed transformation are insufficient for modeling the structural function in finite samples — this finding is corroborated by recent theoretical results for NPIV in Kim et al. (2025).

2.1.2 Projected Loss Minimization

The methods we have discussed so far impose linearity in $f_0(D)$ to obtain a computationally-efficient two-stage procedure, while allowing $\mathbb{E}[\cdot|Z]$ to be arbitrary. In an important extension, Chen and Ludvigson (2009) and Chen et al. (2023) show that we can alternatively impose linearity in $\mathbb{E}[\cdot|Z]$, while allowing $f_0(D)$ to be arbitrary. The algorithm minimizes the mean squared error of prediction Y given D, but where the predictions are first projected onto a basis $\phi(Z)$ — therefore we call this optimization problem "Projected Loss Minimization". This approach has been adopted as a sub-step of more complicated iterative methods like Xu et al. (2020); Bakhitov and Singh (2022), and will be the starting place for our method that we introduce in Section 3.

We now describe the approach. The challenge with solving (4) when representing f_0 with a machine learning model is that for each candidate function $f(D)$, we have to solve a nested regression problem to estimate $\mathbb{E}[f(D)|Z]$. However, if we estimate $\mathbb{E}[f(D)|Z]$ using ridge regression in some basis ϕ, this has a closed-form solution, resulting in a straightforward convex optimization problem.

Let $\phi : \mathcal{Z} \to \mathbb{R}^{d_\phi}$ denote a feature map. Let $\mathcal{T}_\phi$ denote the operator that maps $f \in \mathcal{F}$ onto the best approximation of $\mathbb{E}[f(D)|Z]$ that is linear in ϕ. In other words $(\mathcal{T}_\phi f)(z) = \phi(z)^\top \beta(f)$, where

$$\beta(f) := \underset{\beta \in \mathbb{R}^{d_\phi}}{\operatorname{argmin}} \mathbb{E}[(f(D) - \phi(Z)^\top \beta)^2].$$

In the projected loss minimization framework, we replace (4) with:

$$\min_{f \in \mathcal{F}} \mathbb{E}\left[\left(Y - (\mathcal{T}_\phi f)(Z)\right)^2\right]. \tag{5}$$

The finite sample version of (5) can be solved efficiently, even over complicated function classes $\mathcal{F}$: given a sample of n independent observations $(d_i, z_i, y_i)_{i=1}^n$, let $y \in \mathbb{R}^n$ denote the outcome vector and $\Phi \in \mathbb{R}^{n \times d_\phi}$ the feature matrix with i-th row $\phi(z_i)^\top$. With slight abuse of notation, let $f \in \mathbb{R}^n$ denote the vector with i-th element $f(d_i)$. For $\lambda \geq 0$, define the (regularized) projection matrix:

$$P_\phi = \Phi(\Phi^\top \Phi + \lambda I)^+ \Phi^\top, \tag{6}$$

where $+$ denotes the Moore-Penrose pseudoinverse.[3] The sample analog of (5) is

$$\min_{f \in \mathcal{F}} \frac{1}{n} \|y - P_\phi f\|_2^2. \tag{7}$$

The projected loss (7) is a convex optimization problem amenable to standard machine learning algorithms including gradient boosting and neural networks.

An exactly analogous projection approach works for solving the GMM-type problem Equation (2), resulting in the finite sample problem:

$$\min_{f \in \mathcal{F}} \frac{1}{n} \|P_\phi (y - f)\|_2^2, \tag{8}$$

The two optimization problems (7) and (8) have identical gradients, and so are equivalent. Chen and Ludvigson (2009); Chen et al. (2023) solve Equation (8) using sieves and so they call their method "Sieve Minimum Distance". Fonseca et al. (2024) solves a slightly different optimization problem, but also represents $\mathbb{E}[f(D)|Z]$ via the closed-form ridge regression solution in an RKHS.

2.1.3 Iterative Methods

Notice that for the ϕ-projected optimization problem (5) to approximate the original problem (4) accurately, $\mathbb{E}[f(D)|Z]$ must be well-approximated by linear functions of $\phi(Z)$ uniformly over $f \in \mathcal{F}$. This requirement becomes increasingly restrictive as $\mathcal{F}$ grows more complex. Recent work addresses this limitation by adaptively learning ϕ jointly with optimizing over f. These more complicated procedures still involve solving the projected loss minimization problem (7) as a sub-step.

For example, Xu et al. (2020) and Bakhitov and Singh (2022) employ an iterative procedure. First, given ϕ, they solve (7) for f. Second, given a $\hat{f}$, they update ϕ by solving

$$\min_{\phi, \beta} \frac{1}{n} \|\hat{f} - \Phi\beta\|_2^2. \tag{9}$$

They alternate between these two stages until convergence. This constitutes a bi-level optimization problem, which can be quite difficult to solve (Hong et al., 2023; Petrulionytė et al., 2024).

Several methods (Bennett et al., 2019; Dikkala et al., 2020; Muandet et al., 2020; Liao et al., 2020)

[3] This formulation supports the case where $d_\phi > n$ and infinite-dimensional ϕ. In these settings, the projection can still be computed efficiently as we describe in Appendix A. When $\lambda = 0$, the inverse need not exist; the pseudoinverse instead provides the minimum-norm solution to the implied least-squares problem.

solve (2) directly with minimax optimization. This is equivalent to learning ϕ and f jointly while solving (8):

$$\min_{f \in \mathcal{F}} \max_{\phi} \frac{1}{n} \|P_\phi(y - f)\|_2^2, \tag{10}$$

While theoretically appealing, such minimax formulations present substantial computational challenges in practice. Section 7 of Dikkala et al. (2020) demonstrates that (10) can be implemented using tree ensembles by alternating between first and second stages, as in Xu et al. (2020) — they call this algorithm "Ensemble IV".

Finally, a few methods based on conditional density estimation do not fit neatly into the projection-based framework we've outlined above. Deep IV (Hartford et al., 2017; Li et al., 2024) is a two stage procedure using arbitrary machine learning algorithms, but where the first stage requires estimating the full conditional distribution of the treatments given the instruments. This is a notoriously difficult problem except with very low dimensional instruments, and is more or less infeasible when both treatments and instruments are high-dimensional, as is the case with rich covariates. See Ji et al. (2023) for a related discussion on conditional density estimation for partial identification.

3 Two-Stage Machine Learning (2SML)

Our algorithm, Two-Stage Machine Learning, solves the projected minimization problem (7), but learns ϕ adaptively from the data using the reduced form. Recall that the central challenge with (7) is finding features ϕ such that $\mathbb{E}[f(D)|Z]$ is approximately linear in $\phi(Z)$ for all f. If f is linear, this reduces to estimating $\mathbb{E}[D|Z]$. But for flexible non-linear function classes like tree ensembles or neural networks, we need fixed features ϕ that approximate $\mathbb{E}[f(D)|Z]$ uniformly over all $f \in \mathcal{F}$, a difficult and potentially impossible task.

Our key insight is to side-step this challenge by directly targeting the population minimizer of problem (4), which is $f_0(D)$. From the definition of the NPIV problem, we know that $\mathbb{E}[f_0(D)|Z] = \mathbb{E}[Y|Z]$. This suggests a natural strategy: if we can construct ϕ such that $\mathbb{E}[Y|Z]$ is linear in $\phi(Z)$, then $\mathbb{E}[f_0(D)|Z]$ is guaranteed to be linear in $\phi(Z)$ as well. The following result is immediate:

Proposition 1. *For any $\phi : \mathcal{Z} \to \mathbb{R}^{d_\phi}$, if there exists $\beta_0 \in \mathbb{R}^{d_\phi}$ such that $\mathbb{E}[Y|Z = z] = \phi(z)^\top \beta_0$, then $(\mathcal{T}_\phi f_0)(z) = \mathbb{E}[f_0(D)|Z = z]$. Furthermore, f_0 achieves the minimum of the projected loss (5), and the minimum of (4) and (5) are identical.*

The conditional expectation $\mathbb{E}[Y|Z]$ is the best mean-squared error predictor of Y given Z. Therefore, we can predict Y given Z using machine learning, and then construct a representation ϕ from the resulting predictor (in a way that describe in more detail below).

The observations above motivate our two-stage procedure:

Stage 1 (Reduced Form): Fit a machine learning model $\hat{g}(Z)$, to predict Y given Z. Extract a feature representation $\phi(Z)$ from this predictor such that $\hat{g}(Z) = \phi(Z)^\top \beta$.

Stage 2 (Projected Loss Minimization): Solve the optimization problem:

$$\min_{f \in \mathcal{F}} \frac{1}{n} \|y - P_\phi f\|_2^2$$

with the learned features ϕ from Stage 1 to get an estimate $\hat{f}$.

By learning ϕ directly from the reduced-form relationship, we ensure that the features are well-suited for approximating the conditional expectation of the true structural function, but without the computational complexity of iterative or minimax methods. Note that because we make use of very flexible machine learning function classes like tree ensembles to learn ϕ, the two stages must be fit in separate samples.

Not all ways of constructing the basis ϕ from the predictor $\hat{g}$ are equally good. For example, if $\hat{g}(Z)$ is a sufficiently good predictor, then the 1-dimensional representation $\phi(z) = \hat{g}(z)$ would satisfy the condition in Proposition 1. However, we will later show, empirically and theoretically, that this basis is brittle and can lead to large estimation errors.

Instead, we will exploit the fact that virtually all machine learning algorithms produce predictors of the form $\hat{g}(Z) = \phi(Z)^\top \beta$ for some feature map ϕ that is learned from the data. For example, with gradient-boosted trees, the prediction of the entire ensemble is a linear combination of the output of individual trees. Thus, we could fit a predictor of Y given Z using a gradient-boosted tree ensemble and then take ϕ to be the vector of outputs of each tree in the ensemble. This representation satisfies the conditions of Proposition 1, and we will demonstrate that it works well in practice. Similarly, for neural networks the natural choice of ϕ is the last-layer embedding.

We provide a step-by-step description of our procedure in Algorithm 1.

Algorithm 1 Two Stage Machine Learning (2SML)

1: **INPUT:**
- Dataset with n observations of (Y, D, Z)
- Function classes $\mathcal{G}$ and $\mathcal{F}$
- Projection regularization parameter $\lambda \geq 0$

2: Divide the observations into two disjoint samples.

3: **FIRST STAGE:**
In the first sample, solve the reduced form prediction problem:

$$\hat{g} = \underset{g \in \mathcal{G}}{\text{argmin}} \, \|y - g(Z)\|_2^2,$$

with corresponding representation ϕ such that $\hat{g}(z) = \phi(z)^\top \beta$, for some β.
For example, for gradient-boosted trees, ϕ would be the output of each individual tree.

4: Let Φ be the matrix with rows $\phi(z_i)$ for i in the second sample. Compute the regularized projection matrix:

$$P_\phi = \Phi(\Phi^\top \Phi + \lambda I)^+ \Phi^\top.$$

5: **SECOND STAGE:**
In the second sample, solve the projected loss minimization problem:

$$\hat{f} = \underset{f \in \mathcal{F}}{\text{argmin}} \, \|y - P_\phi f(D)\|_2^2.$$

6: **OUTPUT:** $\hat{f}$, the estimate of the structural function.

3.1 A Brief Preview: The Importance of ML for the Reduced Form

On what kind of datasets are machine learning models necessary compared to sieve or kernel models which are also nonparametric? Recall that the assumption underlying methods that use sieve or kernels bases for $\phi(Z)$ — including Newey and Powell (2003); Singh et al. (2019); Chen et al. (2023) — is that projecting onto this fixed $\phi(Z)$ is sufficient to model $\mathbb{E}[f(D)|Z]$ for all f. In particular, because $\mathbb{E}[f_0(D)|Z] = \mathbb{E}[Y|Z]$, this assumption implies that ridge regression of Y on $\phi(Z)$ must be the best out-of-sample predictor of Y given Z. If instead tree ensembles provide a significantly better predictor of Y given Z, then this is direct evidence that using a sieve/kernel for $\phi(Z)$ is insufficient.

In a brief preview of our empirical application, we now demonstrate that gradient-boosted tree ensembles (GBoost) are a substantially better predictor for the reduced form in the demand dataset from Compiani (2022) using NielsenIQ scanner data. This dataset has [redacted] observations, Y is market-share for organic strawberries, and Z is 9-dimensional, with 5 instruments and 4 covariates — we defer a complete description of the setting to Section 7. We compare the reduced form

MSE of GBoost to (1) ridge regression on the spline basis from Chen et al. (2023); (2) kernel ridge regression. For each of these predictors, we test for a statistically-significant difference in out-of-sample mean-squared error vs GBoost using a one-sided permutation test. If this permutation test rejects, this is evidence that the corresponding $\phi(Z)$ does not sufficiently model $\mathbb{E}[f_0(D)|Z]$. We compare this to ridge regression in our proposed representation $\phi(Z)$, which uses the output of the individual trees in the GBoost model. For our Two-Stage Machine Learning approach to work, this should predict at least as well as the original GBoost model. We summarize the results in Table 2.

The gradient-boosted tree ensemble is an enormously better predictor for the reduced form task than splines or kernel ridge. This suggests that solving NPIV with a spline or kernel basis for the instruments will be *severely* mis-specified, even if using a gradient-boosting or neural network model for the structural function as in Chen et al. (2023). By contrast, ridge regression on our proposed GBoost basis actually improves on the performance of the baseline GBoost model, demonstrating the strength of our proposed basis. For comparison, at the end of Table 2, we also include the MSE of a neural network predictor. The neural network outperforms the sieve and kernel methods, but still has a holdout R^2 that is 0.13 worse than that of GBoost.

Table 2: Reduced Form Prediction Accuracy for Compiani (2022)

	Predictor	Hold-out R^2	Hold-out MSE	$\hat{p}$ for $H_0 =$ MSE $\leq$ GBoost
With trees	GBoost	[redacted]	[redacted]	-
	Ridge on GBoost Basis	[redacted]	[redacted]	[redacted]
	Spline Ridge	[redacted]	[redacted]	[redacted]
	Kernel Ridge	[redacted]	[redacted]	[redacted]
	Neural Network	[redacted]	[redacted]	[redacted]

Notes: We randomly use 75% of the data for training, and the remaining 25% as the hold-out. All hyperparameters are chosen with cross-validation in the traing set. We test for a statistically-significant difference in MSE vs GBoost using a permutation test with $10,000$ permutations. The resulting p-value is an estimate, and an upper 95% confidence interval on an estimated 0 with this many permutations is 0.00037.

4 Standard Errors for Linear Functionals with Debiasing

Often, we are interested in a scalar summary of the structural function f_0. Let θ be a continuous linear functional over $f : \mathcal{D} \to \mathbb{R}$ that, for some m, takes the form:

$$\theta(f) := \mathbb{E}[m(f, D)].$$

We consider estimation and inference for the estimand $\theta_0 := \theta(f_0)$.

Example (Impulse Response)**.** *Consider a setting with $f_0(D, X)$ where $D \in \mathbb{R}$. Then,*

$$\theta(f_0) = \mathbb{E}[f_0(D+1, X) - f_0(D, X)]$$

is the average impulse response for the structural function.

Example (Own- and Cross-Price Elasticities)**.** *Consider a demand estimation setting with two goods where Y is the log market share for good 1. Let the structural (demand) function be $f_0(D_1, D_2, X)$, where D_1 is the log price of good 1, D_2 is the log price of good 2, and X are exogenous market characteristics. Then:*

$$\theta_o(f_0) = \mathbb{E}\left[\frac{\partial f_0(D_1, D_2, X)}{\partial D_1}\right] \quad , \quad \theta_c(f_0) = \mathbb{E}\left[\frac{\partial f_0(D_1, D_2, X)}{\partial D_2}\right]$$

are the own-price and cross-price elasticities respectively.

The simple *plug-in* estimator for θ_0 using the output of Algorithm 1 is:

$$\hat{\theta}_P = \frac{1}{n}\sum_{i=1}^{n} m(\hat{f}, D_i).$$

However, even if the estimator of $\hat{f}$ is consistent for f_0, machine learning estimators usually leverage bias in order to reduce variance. This may be optimal for uniformly estimating f_0, but the bias will pass through to the plug-in estimate. As a result, $\hat{\theta}_P$ will generally not be asymptotically normal, and it is unclear how to obtain valid confidence intervals.

We now provide a debiasing procedure that corrects for the bias when estimating θ_0, resulting in an asymptotically normal estimator. Our approach is based on the debiasing framework of Chernozhukov et al. (2023). If f_0 were a conditional mean instead of the solution to the NPIV conditional moment equation, then we would have a standard double/debiased machine learning problem (Chernozhukov et al., 2018). In that standard setting, debiasing involves estimating a nuisance function called the *Riesz representer* of the functional θ[4]. Let $L_2(D)$ be the Hilbert space of functions of D with finite second moment. When θ is linear and continuous, there exists a unique function — the Riesz representer — $\alpha_0 \in L_2(D)$ such that:

$$\theta(f) = \mathbb{E}[\alpha_0(D) f(D)], \quad \forall f \in L_2(D).$$

[4]For earlier uses of the Riesz representer for semiparametric estimation in econometrics see Ai and Chen (2003); Chen et al. (2006); Ai and Chen (2007).

We can estimate α_0 by minimizing the *Riesz loss* as in Chernozhukov et al. (2022b, 2021, 2022a):

$$\alpha_0 = \underset{\alpha \in L_2(D)}{\operatorname{argmin}} \mathbb{E}[(\alpha(D) - \alpha_0(D))^2] = \underset{\alpha \in L_2(D)}{\operatorname{argmin}} \mathbb{E}[\underbrace{\alpha(D)^2 - 2m(\alpha; D)}_{\text{Riesz loss}}]. \tag{11}$$

Remarkably, even though α_0 is unknown, minimizing the average (observable) Riesz loss is equivalent to minimizing the average squared error for α_0.[5]

In the NPIV setting, where f_0 is defined by a conditional moment equality, α_0 is no longer the relevant debiasing nuisance. Instead, Severini and Tripathi (2012) show that $\theta(f_0)$ is identified if and only if there exists a $q_0 \in L_2(Z)$ such that $\mathbb{E}[q_0(Z)|D] = \alpha_0(D)$.[6] Given estimates $\hat{f}$ for f_0 and $\hat{q}$ for q_0, the resulting debiased NPIV estimator from Chernozhukov et al. (2023) is:

$$\hat{\theta}_D = \frac{1}{n} \sum_{i=1}^{n} m(\hat{f}; D_i) + \hat{q}(Z_i)(Y_i - \hat{f}(D_i)). \tag{12}$$

All that remains is to obtain an estimate $\hat{q}$ of the debiasing nuisance. Since $\mathbb{E}[q_0(Z)|D] = \alpha_0(D)$ is also a conditional moment equation, we can use a two stage machine learning procedure, just as we used to estimate $\hat{f}$. Recall that the structural function f_0 is defined by the squared loss minimization problem in Equation (4), which given a learned representation ϕ, we replaced with the projected loss minimization problem Equation (5). A similar argument holds for q_0, but applied to the Riesz loss (11) instead of the squared loss, and with the roles of D and Z reversed. This idea — that the same strategy can be used to estimate both $\hat{f}$ and $\hat{q}$ — is not new, and was applied to minimax estimators in Ghassami et al. (2022); Bennett et al. (2022). Our procedure for estimating $\hat{q}$ works as follows:

Stage 1 (Riesz Regression): Fit a machine learning model $\hat{\alpha}(D)$, that minimizes the Riesz loss (11). Extract a feature representation $\varphi(D)$ from this predictor such that $\hat{\alpha}(D) = \varphi(D)^\top \beta$.

Stage 2 (Projected Loss Minimization): Minimize the projected Riesz loss using the learned features φ from Stage 1 to get an estimate $\hat{q}$.

The projected Riesz loss minimization problem in the sample takes a convenient form by exploiting linearity — we defer the derivation to Appendix B. Let $\varphi \in \mathbb{R}^{n \times d_\varphi}$ be the matrix with rows $\varphi(D_i)$, and let $\varphi_{\text{cf}} \in \mathbb{R}^{n \times d_\varphi}$, be the matrix with rows equal to $m(\varphi, D_i)$. Let $\bar{\varphi}_{\text{cf}} \in \mathbb{R}^{d_\varphi}$ be the sample

[5] Chen et al. (2014); Chen and Pouzo (2015) also estimate the Riesz representer directly with least squares.
[6] See Ai and Chen (2012) for a related derivation of the efficient score.

average of φ_{cf}. Define:

$$P_\varphi \coloneqq \varphi(\varphi^\top\varphi + \lambda I)\varphi^\top, \quad P_{\text{cf}} \coloneqq \bar{\varphi}_{\text{cf}}(\varphi^\top\varphi + \lambda I)\varphi^\top.$$

To shorten notation, we write q for the vector with entries $q(Z_i)$. Then the second stage optimization problem is:

$$\min_{q\in\mathcal{Q}} \left\{ \frac{1}{n} q^\top P_\varphi q - 2P_{\text{cf}} q \right\}. \tag{13}$$

In the next section, we will establish rates of convergence for both $\hat{f}$ and $\hat{q}$ using our two stage procedure, along with conditions that guarantee that the debiased estimate $\hat{\theta}$ is asymptotically normal. Valid confidence intervals can be computed with standard error $\hat{\sigma}/\sqrt{n}$, where

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} \left(m(\hat{f}; D_i) + \hat{q}(Z_i)(Y_i - \hat{f}(D_i)) - \hat{\theta}_D \right)^2. \tag{14}$$

Note that both $\hat{\theta}_D$ and $\hat{\sigma}^2$ should be estimated using cross-fitting as in Chernozhukov et al. (2023), but we suppress cross-fitting in the main text for notational convenience. We describe how to compute the cross-fit estimate in Appendix D.

5 Theoretical Guarantees

In this section, we present a formal definition of our Two-Stage Machine Learning estimator minimizing a general loss function. Special cases include (1) the squared loss for the outcomes, resulting in an estimator for f_0; and (2) the Riesz loss, resulting in an estimator for q_0. We prove finite-sample-valid rates of convergence for our estimates, and give examples instantiating these rates for different choices of machine learning algorithms. Finally, by applying these rates for the nuisances, we establish conditions under which the debiased estimate $\hat{\theta}_D$ is asympotically normal with standard error $\hat{\sigma}/\sqrt{n}$. Proofs are deferred to Appendix C.

Notation: Let $L_2(D)$ denote the Hilbert space of functions $f : \mathcal{D} \to \mathbb{R}$ that are square integrable with respect to the distribution of D, i.e. $\mathbb{E}[f(D)^2] < \infty$. The inner product in $L_2(D)$ is $\langle f, g\rangle \coloneqq \mathbb{E}[f(D)g(D)]$, and we denote the usual Hilbert space norm $\|f\|_D \coloneqq \mathbb{E}[f(D)^2]^{1/2}$. We use the same notation for $L_2(Z)$.

5.1 General Loss and Target Function

We will consider loss functions defined on functions $g \in L_2(Z)$. For $g \in L_2(Z), z \in \mathcal{Z}, y \in \mathcal{Y}$, let $\ell(g; z, y) \in \mathbb{R}$ denote the loss. Define the population risk:

$$L(g) \coloneqq \mathbb{E}[\ell(g; Z, Y)].$$

For $g, g' \in L_2(Z)$, we will write $DL(g)[g']$ for the directional derivative of the risk functional L at g in the direction g':

$$DL(g)[g'] \coloneqq \frac{d}{dt} L(g + tg')\Big|_{t=0}.$$

We impose the following assumptions on the population risk:

Assumption 1 (Requirements on the Risk)**.** *The population risk $L(g)$ satisfies the following properties:*

1. *$L(g)$ is A-smooth: for all g, g',*

$$L(g) - L(g') - DL(g')[g - g'] \leq \frac{A}{2}\|g - g'\|_Z^2.$$

2. *$L(g)$ is B-strongly convex: for all g, g',*

$$L(g) - L(g') - DL(g')[g - g'] \geq \frac{B}{2}\|g - g'\|_Z^2.$$

These are standard requirements needed to apply finite-sample empirical risk minimization bounds.

We now define the target estimand of interest (corresponding to the structural function in NPIV) for this general loss function. Strong convexity and smoothness of the loss from Assumption 1 guarantee the existence of a unique minimizer of the population risk over $L_2(Z)$, which we denote:

$$g_0 \coloneqq \operatorname*{argmin}_{g \in L_2(Z)} L(g).$$

Define the conditional expectation operator $\mathcal{T} : L_2(D) \to L_2(Z)$ with:

$$(\mathcal{T} f)(z) = \mathbb{E}[f(D)|Z = z].$$

Then the target function f_0 is some function in $L_2(D)$ that satisfies the conditional moment equa-

tion,

$$\mathcal{T} f_0 = g_0.$$

While g_0 is guaranteed to be unique, the target function f_0 need not be the only function in $L_2(D)$ that satisfies this equality.

Example (NPIV Structural Function)**.** *The NPIV structural function f_0 that satisfies $\mathbb{E}[f_0(D)|Z] = \mathbb{E}[Y|Z]$ is a special case for the loss function $\ell_{sq}(g; z, y) := (y - g(z))^2$ for $g \in L_2(Z)$ and corresponding risk $L_{sq}(g)$. We have $\mathbb{E}[Y|Z]$ is the minimizer of $L_{sq}(g)$ over $L_2(Z)$.*

Example (Debiasing Nuisance)**.** *Abusing notation slightly by reversing the roles of Z and D, the debiasing nuisance function q_0 that satisfies $\mathbb{E}[q_0(Z)|D] = \alpha_0(D)$ is a special case for the loss function $\ell_{rr}(\alpha; d) := \alpha(d)^2 - 2m(\alpha; d)$ for $\alpha \in L_2(D)$ and corresponding risk $L_{rr}(\alpha)$. We have that the Riesz representer α_0 of the functional θ is the minimizer of $L_{rr}(\alpha)$ over $L_2(D)$.*

As these two loss functions are quadratic in their first argument, they both satisfy Assumption 1. By encompassing both losses in a general framework, the rates of convergence we establish below for our 2SML algorithm will apply to estimating both nuisances.

5.2 First and Second Stage Function Classes

To formalize the representation learning step from the first stage while accommodating features ϕ that are potentially infinite-dimensional, we use reproducing kernel Hilbert spaces (RKHS's) (Schölkopf and Smola, 2002). For a symmetric, positive semi-definite (PSD) kernel k defined on $\mathcal{Z}$, let the corresponding RKHS be denoted $\mathcal{H}_k$. We define the first stage function class $\mathcal{G} \subseteq L_2(Z)$, with the following requirement:

Assumption 2 (First Stage Function Class)**.**

$$\mathcal{G} \subseteq \bigcup_{k:\mathbb{E}[k(Z,Z)]<\infty} \mathcal{H}_k.$$

Note that this restriction is extremely weak; for example, $\bigcup_{k:\mathbb{E}[k(Z,Z)]<\infty} \mathcal{H}_k$ contains all continuous functions (and many discontinuous ones). An important special case that subsumes most machine learning algorithms is when $\mathcal{G}$ is defined over finite-dimensional kernels of a fixed dimension:

Proposition 2. *Fix $1 \leq d_\phi < \infty$. Then,*

$$\{g(z) = \phi(z)^\top \beta \text{ s.t. } \phi : \mathcal{Z} \to \mathbb{R}^{d_\phi}, \mathbb{E}[\|\phi(Z)\|_2^2] < \infty, \beta \in \mathbb{R}^{d_\phi}\} \subseteq \bigcup_{k:\mathbb{E}[k(Z,Z)]<\infty} \mathcal{H}_k.$$

Proof. For each ϕ, we can define the kernel $k(x, y) = \phi(x)^\top \phi(y)$. Then apply Cauchy-Schwarz. □

As concrete examples, this finite-dimensional case would include any neural network with embedding width d_ϕ and bounded weights, or any gradient boosting ensemble composed of d_ϕ individual bounded boosters. These examples cannot be written as a single RKHS because they choose their basis ϕ adaptively from the data. Our general formulation for $\mathcal{G}$ also allows using an infinite-dimensional RKHS in the first stage (as in Singh et al. (2019)), or performing a kernel-learning step to adaptively choose between infinite-dimensional RKHS's (Lanckriet et al., 2004). These infinite-dimensional settings still have natural feature maps: for a PSD kernel k and corresponding RKHS $\mathcal{H}_k$, there exists a feature map $\phi : \mathcal{Z} \to \mathcal{H}_k$ with the special property that for any $g \in \mathcal{H}_k$, $g(z) = \langle \phi(z), g \rangle$.

Our second stage function class is $\mathcal{F} \subseteq L_2(D)$.

Assumption 3 (Requirements on the Function Classes)**.** *We require that:*

1. *(Conditions for Minimizers)* $\mathcal{G}$ *and* $\mathcal{F}$ *are non-empty, closed, and convex,*
2. *(Lipschitz)* $\ell(g; z, y)$ *is* C*-Lipschitz with respect to its first argument over* $g \in \mathcal{G}$*,*
3. *(Boundedness)* $\sup_{g \in \mathcal{G}} \|g\|_\infty \leq 1$, $\sup_{f \in \mathcal{F}} \|f\|_\infty \leq 1$*,*
4. *(Realizability)* $f_0 \in \mathcal{F}$*.*

The first condition guarantees the existence of minimizers of the population losses. This could be replaced with directly asserting the existence of relevant minimizers. The second and third conditions are used in the arguments for achieving fast rates with empirical risk minimization. Note that without loss of generality, we can consider functions uniformly-bounded by a constant instead of 1, but we choose 1 for notational simplicity.

5.3 First and Second Stage Optimization Problems

Now we define the relevant empirical risk minimizers for our 2SML estimator. Our estimator uses two independent samples of Z, D, Y, one for the first stage with m iid observations, and one for the second stage with n iid observations. We will use $\hat{\mathbb{E}}_m[\cdot]$ and $\hat{\mathbb{E}}_n[\cdot]$ to denote sample averages for the first and second sample respectively. We will write the empirical risk: $L_n(g) := \hat{\mathbb{E}}_n[\ell(g; Z, Y)]$ and likewise for $L_m(g)$.

The first stage of 2SML solves the following empirical risk minimization problem in the first sample with m observations:

$$\hat{g} \in \operatorname*{argmin}_{g \in \mathcal{G}} L_m(g).$$

By construction, there exists a PSD kernel $\hat{k}$ with $\mathbb{E}[\hat{k}(Z,Z)] < \infty$, and a corresponding RKHS $\mathcal{H}_{\hat{k}}$ such that $\hat{g} \in \mathcal{H}_{\hat{k}}$. Let $\hat{\phi} : \mathcal{Z} \to \mathcal{H}_{\hat{k}}$ be the corresponding feature map.

Next, we formally define the projection part of the projected loss minimization framework. We define the operator $\mathcal{T}_{\hat{\phi}} : L_2(D) \to \mathcal{H}_{\hat{k}}$:

$$\mathcal{T}_{\hat{\phi}} f \coloneqq \operatorname*{argmin}_{g \in \mathcal{H}_{\hat{k}}} \{\mathbb{E}[(f(D) - g(Z))^2]\}.$$

Equivalently, $\mathcal{T}_{\hat{\phi}} f$ is the projection in the $L_2(Z)$ norm of $\mathbb{E}[f(D)|Z]$ onto $\mathcal{H}_{\hat{k}}$. We also define the sample version of this operator. Let $\|\cdot\|_{\mathcal{H}_{\hat{k}}}$ denote the usual Hilbert space norm for $\mathcal{H}_{\hat{k}}$. Define the norm ball of radius b, $\mathcal{H}_{\hat{k}}^b \coloneqq \{g \in \mathcal{H}_{\hat{k}} : \|g\|_{\mathcal{H}_{\hat{k}}} \leq b\}$. Then define,

$$\hat{\mathcal{T}}_{\hat{\phi}} f \coloneqq \operatorname*{argmin}_{g \in \mathcal{H}_{\hat{k}}^b} \{\hat{\mathbb{E}}_n[(f(D) - g(Z))^2]\}. \tag{15}$$

In practice, instead of the constrained form with radius b, we implement the operator $\hat{\mathcal{T}}_{\hat{\phi}}$ using the equivalent penalized form with hyperparameter λ. This has the advantage of being easily computable in closed-form with the familiar projection matrix from Algorithm 1 (see Appendix A for the closed-form solution in an infinite-dimensional RKHS).

We impose the following requirements on the kernel $\hat{k}$ and the norm radius b:

Assumption 4 (Regularity Conditions on $\mathcal{H}_{\hat{k}}^b$)**.** *We impose the following conditions:*

1. *The loss $\ell(g; z, y)$ is C-Lipschitz on $\mathcal{H}_{\hat{k}}^b$,*
2. $\sup_{g \in \mathcal{H}_{\hat{k}}^b} \|g\|_\infty \leq 1$,
3. *The radius b is sufficiently large such that $\mathcal{T}_{\hat{\phi}} f_0 \in \mathcal{H}_{\hat{k}}^b$.*

The first two requirements are standard boundedness assumptions required for our empirical risk minimization results. They would be satisfied, for example, with $\ell(g; z, y) = (g(z) - y)^2$ when $\hat{k}$ is continuous and $\mathcal{Z}$ is compact. The third condition is a realizability condition to streamline the presentation of our main results — we demonstrate how to relax this assumption in Appendix C.3.

To construct our final estimate of the target function, the second stage of 2SML solves the following empirical risk minimization problem in the second sample with n observations:

$$\hat{f} \in \underset{f \in \mathcal{F}}{\operatorname{argmin}} \, L_n(\hat{\mathcal{T}}_{\hat{\phi}} f).$$

5.4 Convergence in the Weak Metric

Following the previous NPIV literature as in Newey and Powell (2003) and Dikkala et al. (2020), we first prove convergence in the weak metric — $\|\mathcal{T}(\hat{f} - f_0)\|_Z$ — and then in Section 5.5 impose additional conditions to establish convergence in the strong metric — $\|\hat{f} - f_0\|_D$.

We use nonasymptotic techniques to establish high-probability bounds on the error. The bounds (and resulting rate of convergence) depend on the complexity of the function classes involved. The measure of complexity we adopt is the *critical radius*, a standard tool from statistical learning theory; see Wainwright (2019) and Foster and Syrgkanis (2023) for a review. Define the local Rademacher complexity (Bartlett et al., 2005) for a function class $\mathcal{A} \subseteq L_2(Z)$, sample size n, and radius $\delta \geq 0$:

$$\mathcal{R}_n(\mathcal{A}, \delta) := \mathbb{E}\left[\sup_{g \in \mathcal{A}: \|g\|_Z \leq \delta} \left| \frac{1}{n} \sum_{i=1}^{n} \epsilon_i g(z_i) \right| \right],$$

where the expectation is taken over both independent Rademacher random variables $\epsilon_1, ..., \epsilon_n$, and over n iid observations of Z, $z_1, ..., z_n$. The critical radius, δ_n, of the class $\mathcal{A}$ with sample size n is the smallest solution to the inequality:

$$\frac{\mathcal{R}_n(\mathcal{A}, \delta)}{\delta} \leq \delta. \tag{16}$$

We are now ready to state our main result.

Theorem 1 (Weak Convergence of 2SML with General Loss)**.** *Let δ_n^b be an upper bound on the critical radius of $\mathcal{H}_{\hat{k}}^b$. Given Assumptions 1, 2, 3, and 4, with probability at least $1 - 2\eta$ we have:*

$$\|\mathcal{T}(\hat{f} - f_0)\|_Z \leq O\left(\|\hat{g} - g_0\|_Z + \delta_n^b + \Delta_{\hat{\phi}} + \sqrt{\frac{\log(1/\eta)}{n}} \right),$$

where:

$$\Delta_{\hat{\phi}} := L(\mathcal{T}\hat{f}) - L(\hat{\mathcal{T}}_{\hat{\phi}}\hat{f}).$$

The first two terms in the bound are: (1) the first-stage error, (2) the complexity of the projection

step. Terms (1) and (2) depend on the complexity of the function class $\mathcal{G}$. We will discuss concrete rates for these terms next, but first we comment on the third term, $\Delta_{\hat{\phi}}$, which is essentially a measure of whether ridge regression of $\hat{f}(D)$ on $\hat{\phi}(Z)$ is a good approximation of $\mathbb{E}[\hat{f}(D)|Z]$.

We need $\Delta_{\hat{\phi}}$ to be less than or equal to zero asymptotically to guarantee convergence of $\hat{f}$:

Assumption 5. $\Delta_{\hat{\phi}} \leq 0$.

This condition is testable from the data without knowledge of the true target function f_0. We describe a feasible permutation test in Appendix C.4. A sufficient condition is that ridge regression in $\hat{\phi}(Z)$ is the best mean-squared predictor of $\hat{f}(D)$. By construction this is guaranteed to be true for f_0, but it must also hold for $\hat{f}$. We find in practice that when we fit the reduced form using gradient-boosted trees and take $\hat{\phi}$ to be the output of each individual tree in the ensemble, Assumption 5 always holds, both in our synthetic datasets in Section 6, and on the real world datasets from Compiani (2022) and Card (1995). However, Assumption 5 can fail to hold for other function classes when $\hat{\phi}$ is too simple relative to $\mathcal{F}$ — one such example is when $\hat{\phi}$ is chosen to be the 1-dimensional basis $\hat{\phi}(z) = \hat{g}(z)$. We also find violations of Assumption 5, when $\hat{\phi}$ is a sparse Lasso basis and the regularization hyperparameter is chosen to be too high. We find that undersmoothing the Lasso prevents these violations in practice.

We now turn to establishing rates of converges for the first two terms, $\|\hat{g} - g_0\|_Z$ and δ_n^b. The term $\|\hat{g} - g_0\|_Z$ is the L_2 error of the first stage, which can be controlled using any off-the-shelf machine learning result that provides a rate of convergence. This includes results giving rates without the critical radius machinery — for a recent example using boosting see Luo et al. (2025). Note that since we estimate $\hat{g}$ using the first sample, the rate of convergence depends on m. Very often these rates take the form $O\left(d\sqrt{\frac{\log m}{m}}\right)$, where d is some measure of the "dimension" of the machine learning algorithm.

For completeness, we provide a standard critical radius bound for the first stage. Define the star hull of a set $\mathcal{A}$ as $\text{star}(\mathcal{A}) \coloneqq \{\alpha g | g \in \mathcal{A}, \alpha \in [0,1]\}$.

Proposition 3 (First Stage)**.** *Let g^* be the minimizer of $L(g)$ over $\mathcal{G}$. Let $\delta_m^{\mathcal{G}}$ be an upper bound on the critical radius of star$(\mathcal{G} - g^*)$. Given Assumptions 1 and 3, with probability at least $1 - \eta$:*

$$\|\hat{g} - g_0\|_Z \leq O\left(\min_{g \in \mathcal{G}} \|g - g_0\|_Z + \delta_m^{\mathcal{G}} + \sqrt{\frac{\log(1/\eta)}{m}}\right).$$

The second term in Theorem 1, δ_n^b is the critical radius of the Hilbert space ball $\mathcal{H}_{\hat{k}}^b$. Critical radii

for different kernels have been extensively characterized, see for example Bartlett et al. (2005); Wainwright (2019). Note that $\hat{k}$ is chosen through the first stage, and so δ_n^b depends on $\mathcal{G}$. We now provide a specific example of $\mathcal{G}$ with rates of convergence for the first two terms in Theorem 1.

Example (Tree Ensembles): Suppose we can achieve zero approximation error, i.e. $g^* = g_0$, when $\mathcal{G}$ is the set of all tree ensembles composed of a convex combination of T individual trees. Any $g \in \mathcal{G}$ can be written as a linear function of T trees, satisfying the conditions in Proposition 2 with $d_\phi = T$. For concreteness, we can apply the result from Syrgkanis and Zampetakis (2020) for individual trees with t leaves and d binary input features to get:

$$\delta_m^{\mathcal{G}} \leq O\left(\sqrt{\frac{Tt\log(dt)\log(m)}{m}}\right).$$

This is a typical result for the complexity of trees — compare to Theorem 1 from Li et al. (2024).

For the second stage, we have $\mathcal{H}_{\hat{k}}^b$ is a subset of all linear functions of the features $\hat{\phi}$ with dimension T, and so using standard results (Wainwright, 2019), we can take:

$$\delta_n^b \leq \sqrt{\frac{T}{n}}.$$

Putting these together with Assumption 5, we get:

$$\|\mathcal{T}(\hat{f} - f_0)\|_Z \leq O_p\left(\sqrt{\frac{Tt\log(dt)\log(m)}{m}} + \sqrt{\frac{T}{n}}\right).$$

The rate is dominated by the first term, the complexity of the reduced form prediction task. If we begin with N total samples split evenly, then $\hat{f}$ converges at rate $\sqrt{\log(N)/N}$, which importantly is fast enough to satisfy the rate requirements for valid inference in the next section.

Remark 1. *If the testable condition Assumption 5 holds, then our error bound depends entirely on the difficulty of the reduced-form prediction task. In practice, we use gradient-boosted trees. However, previous NPIV estimators including Newey and Powell (2003), Singh et al. (2019), and Chen et al. (2023) can be written as special cases of the empirical risk minimization setup from Section 5.3. In particular, they correspond to the case where $\mathcal{G}$ is a single RKHS with a given kernel. Therefore, when Assumption 5 holds, they inherit our error bound based on the reduced form. In the settings we consider, gradient-boosted trees provide much better predictions for the reduced form than sieve or kernel function classes. This may partially explain our method's superior performance in practice.*

5.5 Inference for Linear Functionals with Debiasing

We now provide inference guarantees for the estimand from Section 4: the linear functional $\theta_0 := \theta(f_0)$ where f_0 is the NPIV structural function, and where the debiasing nuisance for θ is q_0.

Let $\hat{f}$ and $\hat{q}$ be estimates that solve the second-stage optimization problem in Section 5.3 for the squared loss ℓ_{sq} and the Riesz loss ℓ_{rr} respectively. For $\hat{f}$ let the first stage function class be $\mathcal{G}_{\text{sq}}$, the second stage function class be $\mathcal{F}_{\text{sq}}$, the learned features be $\hat{\phi}$ and the norm ball in the corresponding RKHS be $\mathcal{H}_{\hat{\phi}}^{b_1}$. For $\hat{q}$, define $\mathcal{G}_{\text{rr}}, \mathcal{F}_{\text{rr}}, \hat{\varphi}$, and $\mathcal{H}_{\hat{\varphi}}^{b_2}$ in the same way. Assume that Assumptions 2, 3, 4, and 5 hold for both. Applying Theorem 1 to the two nuisance estimates $\hat{f}$ and $\hat{q}$, we can establish asymptotic normality and valid inference for the debiased point estimate (12). In what follows we will write $\mathcal{T}_{D\to Z}(f)$ for $\mathbb{E}[f(D)|Z]$ and $\mathcal{T}_{Z\to D}(q)$ for $\mathbb{E}[q(Z)|D]$.

Consider the debiased point estimate $\hat{\theta}_D$ from (12) and variance $\hat{\sigma}^2$ from (14) formed using the two nuisance estimates $\hat{f}$ and $\hat{q}$. Chernozhukov et al. (2023) show, under standard regularity conditions (i.e. boundedness of certain moments, described in Appendix C.5), that if the following rate conditions hold:

1. $\|\hat{f} - f_0\|_D = o_p(1)$,
2. $\|\hat{q} - q_0\|_Z = o_p(1)$,
3. $\min(\ \|\mathcal{T}_{D\to Z}(\hat{f} - f_0)\|_Z \|\hat{q} - q_0\|_Z\ ,\ \|\hat{f} - f_0\|_D \|\mathcal{T}_{Z\to D}(\hat{q} - q_0)\|_D\ ) = o_p(n^{-1/2})$,

then we have asymptotic normality:

$$\sqrt{n}(\hat{\theta} - \theta_0) \xrightarrow{d} \mathcal{N}(0, \sigma^2), \text{ and } \hat{\sigma}^2 \xrightarrow{p} \sigma^2,$$

where σ^2 is the variance of the efficient influence function. Therefore, for $a \in [0, 1]$, we get a valid confidence interval:

$$\mathbb{P}\{\theta_0 \in \hat{\theta} \pm c_a \hat{\sigma} n^{-1/2}\} \to 1 - a,$$

where c_a is the $(1 - a/2)$-quantile of the standard normal distribution. Condition 3 is the usual product rate requirement, but allowing a mix of convergence in the strong metric and the projected weak metric. We rely on existing results to relate the weak metric, $\|\mathcal{T}_{D\to Z}(\hat{f} - f_0)\|_Z$ to the strong metric $\|\hat{f} - f_0\|_D$ and similarly for $\hat{q}$. Chen and Pouzo (2012) introduce the *measure of ill-posedness*

with respect to the function class $\mathcal{F}$:

$$\tau := \sup_{f \in \mathcal{F}} \frac{\|f - f_0\|_D}{\|\mathcal{T}_{D \to Z}(f - f_0)\|_Z}.$$

If τ is bounded, the error in the strong metric is controlled by:

$$\|\hat{f} - f_0\|_D \leq \tau \|\mathcal{T}_{D \to Z}(\hat{f} - f_0)\|_Z.$$

Other work use different techniques to secure convergence in the strong metric, like combining Tikhonov regularization and a source condition assumption Bennett et al. (2023); Li et al. (2024). A similar analysis could be applied here, but we use bounded measure of ill-posedness for simplicity.

Proposition 4. *The following conditions on $\hat{f}$ and $\hat{q}$ are sufficient for the rate conditions from Chernozhukov et al. (2023) to hold:*

1. *Either f_0 or q_0 has bounded measure of ill-posedness, i.e.*

$$\min\left(\sup_{f \in \mathcal{F}_{sq}} \frac{\|f - f_0\|_D}{\|\mathcal{T}_{D \to Z}(f - f_0)\|_Z}, \sup_{q \in \mathcal{F}_{rr}} \frac{\|q - q_0\|_Z}{\|\mathcal{T}_{Z \to D}(q - q_0)\|_D}\right) \leq \tau < \infty;$$

2. *If one of f_0 or q_0 does not have a bounded measure of ill-posedness, the corresponding nuisance estimate must still converge in the strong metric but possibly at an arbitrarily slow rate;*

3. *The function classes $\mathcal{G}_{sq}, \mathcal{G}_{rr}, \mathcal{H}^{b1}_{\hat{\phi}}, \mathcal{H}^{b2}_{\hat{\varphi}}$ satisfy the product rate condition,*

$$\max\{\delta_n^{b_1}, \|\hat{g} - g_0\|_Z\} \cdot \max\{\delta_n^{b_2}, \|\hat{\alpha} - \alpha_0\|_D\} = o_p(n^{-1/2}).$$

Proof. By applying Theorem 1 we get $\|\mathcal{T}_{D \to Z}(\hat{f} - f_0)\|_Z = O_p(\delta_n^{b1} + \|\hat{g} - g_0\|_Z + n^{-1/2})$ and $\|\mathcal{T}_{Z \to D}(\hat{q} - q_0)\|_D = O_p(\delta_n^{b2} + \|\hat{\alpha} - \alpha_0\|_D + n^{-1/2})$. If either has bounded measure of ill-posedness, then either $\|\hat{f} - f_0\|_D$ or $\|\hat{q} - q_0\|_Z$ inherits the same rate, so we satisfy Condition 3. If the other does not satisfy any bounded measure of ill-posedness, then we still require that the nuisance converge in the strong metric, but possibly at a much slower rate than the weak metric convergence provided by Theorem 1. □

Remark 2. *The conditions in Proposition 4 are sufficient but not necessary to achieve the product rate requirement. For example, consider a setting where Theorem 1 would guarantee that both nuisances achieve a typical convergence rate of $\sqrt{\log n / n}$ in the weak metric, but that in the strong metric, that rate falls to $n^{-1/6}$. The rate conditions for asymptotic normality are still satisfied. This is similar to the inference results*

for sieves under ill-posedness in Chen and Pouzo (2015).

Remark 3. *The inference guarantees in Bennett et al. (2022) don't require any kind of ill-posedness control, however, they require a stronger identification condition on $\theta(f_0)$ than the existence of q_0 such that $\mathbb{E}[q_0(Z)|D] = \alpha_0(D)$. In their setup, they minimize the Riesz loss, but using a slightly different projection scheme, which they solve using a minimax formulation. In future work, a straightforward extension would be to solve for their proposed debiasing nuisance using our algorithm.*

6 Evaluation in Simulation

We evaluate our 2SML procedure in three ways. First, we can directly assess how well we minimize the NPIV objective (4) out-of-sample using real data from IV applications. This doesn't require access to the true structural function. We defer this evaluation to our empirical application in Section 7.

In this section, we use synthetic and semi-synthetic data to evaluate 2SML in two ways that do require ground-truth access to the true structural function. First, we use a semi-synthetic setup to evaluate how well our estimate of the structural function $\hat{f}$ uniformly approximates the true structural functional using the metric $\|\hat{f} - f_0\|_D$. This is a goal unto itself if we care about accurately capturing the heterogeneity in f_0, or counterfactual prediction. The mean squared error for estimating f_0 also translates directly into the size of the confidence interval for linear functional estimands. We show that 2SML with gradient boosted trees achieves out-of-sample R^2 improvements of around 0.1 and 0.15 in two novel IV benchmarks based on real-world datasets.

Second, we perform a coverage simulation for inference on the average derivative of f_0. Here we use a non-linear synthetic setup but with parameterizable dependence between the treatment and covariates that controls whether the parameter of interest is well-identified. In the well-identified case, our debiasing procedure improves coverage from 69.6% (without debiasing) to 94.4%. In a very poorly-identified setting, our debiased confidence interval undercovers slightly at 88.4%, as expected theoretically (Dorn, 2025), but this is an enormous improvement over the 3.6% coverage without debiasing.

6.1 Semi-Synthetic Evaluation of the Structural Function Estimate

NPIV methods are often benchmarked on purely synthetic data. These are typically low dimensional, and use hand-designed functional forms with limited heterogeneity — see for example

the commonly-used conference demand problem from Hartford et al. (2017). In these simple settings, there is often no advantage to using a complex model like gradient boosted tree ensembles compared to a sieve or kernel ridge model; the strength of tree ensembles lies in their excellent predictive performance on complicated real-world datasets. A few papers use high-dimensional designs with images as treatment (Bennett et al., 2019; Dikkala et al., 2020; Xu et al., 2020), but these designs are not representative of typical economics data in IV applications.

We design two semi-synthetic benchmarks using real data on taxi fares and house prices from Grinsztajn et al. (2022). These are so-called "tabular" datasets (where each observation has a mix of numerical and categorical attributes) typical of economics applications. Our basic approach is to take an existing prediction task, and split it into a training sample and an test sample. We choose one highly predictive attribute as the endogenous variable ("the treatment"), and then add correlated noise to both that variable and outcome in the training sample. Transformations of the original "treatment" variable (before the noise is added) serve as valid instruments. In this way, we can make sure that the assumptions underlying NPIV are met, without having to specify the relationship between treatment, covariates and outcomes. In practice, these relationships feature complicated heterogeneity that requires machine learning methods to model sufficiently. Our benchmark task is to fit an IV model in the endogenous training sample and predict on the unconfounded test sample — the prediction squared error on the test sample is equal to the L_2 error for the structural function up to a constant. We give describe how we construct the benchmarks in Appendix E.1. We provide a brief summary of our two settings in Table 3.

We compare 2SML using gradient-boosted trees in the first and second stages (GBoost 2SML) against several baselines: a naive ML estimator ignoring the instrument; classic linear two-stage least squares; Kernel IV from Singh et al. (2019); Ensemble IV, a minimax method proposed in Dikkala et al. (2020) using random forests; and an oracle estimator that gets direct access to the unconfounded outcomes. The results are summarized in Table 4. Gboost 2SML substantially outperforms the other methods. Notice that the performance of the minimax random forest method EnsembleIV is unstable, collapsing in the Census Housing task.

Table 3: Dataset Characteristics

Dataset	Train Samples	Test Samples	Outcome Variable	# of Features
NYC Green Cab	406,600	174,258	Log Taxi Fare	16
Census Housing	15,948	6,836	Log Median House Price	16

Table 4: Out-of-Sample R^2

Dataset	Naive ML	2SLS	Kernel IV	Ensemble IV	GBoost 2SML	Oracle
NYC Green Cab	-0.07	0.53	0.52	0.57	**0.72**	0.86
Census Housing	0.14	0.18	0.43	-0.01	**0.53**	0.80

Notes: Each estimator is fit in the training sample, and R^2 values are computed in the hold-out sample. Naive ML is a GBoost model fit ignoring endogeneity. The Oracle estimator is GBoost model fit with endogeneity removed, representing an upper bound on predictive performance of any IV estimator.

6.2 Simulation Results for Coverage

We perform purely-synthetic Monte Carlo simulations to assess coverage of our asymptotic normal confidence intervals using 2SML with and without debiasing. Our results demonstrate that plug-in IV models without bias correction can dramatically undercover, whereas our debiasing procedure recovers correct coverage. We design an average derivative estimation task with an endogenous non-linear outcome and with non-linear dependence between the treatment, instrument, and covariates. The strength of the dependence between treatment and covariates induces a non-parametric form of multi-collinearity, allowing us to vary the degree of identification for the average derivative parameter (similar to controlling overlap for the average treatment effect). We describe our data-generating process in Appendix E.2.

We run Monte Carlo simulations with $n = 2000$ repeated for 250 trials. We focus on two settings: (1) a setting with moderate dependence between treatment and covariates resulting in a challenging but well-identified estimation task, and (2) a setting with strong dependence between treatment and covariates such that the average derivative is very poorly-identified. This poorly-identified setting is intended to be a particularly challenging case where even a debiased estimator should fail to achieve perfect coverage — see Kang and Schafer (2007); Dorn (2025) and similar. We provide point estimates and 95% confidence intervals for two estimators: a 2SML model without debiasing; and a debiased estimate where both nuisances are fit with 2SML. We summarize the results in Table 5.

In the well-identified setting, the 2SML plug-in (without debiasing) significantly undercovers, with only 70% of the confidence intervals containing the true parameter. The debiased 2SML estimate restores correct coverage while simultaneously achieving a small improvement in RMSE for the target estimand. In the poorly-identified setting, the coverage of the 2SML plug-in collapses to 3.6%, while the debiased estimator achieves coverage of 88.4%, even in this especially challenging case. Note that here the RMSE for the plug-in and debiased estimates are roughly the same;

instead, the improved coverage for Debiased 2SML is achieved through a nearly 4x increase in the standard error, more accurately reflecting the true amount of uncertainty.

Table 5: Monte Carlo Results for Average Derivative Estimation Task

Method	Well-Identified Setting				Poorly-Identified Setting			
	Bias	Std. Err.	RMSE	Coverage	Bias	Std. Err.	RMSE	Coverage
Plug-in 2SML	-0.029	0.024	0.044	0.696	-0.094	0.022	0.098	0.036
Debiased 2SML	-0.010	0.036	0.038	0.944	-0.056	0.079	0.101	0.884

Notes: Metrics averaged over 250 trials with $n = 2000$. Coverage is for the 95% confidence interval. Dependence between treatments and covariates generates a non-linear form of multicollinearity. In the "Well-Identified Setting", this dependence is moderate, and in the "Poorly-Identified Setting", this dependence is strong.

7 Empirical Application: Demand Estimation

We now present an empirical application to demand estimation using the California supermarket data from Compiani (2022), where consumers choose between organic strawberries, non-organic strawberries, and an outside option (other fresh fruits). A key characteristic of this dataset is bunching at 9-ending price points, as illustrated in Figure 1. Of our [redacted] observations on organic strawberries, [redacted] have prices ending in 0.99, and [redacted] have a price of exactly [redacted] per pound. We will demonstrate that our non-parametric approach using tree ensembles is especially valuable in this setting, achieving a nearly 7x reduction in NPIV estimation error compared to the best prior method. Our model captures strong discontinuities at the dollar boundary, resulting in an estimated price elasticity of [redacted], between 2.5-6x larger than the estimates previously reported in Compiani (2022) and Chen et al. (2023) using the same dataset. The tendency of prices to bunch at 9-endings is widely-documented, so while our application is to strawberry demand, the takeaways should be more broadly applicable.

Figure 1: Counts of Price Per Pound for Organic Strawberries

We use our debiased 2SML procedure to estimate the price elasticity of demand for organic strawberries by estimating the average derivative following the specification in Chen et al. (2023). All observations are at the store-week level. $Y \in \mathbb{R}$ is the log market share for organic strawberries, $D \in \mathbb{R}^2$ are the log prices for organic and non-organic strawberries, and $X \in \mathbb{R}^4$ are covariates including: income, taste for organic products,[7] state-level sales of other fruit, and average price of other fruit. The instruments $Z \in \mathbb{R}^5$ include 3 Hausman IVs (average prices at stores not in the same marketing area), and spot prices for organic and non-organic strawberries. See the online Appendix of Compiani (2022) for a complete description of the construction of the dataset. The own-price elasticity of organic strawberries is $\mathbb{E}[\partial f_0(D_1, D_2, X)/\partial D_1]$.

Note that our primary goal is to match the setup of Chen et al. (2023) as closely as possibly, in order to directly compare the results for identical estimands. However, we recognize there are potential limitations of this setup from an applied point of view. For example, we require endogeneity to enter additively, which may be inconsistent with theoretical microfoundations (Berry and Haile, 2016), and we use Hausman instruments, which place strong independence assumptions on unobservables across markets. We leave further exploration of demand estimation in particular to future work. Finally, Chen et al. (2023) use a differentiable model and compute the derivative analytically. By contrast, our best fitting model using GBoost has strong discontinuities at the dollar boundary. Furthermore, the price distribution itself is partly discrete — most prices per pound are in round cents, like [redacted].[8] Therefore, we calculate the derivative with symmetric differencing: $\mathbb{E}[(f_0(D_1 + \epsilon, D_2, X) - f_0(D_1 - \epsilon, D_2, X))/2\epsilon]$ with $\epsilon = 0.01$ (large enough to be at least 1 cent for all observations). If we believe the true demand function is discontinuous, then this ϵ-shock response could itself be considered a valid estimand. We consider alternative specifications in the Appendix.

7.1 Estimation Error for GBoost 2SML vs Previous Methods

Two-Stage Machine Learning with tree ensembles provides a substantially better estimate of the structural function than existing methods. We demonstrate this using an out-of-sample measure of the NPIV estimation error. Recall that the structural function is the solution to the nested regression problem (4). Minimizing this objective over $f \in \mathcal{F}$ directly is challenging because for each candidate f, we have to estimate $\mathbb{E}[f(D)|Z = z]$. However, for any *given* candidate $\hat{f}$, estimating $\mathbb{E}[\hat{f}(D)|Z = z]$ is straightforward: we train a machine learning model to predict $\hat{f}(D)$ given Z.

[7] The percentage of total yearly sales of lettuce that are organic at the store.

[8] Although some are not, like [redacted].

Table 6: Comparison of Estimation Error Across NPIV Estimators

	Estimator	NPIV MSE	NPIV R^2	MSE vs $\mathbb{E}[Y\|Z]$
Our method	GBoost 2SML	[redacted]	[redacted]	[redacted]
Without trees	2SLS	[redacted]	[redacted]	[redacted]
	Sieve IV (Newey and Powell, 2003)	[redacted]	[redacted]	[redacted]
	Kernel IV (Singh et al., 2019)	[redacted]	[redacted]	[redacted]
	Deep IV (Hartford et al., 2017)	[redacted]	[redacted]	[redacted]
	Deep Feature IV (Xu et al., 2020)	[redacted]	[redacted]	[redacted]
With trees	GBoost/Spline (Chen et al., 2023)	[redacted]	[redacted]	[redacted]
	Ensemble IV (Dikkala et al., 2020)	[redacted]	[redacted]	[redacted]

Notes: "NPIV MSE" is the value of (17) averaged over test folds with cross-fitting. "NPIV R^2" is the corresponding R^2 value. The best achievable value for (17) by any NPIV method is the MSE of the reduced form, which is [redacted]. The column "MSE vs $\mathbb{E}[Y|Z]$" shows how close each method is to achieving that optimal value. The true structural function will achieve approximately 0.

Call this estimate, $\hat{T}\hat{f}$. We can then get an estimate of the objective from (4), by computing:

$$\sum_{i=1}^{n}(Y_i - (\hat{T}\hat{f})(Z_i))^2. \tag{17}$$

Because $\hat{T}\hat{f}$ is a function of Z, the best possible value of (17) achievable by any $\hat{f}$ is the MSE of $\mathbb{E}[Y|Z]$. Furthermore, because $\mathbb{E}[Y|Z] = \mathbb{E}[f_0(D)|Z]$, we know that this best possible MSE is achieved by the true f_0. This suggests a simple end-to-end procedure: we estimate $\hat{T}\hat{f}$ for a set of NPIV estimates, and then in a separate sample compare their MSEs to that of the best reduced form predictor. For a good estimate of the structural function, the difference should be nearly zero.

We perform this procedure with 4-fold cross-fitting on our dataset for organic strawberries. In each training fold, we compute estimates $\hat{f}$ for a variety of NPIV estimators including our 2SML method with GBoost. We compare our method against 2SLS, Sieve IV and Kernel IV, which are linear in a fixed feature transformation; Deep IV, which solves a conditional density estimation problem in the first stage; Deep Feature IV, an iterative method using deep neural networks; the spline minimium distance method from Chen et al. (2023), using a gradient-boosted tree ensemble to represent f_0; and Ensemble IV, an adversarial/minimax method using tree ensembles. For each $\hat{f}$, we estimate the corresponding $\hat{T}\hat{f}$ using machine learning (picking the best model by cross-validation within the training fold).[9] We then compute (17) in the test fold. We show the results averaged across the four folds in Table 6.

[9]If $\hat{f}$ was fit with Kernel IV for example, the model for $\mathbb{E}[\hat{f}(D)|Z]$ selected via cross-validation need not be a kernel ridge model. Indeed, we find for $\hat{f}$ fit using Sieve IV and Kernel IV, the best predictor for $\hat{f}(D)$ given Z in our demand dataset is a tree ensemble. This again emphasizes that the sieve/kernel bases are misspecified.

The key measure of success is the rightmost column, which compares the value of (17) to the best achievable value — the reduced form MSE of [redacted]. The true structural functional would achieve a value of approximately zero. Our GBoost 2SML estimator achieves a *nearly seven-fold reduction* in MSE vs $\mathbb{E}[Y|Z]$ compared to the next best method, the GBoost/Spline estimator following Chen et al. (2023). This corroborates our finding in Table 2 that a spline basis cannot sufficiently model $\mathbb{E}[f_0(D)|Z]$ in this dataset. The minimax estimator, Ensemble IV, from Dikkala et al. (2020) has over 8x the estimation error of our approach. Note that even though Ensemble IV uses tree ensembles for both the structural function and the instruments, it does not perform as well as the simpler two-stage GBoost/Spline approach, highlighting the limitations of adversarial approaches. Chen et al. (2023) report a similar finding. The remaining methods that do not use tree ensembles perform even worse. Sieve and Kernel IV have 10x higher estimation than our method. Notably, the neural network-based approaches Deep IV and Deep Feature IV perform particularly poorly.

7.2 GBoost 2SML Estimates a Much Higher Price Elasticity

Our debiased point estimate for the average price elasticity is [redacted]. This is roughly [redacted] larger than the elasticity estimate of -5.5 reported in Compiani (2022), and [redacted] larger than the elasticity estimates in Chen et al. (2023), which range from -2.2 to -3.4. In Table 7, we compare our debiased estimate with the plug-in estimates from the models in Table 6. Note that for smooth methods — like 2SLS, Sieve IV, and Kernel IV — the estimate of the price elasticity is around [redacted]. By contrast, the price elasticities estimated using tree ensembles are substantially higher. The GBoost/Spline and Ensemble IV estimates are around [redacted]: higher than the smooth estimates, but still smaller than our point estimate of [redacted], and falling below our 95% confidence interval. Note that the neural network approaches (which perform particularly poorly in Table 6) estimate an elasticity as low as [redacted].

We find that our point estimate is mainly driven by discontinuities at the dollar boundary, which gradient-boosted tree ensembles excel at modeling. In Figure 2, we plot how our 2SML estimate of the price elasticity varies with own-price. We find large negative price elasticities at 99-endings. For example, [redacted] of all our observations have a price of exactly [redacted]. Among just these observations, our estimated price elasticity is around [redacted]. This estimate is inline with existing studies on 99-ending prices. For example, Schindler and Kibarian (1996) find at a clothing retailer that 99-ending prices had 8% more sales volume than 00-ending prices. To compare magnitudes, note that at [redacted], a change to [redacted] is a [redacted] difference. Thus an 8% change in demand would correspond

Table 7: Estimated Own-Price Elasticities for Organic Strawberries

	Estimator	Price Elasticity	Debiased Price Elasticity
Our method	GBoost 2SML	[redacted]	[redacted]
Without trees	2SLS	[redacted]	
	Sieve IV	[redacted]	
	Kernel IV	[redacted]	
	Deep IV	[redacted]	
	Deep Feature IV	[redacted]	
With trees	GBoost/Spline	[redacted]	
	Ensemble IV	[redacted]	

Notes: Estimates are average price elasticities using 4-fold cross-fitting. The top of the table contains our main estimate with debiased standard errors clustered at the store level. The bottom of the table includes plug-in estimates across other NPIV methods.

to a price elasticity of [redacted].

Notice that in Figure 2 there is consistently a *positive* price elasticity leading up to the 99-endings. We estimate that changing prices from [redacted] actually increases demand. This corroborates the finding in Snir and Levy (2021) that shoppers perceive 9-ending prices as being lower than non-9-ending prices, even though this is not actually the case on average. This could be evidence for violations of profit maximization, but there are alternative explanations — for example, 99-endings may signal lower quality, there may be regulatory restrictions, or there may exist unobserved confounders that violate the spacial independence assumption of Hausman instruments.

7.3 Why is the 2SML Standard Error So Large?

Our debiased point estimate of [redacted] for the own-price elasticity comes with a fairly large standard error of [redacted]. The resulting 95% confidence interval is [redacted]. The large standard error is due to the debiasing step. By contrast, the naive standard error for the 2SML plug-in estimate of [redacted] is [redacted] (although this estimate is biased and so the standard error is not asymptotically valid).

The debiasing nuisance $\hat{q}(Z_i)$ directly models how sensitive the final point estimate is to each observation. In the bias correction term, the debiasing nuisance is then multiplied by the prediction residuals of the structural function: $\hat{q}(Z_i)(Y_i - \hat{f}(D_i))$. When $\hat{q}(Z_i)$ can take on extremely large values, then our variance and standard errors can drastically increase. We find that the averaged squared value of $\hat{q}(Z_i)$ is [redacted], and this is largely driven by observations right above 9-ending price points. For example, the average squared value of $\hat{q}(Z_i)$ for observations with price between

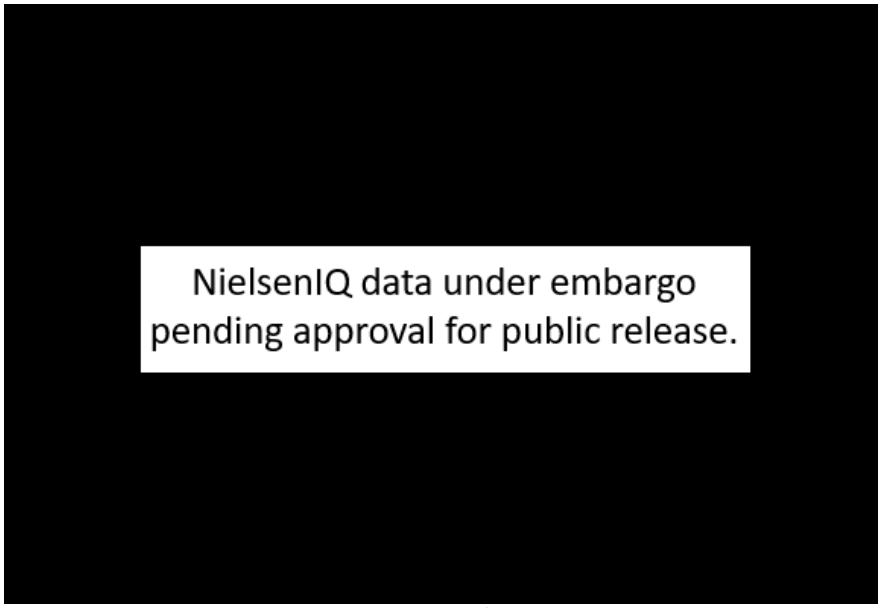

Figure 2: Own-Price Elasticity for Organic Strawberries

Notes: Average price elasticities within own-price bins for the structural function estimated using two stage machine learning with gradient-boosted trees. Bin size was chosen relative to $\epsilon = 0.01$ used for symmetric differencing. The x-axis values are the center of the bins.

Table 8: Elasticity Estimates Under Small Perturbations to Price

Noise Std.	Debiased 2SML Elasticity (s.e.)	Sieve IV Elasticity
None	[redacted]	[redacted]
1e-3	[redacted]	[redacted]
5e-3	[redacted]	[redacted]
1e-2	[redacted]	[redacted]

Notes: Each row corresponds to a reanalysis with small mean-zero Gaussian noise added to the log own-price. Noise Std." is the standard deviation of the noise. Since the noise is added to log prices, 1e-3 corresponds to a 0.1% change. The noise is small enough that smooth models like Sieve IV are nearly unaffected.

[redacted] is [redacted]. To understand what is going on, note that because a large percentage of the observations have prices bunched at price points with 9-endings like [redacted], the estimate of the price elasticity is highly sensitive to observations with prices just above these price points. However, there are [redacted] observations with prices in the range [redacted] — see Figure 1 — meaning that we have a small effective sample size for estimating the discontinuity at the dollar. This is not a mistake or limitation of our method, but a reflection of the underlying high uncertainty.

Finally, we perform a simple perturbation experiment that cleanly illustrates the extent to which our large point estimate and large standard errors are driven by discontinuities at price points. In this experiment, we add a small amount of noise to the log price of organic strawberries in the

training data, erasing information at the discontinuities. The estimand under noise can be interpreted as a smoothed version of the original estimand. The amount of noise is so small relative to the total variation in the data that classical attenuation bias is not a concern — in fact the estimate of the elasticity using Sieve IV barely changes over the levels of noise we consider. We collect the results in Table 8. As we add more noise to price, our debiased estimate of the price elasticity falls to around [redacted] — in line with the estimates from smooth models like Sieve IV and 2SLS from Table 7. The size of the standard error drops from [redacted] to [redacted]. The noise added to price improves identification (for the smoothed estimand) by shrinking the size of $\hat{q}(Z_i)$, which was previously driven by the discontinuity at the dollar. Intuitively, the smoothed version of the elasticity estimand is better-identified because there is plenty of global variation in prices.

8 Conclusion

In this paper, we introduced Two-Stage Machine Learning, a simple nonparametric IV procedure that uses machine learning models for both the structural function and the instruments. This contrasts with existing estimators, which either (1) impose a linearity assumption in a fixed sieve or kernel basis, or (2) solve computationally-intensive iterative optimization problems with convergence issues that limit their effectiveness. Our key insight is that we can use the reduced form to learn strong instruments for the structural function. Our first stage learns a basis of instruments by predicting Y given Z, X, and in the second stage we estimate the structural function by predicting Y given D, X, projected onto this basis. We prove finite-sample convergence guarantees for our estimated structural function, and develop a complimentary debiasing procedure that provides valid asymptotic normal inference for scaler summaries like an average elasticity.

We revisit a demand estimation application using California supermarket data that features extensive bunching at price points: [redacted] of all observations end in 0.99. We show that our procedure with tree ensembles achieves a nearly 7x reduction in NPIV estimation error compared to the best prior approach. In particular, tree methods excel at modeling discontinuities, and we find a strong response at the dollar boundary, resulting in an estimated average price elasticity of around [redacted]. This is between 2.5 to 6 times larger than estimates previously reported in the literature using this same dataset.

Our methodology presents opportunities to revisit classic two stage least squares applications using large administrative datasets and machine learning. A number of extensions to our approach

would be useful across applied settings. First, our debiasing approach applies to scalar estimands, but could be extended to debias conditional average treatment effects. Second, it would be interesting to adapt our method to settings with very strong time-series dynamics that are common in macroeconomics. Finally, we find that using the output of each tree in a gradient-boosting ensemble works especially well in our second stage, but there may be other ways to construct a basis from the ensemble. It would be interesting to develop a better theoretical characterization of the corresponding induced Hilbert spaces. We leave these directions for future work.

References

AI, C. AND X. CHEN (2003): "Efficient estimation of models with conditional moment restrictions containing unknown functions," *Econometrica*, 71, 1795–1843.

——— (2007): "Estimation of possibly misspecified semiparametric conditional moment restriction models with different conditioning variables," *Journal of Econometrics*, 141, 5–43.

——— (2012): "The semiparametric efficiency bound for models of sequential moment restrictions containing unknown functions," *Journal of Econometrics*, 170, 442–457.

ANDERSON, E. T. AND D. I. SIMESTER (2003): "Effects of $9 price endings on retail sales: Evidence from field experiments," *Quantitative marketing and Economics*, 1, 93–110.

BAKHITOV, E. AND A. SINGH (2022): "Causal gradient boosting: Boosted instrumental variable regression," in *Proceedings of the 23rd ACM Conference on Economics and Computation*, 604–605.

BARTLETT, P. L., O. BOUSQUET, AND S. MENDELSON (2005): "Local rademacher complexities," .

BARTLETT, P. L., P. M. LONG, G. LUGOSI, AND A. TSIGLER (2020): "Benign overfitting in linear regression," *Proceedings of the National Academy of Sciences*, 117, 30063–30070.

BENNETT, A., N. KALLUS, X. MAO, W. NEWEY, V. SYRGKANIS, AND M. UEHARA (2022): "Inference on strongly identified functionals of weakly identified functions," *arXiv preprint arXiv:2208.08291*.

——— (2023): "Minimax Instrumental Variable Regression and L_2 Convergence Guarantees without Identification or Closedness," in *The Thirty Sixth Annual Conference on Learning Theory*, PMLR, 2291–2318.

BENNETT, A., N. KALLUS, AND T. SCHNABEL (2019): "Deep generalized method of moments for instrumental variable analysis," *Advances in neural information processing systems*, 32.

BERRY, S. AND P. HAILE (2016): "Identification in differentiated products markets," *Annual review of Economics*, 8, 27–52.

BLUNDELL, R., L. PISTAFERRI, AND I. PRESTON (2008): "Consumption inequality and partial insurance," *American Economic Review*, 98, 1887–1921.

CARD, D. (1995): "Using Geographic Variation in College Proximity to Estimate the Return to Schooling," in *Aspects of Labour Market Behaviour: Essays in Honour of John Vanderkamp*, ed. by L. N. Christofides, E. K. Grant, and R. Swidinsky, Toronto: University of Toronto Press, 201–222.

——— (2001): "Estimating the return to schooling: Progress on some persistent econometric problems," *Econometrica*, 69, 1127–1160.

CHEN, J., D. L. CHEN, AND G. LEWIS (2020): "Mostly harmless machine learning: learning optimal instruments in linear IV models," *arXiv preprint arXiv:2011.06158*.

CHEN, J., X. CHEN, AND E. TAMER (2023): "Efficient estimation of average derivatives in NPIV models: Simulation comparisons of neural network estimators," *Journal of Econometrics*, 235, 1848–1875.

CHEN, X., Y. FAN, AND V. TSYRENNIKOV (2006): "Efficient estimation of semiparametric multivariate copula models," *Journal of the American Statistical Association*, 101, 1228–1240.

CHEN, X., Z. LIAO, AND Y. SUN (2014): "Sieve inference on possibly misspecified semi-nonparametric time series models," *Journal of Econometrics*, 178, 639–658.

CHEN, X. AND S. C. LUDVIGSON (2009): "Land of addicts? an empirical investigation of habit-based asset pricing models," *Journal of Applied Econometrics*, 24, 1057–1093.

CHEN, X. AND D. POUZO (2012): "Estimation of nonparametric conditional moment models with possibly nonsmooth generalized residuals," *Econometrica*, 80, 277–321.

——— (2015): "Sieve Wald and QLR inferences on semi/nonparametric conditional moment models," *Econometrica*, 83, 1013–1079.

CHERNOZHUKOV, V., D. CHETVERIKOV, M. DEMIRER, E. DUFLO, C. HANSEN, W. NEWEY, AND J. ROBINS (2018): "Double/debiased machine learning for treatment and structural parameters," .

CHERNOZHUKOV, V., W. NEWEY, V. M. QUINTAS-MARTINEZ, AND V. SYRGKANIS (2022a): "Riesznet and forestriesz: Automatic debiased machine learning with neural nets and random forests," in *International Conference on Machine Learning*, PMLR, 3901–3914.

CHERNOZHUKOV, V., W. K. NEWEY, V. QUINTAS-MARTINEZ, AND V. SYRGKANIS (2021): "Automatic debiased machine learning via Riesz regression," *arXiv preprint arXiv:2104.14737*.

CHERNOZHUKOV, V., W. K. NEWEY, AND R. SINGH (2022b): "Automatic debiased machine learning of causal and structural effects," *Econometrica*, 90, 967–1027.

——— (2023): "A simple and general debiased machine learning theorem with finite-sample guarantees," *Biometrika*, 110, 257–264.

COMPIANI, G. (2022): "Market counterfactuals and the specification of multiproduct demand: A

nonparametric approach," *Quantitative Economics*, 13, 545–591.

DELLAVIGNA, S. AND M. GENTZKOW (2019): "Uniform pricing in us retail chains," *The Quarterly Journal of Economics*, 134, 2011–2084.

DIKKALA, N., G. LEWIS, L. MACKEY, AND V. SYRGKANIS (2020): "Minimax estimation of conditional moment models," *Advances in Neural Information Processing Systems*, 33, 12248–12262.

DORN, J. (2025): "How Much Weak Overlap Can Doubly Robust T-Statistics Handle?" *arXiv preprint arXiv:2504.13273*.

DYNAN, K. E., J. SKINNER, AND S. P. ZELDES (2004): "Do the rich save more?" *Journal of political economy*, 112, 397–444.

FISCHER, S. AND I. STEINWART (2020): "Sobolev norm learning rates for regularized least-squares algorithms," *Journal of Machine Learning Research*, 21, 1–38.

FONSECA, Y., C. PEIXOTO, AND Y. SAPORITO (2024): "Nonparametric instrumental variable regression through stochastic approximate gradients," *Advances in Neural Information Processing Systems*, 37, 131756–131785.

FOSTER, D. J. AND V. SYRGKANIS (2023): "Orthogonal statistical learning," *The Annals of Statistics*, 51, 879–908.

GHASSAMI, A., A. YING, I. SHPITSER, AND E. T. TCHETGEN (2022): "Minimax kernel machine learning for a class of doubly robust functionals with application to proximal causal inference," in *International conference on artificial intelligence and statistics*, PMLR, 7210–7239.

GRINSZTAJN, L., E. OYALLON, AND G. VAROQUAUX (2022): "Why do tree-based models still outperform deep learning on typical tabular data?" *Advances in neural information processing systems*, 35, 507–520.

HARTFORD, J., G. LEWIS, K. LEYTON-BROWN, AND M. TADDY (2017): "Deep IV: A flexible approach for counterfactual prediction," in *International Conference on Machine Learning*, PMLR, 1414–1423.

HAUSMAN, J. A. (1983): "Specification and estimation of simultaneous equation models," *Handbook of econometrics*, 1, 391–448.

——— (1996): "Valuation of new goods under perfect and imperfect competition," in *The economics of new goods*, University of Chicago Press, 207–248.

HONG, M., H.-T. WAI, Z. WANG, AND Z. YANG (2023): "A two-timescale stochastic algorithm

framework for bilevel optimization: Complexity analysis and application to actor-critic," *SIAM Journal on Optimization*, 33, 147–180.

JI, W., L. LEI, AND A. SPECTOR (2023): "Model-agnostic covariate-assisted inference on partially identified causal effects," *arXiv preprint arXiv:2310.08115*.

KANG, J. D. AND J. L. SCHAFER (2007): "Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data," .

KIM, J., D. MEUNIER, A. GRETTON, T. SUZUKI, AND Z. LI (2025): "Optimality and Adaptivity of Deep Neural Features for Instrumental Variable Regression," *arXiv preprint arXiv:2501.04898*.

LANCKRIET, G. R., N. CRISTIANINI, P. BARTLETT, L. E. GHAOUI, AND M. I. JORDAN (2004): "Learning the kernel matrix with semidefinite programming," *Journal of Machine learning research*, 5, 27–72.

LEVY, D., A. SNIR, A. GOTLER, AND H. A. CHEN (2020): "Not all price endings are created equal: Price points and asymmetric price rigidity," *Journal of Monetary Economics*, 110, 33–49.

LI, Z., H. LAN, V. SYRGKANIS, M. WANG, AND M. UEHARA (2024): "Regularized deepiv with model selection," *arXiv preprint arXiv:2403.04236*.

LIAO, L., Y.-L. CHEN, Z. YANG, B. DAI, M. KOLAR, AND Z. WANG (2020): "Provably efficient neural estimation of structural equation models: An adversarial approach," *Advances in Neural Information Processing Systems*, 33, 8947–8958.

LUO, Y., M. SPINDLER, AND J. KÜCK (2025): "High-dimensional L2-boosting: Rate of Convergence," *Journal of Machine Learning Research*, 26, 1–54.

MUANDET, K., A. MEHRJOU, S. K. LEE, AND A. RAJ (2020): "Dual instrumental variable regression," *Advances in Neural Information Processing Systems*, 33, 2710–2721.

NEWEY, W. K. AND J. L. POWELL (2003): "Instrumental variable estimation of nonparametric models," *Econometrica*, 71, 1565–1578.

PETRULIONYTĖ, I., J. MAIRAL, AND M. ARBEL (2024): "Functional bilevel optimization for machine learning," *Advances in Neural Information Processing Systems*, 37, 14016–14065.

SCHINDLER, R. M. AND T. M. KIBARIAN (1996): "Increased consumer sales response though use of 99-ending prices," *Journal of Retailing*, 72, 187–199.

SCHÖLKOPF, B. AND A. J. SMOLA (2002): *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press.

SEVERINI, T. A. AND G. TRIPATHI (2012): "Efficiency bounds for estimating linear functionals of nonparametric regression models with endogenous regressors," *Journal of Econometrics*, 170, 491–498.

SINGH, R. (2024): "Kernel Ridge Riesz Representers: Generalization, Mis-specification, and the Counterfactual Effective Dimension," *arXiv preprint arXiv:2102.11076v4*.

SINGH, R., M. SAHANI, AND A. GRETTON (2019): "Kernel instrumental variable regression," *Advances in Neural Information Processing Systems*, 32.

SNIR, A. AND D. LEVY (2021): "If you think 9-ending prices are low, think again," *Journal of the Association for Consumer Research*, 6, 33–47.

STOCK, J. H. AND M. W. WATSON (2018): "Identification and estimation of dynamic causal effects in macroeconomics using external instruments," *The Economic Journal*, 128, 917–948.

STRAUB, L. (2019): "Consumption, savings, and the distribution of permanent income," *Unpublished manuscript, Harvard University*, 17.

SYRGKANIS, V. AND M. ZAMPETAKIS (2020): "Estimation and inference with trees and forests in high dimensions," in *Conference on learning theory*, PMLR, 3453–3454.

WAINWRIGHT, M. J. (2019): *High-dimensional statistics: A non-asymptotic viewpoint*, vol. 48, Cambridge university press.

XU, L., Y. CHEN, S. SRINIVASAN, N. DE FREITAS, A. DOUCET, AND A. GRETTON (2020): "Learning deep features in instrumental variable regression," *arXiv preprint arXiv:2010.07154*.

A Computing Projections in an RKHS

The 2SML algorithm involves computing the projection $P_\phi f$, where $f \in \mathbb{R}^n$ is the vector with elements $f(D_i)$ for $i \in \{1, ..., n\}$, and where

$$P_\phi = \Phi(\Phi^\top \Phi + \lambda I)^+ \Phi^\top.$$

When d_ϕ is greater than n, including infinite-dimensional ϕ, then we can still efficiently compute P_ϕ using the kernel matrix. Consider an RKHS $\mathcal{H}_k$ with PSD kernel $k : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ and corresponding features $\phi(Z) \in \mathcal{H}_k$. Let K be the $n \times n$ matrix with entries $k(z_i, z_j)$. Then for $\lambda > 0$:

$$P_\phi = K(K + \lambda I)^+$$

When $\lambda = 0$, K need not be invertible, but we can instead adopt the minimum-norm solution by replacing the inverse with the pseudoinverse as in e.g. Bartlett et al. (2020).

B Counterfactual Feature Derivation

Following the derivation for (5), let $\mathcal{T}_\varphi$ denote the operator that maps $q \in L_2(Z)$ onto the best approximation of the conditional expectation $\mathbb{E}[q(Z)|D]$ that is linear in φ. In other words $(\mathcal{T}_\varphi q)(d) = \varphi(d)^\top \beta(q)$, where

$$\beta(q) := \operatorname*{argmin}_{\beta \in \mathbb{R}^{d_\varphi}} \mathbb{E}[(q(Z) - \phi(D)^\top \beta)^2].$$

The projected Riesz loss in the population is:

$$\begin{aligned}
&\min_{q \in \mathcal{Q}} \mathbb{E}[(\mathcal{T}_\varphi q)(D)^2 - 2m(\mathcal{T}_\varphi q, D)] \\
&= \min_{q \in \mathcal{Q}} \mathbb{E}[(\varphi(D)^\top \beta(q))^2 - 2m(\varphi(\cdot)^\top \beta(q), D)] \\
&= \min_{q \in \mathcal{Q}} \mathbb{E}[(\varphi(D)^\top \beta(q))^2 - 2m(\varphi, D)^\top \beta(q)],
\end{aligned}$$

where in the last line we've used the fact that $\mathbb{E}[m(f, D)]$ is a linear operator on $f \in L_2(D)$.

For the sample optimization problem, recall the shorthand definitions from the main text: let $\varphi \in \mathbb{R}^{n \times d_\varphi}$ be the matrix with rows $\varphi(D_i)$, and let $\varphi_{\text{cf}} \in \mathbb{R}^{n \times d_\varphi}$, be the matrix with rows equal to $m(\varphi, D_i)$. Let $\bar{\varphi}_{\text{cf}} \in \mathbb{R}^{d_\varphi}$ be the sample average of φ_{cf}. Write q for the vector with entries $q(Z_i)$.

Then we have:

$$\hat{\beta}(q) = (\varphi^\top \varphi + \lambda I)\varphi^\top,$$

and so the sample optimization problem is:

$$\min_{q \in \mathcal{Q}} \left\{ \frac{1}{n} q^\top P_\varphi q - 2P_{\text{cf}} q \right\}. \tag{18}$$

C Details for Theoretical Results

C.1 Empirical Risk Minimization Lemma

The following lemma is adapted from Bartlett et al. (2005); Foster and Syrgkanis (2023). Define the star hull of a set $\mathcal{G}$ as $\text{star}(\mathcal{G}) := \{\alpha g | g \in \mathcal{G}, \alpha \in [0,1]\}$.

Lemma 1. *Let $\mathcal{A}$ and $\mathcal{B}$ be function classes with $\mathcal{A} \subseteq \mathcal{B} \subseteq L_2(Z)$ such that $sup_{b \in \mathcal{B}} \|b\|_\infty < \infty$. Let $\hat{a} \in \mathcal{A}$ be any function satisfying $L_n(\hat{a}) = \inf_{a \in \mathcal{A}} L_n(a)$ and let $a^* \in \mathcal{A}$ be any function satisfying $L(a^*) = \inf_{a \in \mathcal{A}} L(a)$. Given Assumption 1 and assuming that $\ell(g; z)$ is C-Lipschitz with respect to its first argument over $\mathcal{B}$, for any δ_n satisfying the inequality:*

$$\frac{\mathcal{R}_n(star(\mathcal{B} - a^*), \delta)}{\delta} \leq \delta,$$

then with probability at least $1 - \eta$, with constants that depends only on C and B:

$$\|\hat{a} - a^*\|_Z \leq O\left(\delta_n + \sqrt{\frac{\log(1/\eta)}{n}}\right).$$

Proof. We have

$$\begin{aligned}
\|\hat{a} - a^*\|_2^2 &\leq B(L(\hat{a}) - L(a^*)) \\
&\leq B((L(\hat{a}) - L(a^*)) - (L_n(\hat{a}) - L_n(a^*))),
\end{aligned}$$

where the last line follows because $L_n(\hat{a}) - L_n(a^*) \leq 0$.

Define: $\delta = \delta_n + \sqrt{\log(1/\eta)/n}$. Then applying Lemma 12 of Foster and Syrgkanis (2023), with

probability at least $1 - \eta$:

$$(L(\hat{a}) - L(a^*)) - (L_n(\hat{a}) - L_n(a^*)) \leq O(\delta \|\hat{a} - a^*\|_2 + \delta^2),$$

where the constants depend only on C. Finally, applying the AM-GM inequality with δ and $\|\hat{a} - a^*\|_2$, we get:

$$\|\hat{a} - a^*\|_2^2 \leq O(\delta^2).$$

□

C.2 Proof of Theorem 1

Proof. Define the projected function class,

$$\mathcal{G}_\phi(\mathcal{F}) \coloneqq \{\hat{\mathcal{T}}_\phi f : f \in \mathcal{F}\} \subseteq \mathcal{H}_k^b.$$

and define

$$g_{f^*} \in \operatorname*{argmin}_{g \in \mathcal{G}_{\hat{\phi}}(\mathcal{F})} L(g),$$

where for some $f^* \in \mathcal{F}$, we have $g_{f^*} = \hat{\mathcal{T}}_{\hat{\phi}} f^*$. The function g_{f^*} is guaranteed to exist: since $\mathcal{F}$ is closed and $\hat{\mathcal{T}}_{\hat{\phi}}$ is continuous, $\mathcal{G}_{\hat{\phi}}(\mathcal{F})$ is closed, and since $\mathcal{G}_{\hat{\phi}}(\mathcal{F}) \subseteq \mathcal{H}_k^b$, it is also compact.

In what follows, we'll write $x \lesssim y$ when $x \leq Cy$ for a constant C that possibly depends on A or B. The value of the constant may differ from line to line. Recall that $\Delta_{\hat{\phi}} \coloneqq L(\mathcal{T}\hat{f}) - L(\hat{\mathcal{T}}_{\hat{\phi}}\hat{f})$. Then we have:

$$\begin{aligned}
\|\mathcal{T}(f_0 - \hat{f})\|_Z &= \|g_0 - \mathcal{T}\hat{f}\|_Z \\
&\lesssim L(\mathcal{T}\hat{f}) - L(g_0) \\
&= L(\hat{\mathcal{T}}_{\hat{\phi}}\hat{f}) - L(g_0) + L(\mathcal{T}\hat{f}) - L(g_0) - L(\hat{\mathcal{T}}_{\hat{\phi}}\hat{f}) + L(g_0) \\
&= L(\hat{\mathcal{T}}_{\hat{\phi}}\hat{f}) - L(g_0) + \Delta_{\hat{\phi}} \\
&\lesssim \|g_0 - \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}\|_Z + \Delta_{\hat{\phi}}.
\end{aligned}$$

The inequality on the second line follows from convexity of $L(g)$ and because g_0 is the minimum of $L(g)$ over $g \in L_2(Z)$. The inequality on the fifth line is similar, but applying smoothness.

Next by the triangle inequality we have that:

$$\|g_0 - \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}\|_Z \leq \|\hat{\mathcal{T}}_{\hat{\phi}}f^* - \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}\|_Z + \|g_0 - \hat{\mathcal{T}}_{\hat{\phi}}f^*\|_Z$$

and we can bound:

$$\begin{aligned}
\|g_0 - \hat{\mathcal{T}}_{\hat{\phi}}f^*\|_Z^2 &\lesssim L(\hat{\mathcal{T}}_{\hat{\phi}}f^*) - L(g_0) \\
&\leq L(\hat{\mathcal{T}}_{\hat{\phi}}f_0) - L(g_0) \\
&\lesssim \|g_0 - \hat{\mathcal{T}}_{\hat{\phi}}f_0\|_Z^2,
\end{aligned}$$

where the first line follows from convexity of L, the second line because $\hat{\mathcal{T}}_{\hat{\phi}}f^*$ is the minimum of $L(g)$ over $g \in \mathcal{G}_{\hat{\phi}}(\mathcal{F})$ and $f_0 \in \mathcal{F}$, and the fifth line by smoothness of L. Putting this together we have:

$$\begin{aligned}
\|\mathcal{T}(f_0 - \hat{f})\|_Z &\lesssim \|\hat{\mathcal{T}}_{\hat{\phi}}f^* - \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}\|_Z + \|g_0 - \hat{\mathcal{T}}_{\hat{\phi}}f_0\|_Z + \Delta_{\hat{\phi}} \\
&\leq \|\hat{\mathcal{T}}_{\hat{\phi}}f^* - \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}\|_Z + \|\mathcal{T}_{\hat{\phi}}f_0 - \hat{\mathcal{T}}_{\hat{\phi}}f_0\|_Z + \|g_0 - \mathcal{T}_{\hat{\phi}}f_0\|_Z + \Delta_{\hat{\phi}} \\
&\leq \underbrace{\|\hat{\mathcal{T}}_{\hat{\phi}}f^* - \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}\|_Z}_{\text{term 1}} + \underbrace{\|\mathcal{T}_{\hat{\phi}}f_0 - \hat{\mathcal{T}}_{\hat{\phi}}f_0\|_Z}_{\text{term 2}} + \underbrace{\|g_0 - \hat{g}\|_Z}_{\text{first stage error}} + \Delta_{\hat{\phi}}.
\end{aligned}$$

The last line uses the fact that $\mathcal{T}_{\hat{\phi}}f_0$ is the $L_2(Z)$ projection of g_0 onto $\mathcal{H}_{\hat{k}}$, and $\hat{g} \in \mathcal{H}_{\hat{k}}$.

We complete the proof by bounding terms 1 and 2 using the empirical risk minimization result from Lemma 1.

Bounding term 1: Term 1 is the excess risk of the second stage. By construction, there will exist $g_{\hat{f}} \in \mathcal{G}_{\hat{\phi}}(\mathcal{F})$ such that $g_{\hat{f}} = \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}$ and

$$g_{\hat{f}} \in \operatorname*{argmin}_{g \in \mathcal{G}_{\hat{\phi}}(\mathcal{F})} L_n(g).$$

Therefore, $g_{\hat{f}}$ is the empirical risk minimizer of $L(g)$ over $g \in \mathcal{G}_{\hat{\phi}}(\mathcal{F})$ and g_{f^*} is the population risk minimizer of $L(g)$ over $g \in \mathcal{G}_{\hat{\phi}}(\mathcal{F})$. Furthermore, by construction $\mathcal{G}_{\hat{\phi}}(\mathcal{F}) \subseteq \mathcal{H}_{\hat{k}}^b$. Therefore, we can apply Lemma 1 with $\mathcal{B} = \mathcal{H}_{\hat{k}}^b$: with probability at least $1 - \eta$,

$$\|\hat{\mathcal{T}}_{\hat{\phi}}f^* - \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}\|_Z = \|g_{f^*} - g_{\hat{f}}\|_Z \leq O\left(\delta_n^b + \sqrt{\frac{\log(1/\eta)}{n}}\right).$$

Bounding term 2: Term 2 is the excess risk of empirical risk minimization over $\mathcal{H}_{\hat{k}}^b$ using the squared loss for predicting $f_0(D)$, instead of the risk $L(g)$. This can be bounded using any off-the-shelf result for kernel ridge regression — see for example, Fischer and Steinwart (2020); Singh (2024). We provide a bound based on the critical radius. The squared loss satisfies the conditions in Assumption 1 — quadratic functions are smooth and convex, and because $f_0(D)$ is uniformly-bounded and $\mathcal{H}_b^k$ is uniformly-bounded, then the loss is a uniformly bounded quadratic and therefore Lipschitz.

Therefore, we can apply Lemma 1 using $\mathcal{H}_{\hat{k}}^b$ to get with probability at least $1-\eta$:

$$\|\mathcal{T}_{\hat{\phi}} f_0 - \hat{\mathcal{T}}_{\hat{\phi}} f_0\|_Z \leq O\left(\delta_n^b + \sqrt{\frac{\log(1/\eta)}{n}}\right).$$

□

Proof of Proposition 3

Proof. The result follows by applying the triangle inequality and then Lemma 1. □

C.3 Approximation Bias in Projection Step

We now discuss relaxing the following requirement from Assumption 4: The radius b is sufficiently large such that $\mathcal{T}_{\hat{\phi}} f_0 \in \mathcal{H}_{\hat{k}}^b$.

Define the population projection onto the ball of radius b:

$$\mathcal{T}_{\hat{\phi}}^b f \coloneqq \underset{g \in \mathcal{H}_{\hat{k}}^b}{\operatorname{argmin}} \{\mathbb{E}[(f(D) - g(Z))^2]\}.$$

Then the condition on b is equivalent to assuming $\mathcal{T}_{\hat{\phi}} f_0 = \mathcal{T}_{\hat{\phi}}^b f_0$. In the proof of Theorem 1, we use this condition while bounding term 2. Without this condition, the bound has an additional approximation term:

Applying Lemma 1 using $\mathcal{H}_{\hat{k}}^b$ to get with probability at least $1-\eta$:

$$\|\mathcal{T}_{\hat{\phi}} f_0 - \hat{\mathcal{T}}_{\hat{\phi}} f_0\|_Z \leq O\left(\|\mathcal{T}_{\hat{\phi}} f_0 - \mathcal{T}_{\hat{\phi}}^b f_0\|_Z + \delta_n^b + \sqrt{\frac{\log(1/\eta)}{n}}\right).$$

As b increases, we have a trade-off. The approximation error, $\|\mathcal{T}_{\hat{\phi}} f_0 - \hat{\mathcal{T}}_{\hat{\phi}} f_0\|_Z$ shrinks, but the critical radius, δ_n^b will grow. We can choose b to balance these two terms in such a way that asymptotically the approximation goes to zero, but in finite samples we control the variance term δ_n^b. For an example of such a hyperparameter schedule and resulting rates under minimal additional smoothness conditions, see Theorem 1 of Fischer and Steinwart (2020). The rates in Fischer and Steinwart (2020) take the form $\sqrt{\log(n)/n}$ but raised to a power depending on the smoothness of $\mathcal{T}_{\hat{\phi}} f_0$.

C.4 Testing Assumption 5

Assumption 5 requires that $\Delta_{\hat{\phi}}(\hat{f}) \leq 0$, where

$$\begin{aligned}\Delta_{\hat{\phi}}(\hat{f}) &\coloneqq L(\mathcal{T}\hat{f}) - L(\hat{\mathcal{T}}_{\hat{\phi}}\hat{f}) \\ &= \mathbb{E}\Big[\Big(Y - \mathbb{E}[\hat{f}(D)|Z]\Big)^2\Big] - \mathbb{E}\Big[\Big(Y - (\hat{\mathcal{T}}_{\hat{\phi}}\hat{f})(Z)\Big)^2\Big]\end{aligned}$$

Recall that $\mathcal{T}_{\hat{\phi}}\hat{f}$, defined in Equation (15), is an approximation of $\mathbb{E}[\hat{f}(D)|Z]$ using ridge regression of Y on $\hat{\phi}(Z)$. We know that for the true structural function we have $\Delta_{\hat{\phi}}(f_0) = 0$ as $n \to 0$. This follows because $\mathbb{E}[f_0(D)|Z] = \mathbb{E}[Y|Z]$, and we constructed $\hat{\phi}(Z)$ such that $\mathbb{E}[Y|Z]$ is linear in $\hat{\phi}(Z)$, and so $\hat{\mathcal{T}}_{\hat{\phi}} f_0 \to \mathbb{E}[f_0(D)|Z]$.

We propose a simple, feasible test for Assumption 5. We divide the data into training and test samples (or use cross-fitting). In the training set, we obtain our 2SML estimate $\hat{f}$. Then also in the training set, we fit a machine learning predictor of $\hat{f}(D)$ given Z. We do model selection using cross-validation within the training sample. This produces an estimate of $\mathcal{T}\hat{f}$, call it $\hat{\mathcal{T}}\hat{f}$. This is the same procedure we use in Equation (17). Then in the test sample, we compute the empirical difference-in-means:

$$\frac{1}{n}\sum_{i=1}^{n}\left[\ell(\hat{\mathcal{T}}\hat{f}, Z_i) - \ell(\hat{\mathcal{T}}_{\hat{\phi}}\hat{f}, Z_i)\right]. \tag{19}$$

To test for violations of Assumption 5, we run a permutation test to check if this difference-in-means is statistically-significantly larger than zero. A sufficient but not necessary condition for Equation (19) to equal 0 is that ridge regression in $\phi(Z)$ is the best predictor of $\hat{f}(D)$ given Z, in which case $\hat{\mathcal{T}}\hat{f} = \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}$.

C.4.1 Summary of our findings in the data

We test for this condition in a number of real and synthetic datasets. We summarize the results here and show the tests across datasets in Appendix C.4.2. Our main finding is that Assumption 5 depends on how we construct $\hat{\phi}$. If we fit the reduced form using gradient-boosted trees and take $\hat{\phi}(Z)$ to be the output of each individual tree, we find that the test virtually never rejects. In fact, we usually have that ridge regression on $\hat{\phi}(Z)$ is the best predictor of $\hat{f}(D)$ and therefore $\hat{\mathcal{T}}\hat{f} = \hat{\mathcal{T}}_{\hat{\phi}}\hat{f}$. However, if we make $\hat{\phi}$ excessively simple, then the test can often reject. For example, if the final reduced form prediction of the gradient-boosted tree ensemble is $\hat{g}(Z)$, we could take $\hat{\phi}(Z) = \hat{g}(Z)$. Then we have a 1-dimensional representation such that $\mathbb{E}[Y|Z]$ is approximately linear in $\hat{\phi}(Z)$. When we fit 2SML using this basis, we find that our permutation often rejects, and the performance of the resulting $\hat{f}$ can be very poor. Similarly, if we choose $\hat{\phi}$ using the lasso, if the resulting basis is too sparse, the permutation test can reject. Note that the sparse lasso basis is lower-dimensional than the original covaraites, whereas the gradient-boosted tree basis is often orders of magnitude higher-dimensional than the original covariates.

C.4.2 Tests for Assumption 5 across datasets

All permutation tests use 10,000 permutations. The resulting p-value is an estimate, and an upper 95% confidence interval on an estimated 0 with this many permutations is 0.00037. For each plot in this section, we plot a histogram of the permutation for the difference in means in (19). We use a vertical black line to denote the actual difference in means. Assumption 5 says that the population version of this quantity should be less than or equal to 0.

Organic Strawberry Data; 2SML with Gboost Basis:

First, we test for Assumption 5 in our main specification for our empirical application, as described in Section 7. The reduced form is fit using gradient-boosted trees. The resulting $\phi(Z)$ is 3000-dimensional. We show the result of our permutation test in Figure 3.

Figure 3: Organic Strawberry Data; 2SML with GBoost Basis

The estimated p-value is 0.948, so there good reason to believe that Assumption 5 holds for our main specification. The actual difference in means is -0.007, which is orders of magnitude smaller than the squared means themselves — the MSE of $\hat{\mathcal{T}}\hat{f}$ (which is our measure of the out-of-sample NPIV estimation error) is [redacted].

Organic Strawberry Data; 2SML with 1d Prediction Basis:

Technically, the 1-dimensional basis $\hat{\phi}(z) = \hat{g}(z)$ satisfies the condition in Proposition 1. However, when we use this basis for in our empirical application, this basis violates Assumption 5 as we can see in Figure 4.

Figure 4: Organic Strawberry Data; 2SML with 1d Prediction Basis

The estimated p-value is 0.000, so the test very strongly rejects. To understand the absolute magnitude, the mean squared error of $\hat{\mathcal{T}}_{\hat{\phi}}\hat{f}$ is [redacted], which is essentially equal to 2SML with the full Gboost basis. However, the MSE of $\hat{\mathcal{T}}\hat{f}$ (which is our measure of the out-of-sample NPIV estimation error) increases all the way to [redacted]. This is a severe degredation in performance. Note that [redacted] is still smaller MSE than the MSE for the minimax method, EnsembleIV, in Table 6 which is [redacted]. However the MSE is now worse than the GBoost/Spline estimator following Chen et al. (2023), which achieves an MSE of [redacted].

Card 1995; 2SML with Gboost Basis:

We now test for Assumption 5 using the returns to schooling dataset from Card (1995). The outcome is log wage, treatment is years of schooling, we use proximity to 4-year and 2-year colleges as instruments. We use 5 controls: experience, and indicators for black, southern, SMSA, and marriage. Our cross-validated gboost model for the reduced form results in $\phi(Z)$ that is 322-dimensional. We show the result of our permutation test in Figure 5.

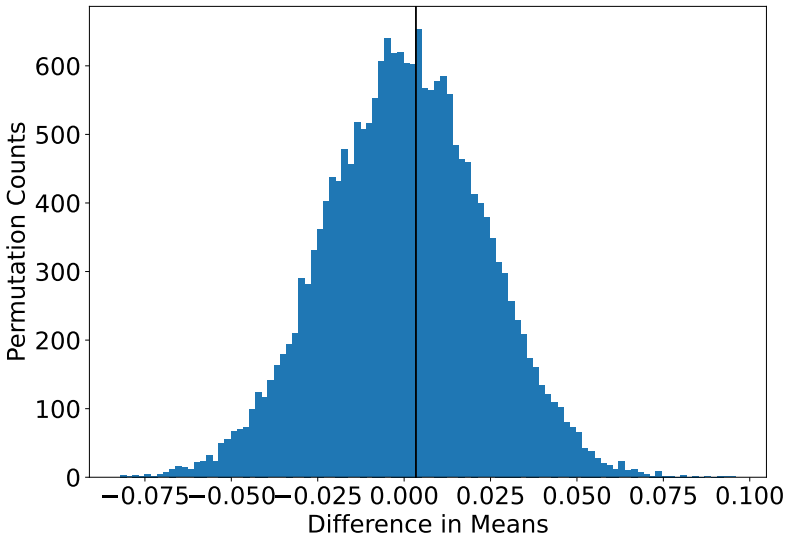

Figure 5: Card 1995; 2SML with GBoost Basis

The estimated p-value is 0.449, so the test does not reject. The actual difference in means is 0.003 which is orders of magnitude smaller than the squared means themselves — the MSE of $\hat{\mathcal{T}}\hat{f}$ is 0.741.

Census Housing Dataset; 2SML with GBoost Basis:

Next, we test Assumption 5 on our semi-synthetic Census housing dataset as introduced in Section 6. We use a cross-validated GBoost model for the reduced form, resulting in a $\phi(Z)$ that is 150-dimensional. The results of the permutation test are in Figure 6

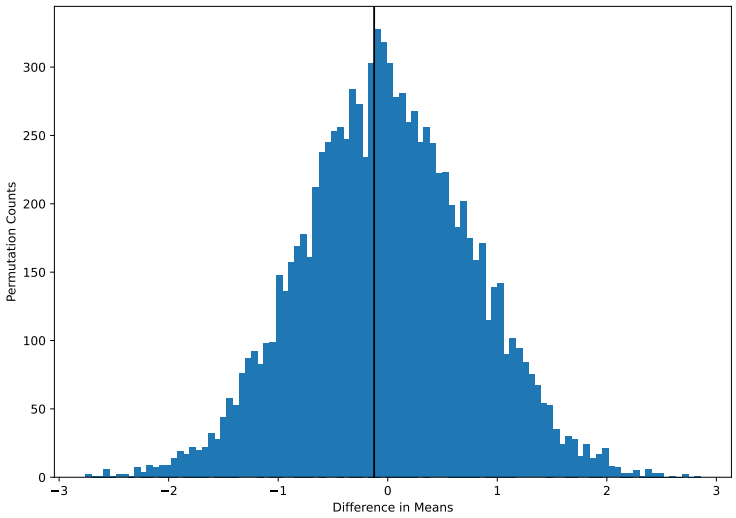

Figure 6: Census Housing; 2SML with GBoost Basis

The estimated p-value is 0.560 so the test does not reject. As in the cases above using the GBoost basis, the actual difference means (-0.124) is two orders of magnitude smaller than the MSE of $\hat{\mathcal{T}}\hat{f}$ (28.4).

NYC Green Cab Dataset; 2SML with GBoost Basis:

Finally, we test Assumption 5 on our semi-synthetic NYC Green Cab dataset as introduced in Section 6. We use a cross-validated GBoost model for the reduced form, resulting in a $\phi(Z)$ that is 200-dimensional. The results of the permutation test are in Figure 7.

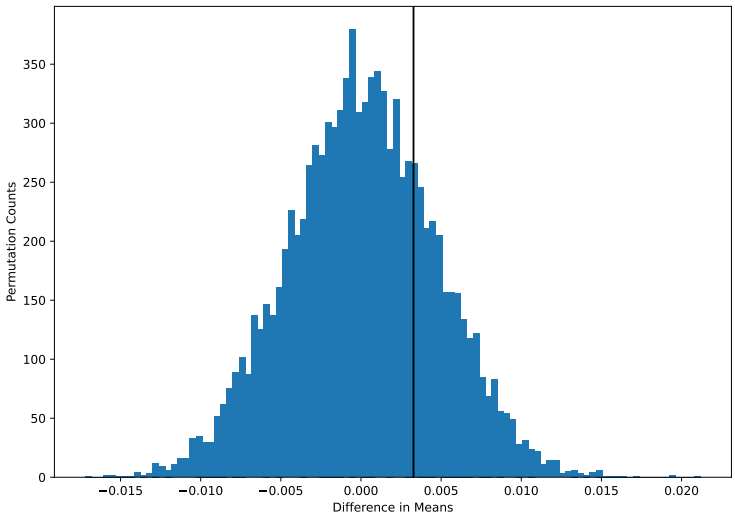

Figure 7: NYC Green Cab; 2SML with GBoost Basis

The estimated p-value is 0.247. This is not significant at the 90% level, but is a smaller p-value than in the other cases where we use a GBoost basis. However, the actual difference means is miniscule: 0.003. This is very small compared to the MSE of $\hat{\mathcal{T}}\hat{f}$ which is 0.883. So if there is a violation Assumption 5 on this dataset, it has an negligible impact on the final performance of the model.

C.5 Moment Conditions for Asymptotic Normality

Assumption 6 (Moment Conditions)**.** *Define* $\sigma^2 := \mathbb{E}[\psi_0(D,Z,Y)^2], \kappa^3 := \mathbb{E}[|\psi_0(D,Z,Y)|^3], \zeta^4 := \mathbb{E}[\psi_0(D,Z,Y)^4]$. *Then the following moment bounds hold for some* $(\bar{Q}, \bar{\sigma}, \bar{q}, \bar{q}')$*:*

1. $\mathbb{E}[m(f;D)^2] \leq \bar{Q}\|f\|_D^2$,
2. $\mathbb{E}[Y - f_0(D)|D] \leq \bar{\sigma}^2$,
3. $\|q_0\|_\infty \leq \bar{q}$, $\|\hat{q}\|_\infty \leq \bar{q}'$,
4. $\{(\kappa/\sigma)^3 + \zeta^2\}n^{-1/2} \to 0$.

Parts of Assumption 6 are already satisfied by applying Assumption 3 to 2SML and 2SRR:

For example, we've assumed that $f_0 \in \mathcal{F}_{\text{ml}}$ and $\sup_{f\in\mathcal{F}_{\text{ml}}}\|f\|_\infty < \infty$ by Assumption 3 applied to 2SML. Thus it's sufficient that Y is bounded almost surely to satisfy Assumption 6 (2). Note that in applying Assumption 3 to 2SML with the squared loss, we have assumed that the squared loss

is Lipschitz. This already implies that Y must be bounded almost surely.

Similarly, we've already assumed that $q_0 \in \mathcal{F}_{\text{rr}}$ and $\sup_{f \in \mathcal{F}_{\text{rr}}} \|f\|_\infty < \infty$ by Assumption 3 applied to 2SRR. This implies that $\|q_0\|_\infty < \infty$ and $\|\hat{q}\|_\infty < \infty$, satisfying Assumption 6 (3).

The other two parts of Assumption 6 require further assumptions placed on our estimand:

Assumption 6 (1) is a *mean-squared continuity* assumption on the linear functional $\theta(f)$. Note that this is stronger than continuity (the conditional required for Riesz representation). Continuity implies there exists $C < \infty$ such that:

$$\mathbb{E}[m(f; D)]^2 \leq C\|f\|_D^2,$$

which is equivalent to the existence of the Riesz representer α_0 that satisfies for some $\bar{M} < \infty$ (which is the operator norm of θ):

$$\mathbb{E}[\alpha_0(D)^2] \leq \bar{M}^2.$$

By contrast, the mean-squared continuity assumption,

$$\mathbb{E}[m(f; D)^2] \leq \bar{Q}\|f\|_D^2,$$

is a sufficient (but not necessary) condition for continuity with $\bar{M}^2 \leq \bar{Q}$.

Finally, Assumption 6 (4) is a condition on the moments of the efficient influence function. Section B.3 of Chernozhukov et al. (2023) provides a set of conditions such that these moments are bounded to be roughly on the same order as the operator norm, $\bar{M}$ or $\bar{M}^2$. Note that $\bar{M}$ is the usual measure of *overlap* of the functional θ, so this boils down to assuming that the degree of overlap puts some constraints on the 3rd and 4th moments of the efficient influence function.

D Cross-Fit Debiased Estimate

Here we describe how to compute the debiased estimate with sample splitting.

1. Randomly partition the n samples into folds, I_k, $k \in \{1, ..., K\}$.
2. For each fold k, estimate the nuisances $\hat{f}_k$ and $\hat{q}_k$ using all data *not* in I_k.

3. Compute the cross-fit debiased point estimate:

$$\hat{\theta}_D \coloneqq \frac{1}{n}\sum_{k=1}^{K}\sum_{i\in I_k} m(\hat{f}_k, D_i) + \hat{q}_k(Z_i)(Y_i - \hat{f}_k(D_i)).$$

4. Estimate the asymptotic variance:

$$\hat{\sigma}^2 \coloneqq \frac{1}{n}\sum_{k=1}^{K}\sum_{i\in I_k} \Big(\hat{\theta}_D - m(\hat{f}_k, D_i) - \hat{q}_k(Z_i)(Y_i - \hat{f}_k(D_i))\Big)^2.$$

E Details for Synthetic and Semi-Synthetic Evaluation

E.1 Forming IV Benchmarks from Prediction Tasks

We begin with a prediction dataset composed of $(X_i^{\text{pred}}, Y_i^{\text{pred}})$ pairs. The goal is produce a semi-synthetic dataset with random variables Y, D, W, Z that follows the DGP, $Y = f_0(D, W) + \epsilon$, where ϵ satisfies $\mathbb{E}[\epsilon|Z, W] = 0$, but such that ϵ is correlated with both Y and D. This turns the original prediction task into an endogenous prediction task that requires leveraging the instrument Z.

First, we fit a machine learning model (selected using cross-validation) that predicts Y^{pred} using X^{pred}. Next we choose a single dimension from X^{pred} to be the "treatment" variable; call this D^{pred} and the remaining features W^{pred}. We do so by computing feature importance measures from the machine learning model, and select the feature with the highest importance. The importance measures are different for different models. For a linear model, we use the size of the coefficients; for kernel ridge models, we use the average derivative with respect to the features; and for gradient boosted trees we use the "gain" feature importances computed by the xgboost library.

Next, we generate an unobserved confounding variable, U, that we use to construct both D and Y. For every observation in the original dataset, we draw U_i iid from a fixed distribution. We use Poisson in our benchmarks when D is integer-valued. Next we form a confounded version of the treatment. For each observation in the original dataset, we form:

$$D_i = D_i^{\text{pred}} + U_i.$$

Then we construct the confounded outcomes. Let the covariates be the same as in the original data, i.e. $W_i = W_i^{\text{pred}}$, Define the exogenous noise $\nu_i = Y_i^{\text{pred}} - f_0(D_i^{\text{pred}}, W_i)$, which by construction is approximately mean zero conditional on D_i^{pred}, W_i. Define a function of the confounder, $\rho(U)$,

such that $\mathbb{E}[\rho(U)] = 0$. Then our confounded outcomes are:

$$Y_i = f_0(D_i, W_i) + \rho(U_i) + \nu_i.$$

Notice that this matches the form of the NPIV problem, with $\epsilon_i = \rho(U_i) + \nu_i$. Naively predicting Y_i with D_i, W_i will result in a highly biased estimate of f_0 because D_i and ϵ_i are correlated.

Finally, we generate an instrument. For some function h (which may have vector valued output), let

$$Z_i = h(D_i^{\text{pred}}).$$

By construction, Z_i is independent of U_i and ν_i, but is predictive of D_i and is possibly correlated with W_i. Thus we satisfy the requirement that $\mathbb{E}[\epsilon|Z, W] = 0$. The relevance of Z_i depends on the functional form of h and the variance of the noise U_i. For simplicity, in our benchmarks we let h be the identity map, so the relevance of Z_i is driven by the variance of U_i.

Our final semi-synthetic dataset is composed of observations (Y_i, D_i, Z_i, W_i), one for each observation in the original dataset. For benchmarking, we divide this dataset into a training set and a hold-out for evaluation. We run different IV procedures in the training set to produce estimates of the structural function $\hat{f}$, and then evaluate them in the hold-out sample using mean-squared error and R^2 against the true structural function f_0.

This end-to-end procedure can be used to turn *any* prediction task into a endogenous prediction task with valid instruments. The following is a summary of the free parameters within this framework that can be varied to produce different benchmarks:

- The choice of the "treatment" feature D^{pred} from the columns of X^{pred}
- The distribution of the confounder, U
- The functional form of the outcome confounding, $\rho(U)$
- The functional form for the instruments, $h(D_i^{\text{pred}})$

Note that the resulting benchmark maintains a variety of the complex correlations that exist in the real data. The structural function f_0 will be the optimal predictor for the original task, which may require very complicated tree ensemble models. The treatment D_i will inherit the real-world correlations with W_i (all we have done is add independent noise to the original feature D_i^{pred}), without us having to specify these correlations in advance. Similarly for the correlation structure

between Z_i and W_i. The main synthetic component is the relationship between Z_i and D_i, and in the examples below, we use simple functional forms for this relationship. However, in NPIV for any candidate function $f(D, W)$, the relevant object, $\mathbb{E}[f(D, W)|Z, W]$, will in general be a complicated non-linear function of Z, making the choice of the functional form h less important.

E.2 Coverage Simulation DGP

Our DGP first draws 3 covariates, X_1, X_2, X_3 and an unconfounded treatment $\tilde{D}$. The variables $\tilde{D}, X_1, X_2$ are draw iid from the standard normal distribution. We set:

$$X_3 = 4 \cdot \text{expit}(\tilde{D} - X_1) - 2 + \epsilon_{\text{mcol}}$$

where $\epsilon_{\text{mcol}} \sim \mathcal{N}(0, \sigma^2_{\text{mcol}})$. We have a confounder $U \sim \mathcal{N}(0, \sigma^2_{\text{c}})$, confounded treatment, $D = \tilde{D} + U$, and instrument $Z = \tilde{D} + \epsilon_{\text{iv}}$ with $\epsilon_{\text{iv}} = \mathcal{N}(0, \sigma^2_{\text{iv}})$. The structural function is:

$$f_0(D, X) = D \cdot (0.2 + \sin(D) + \text{expit}(X_1) - 0.2 \cdot X_3),$$

with estimand:

$$\theta_0 = \theta(f_0) = \mathbb{E}\left[\frac{\partial f_0(D, X)}{\partial D}\right].$$

The outcome is $Y = f_0(D, X) + \rho \cdot U + \epsilon_{\text{out}}$ with $\epsilon_{\text{out}} \sim \mathcal{N}(0, \sigma^2_{\text{out}})$ and where $\rho \in \mathbb{R}$ controls the strength of confounding.

The key feature of our setting, is that as $\sigma^2_{\text{mcol}} + \sigma^2_{\text{iv}} \to 0$, X_3 becomes a *deterministic* function of D and X_1. This induces a nonparametric version of multi-collinearity. In the nonparametric model when $\sigma^2_{\text{mcol}} + \sigma^2_{\text{iv}} \to 0$, the average derivative w.r.t. D becomes unidentified, because any relationship between the outcome and D could be alternatively written as a relationship between the outcome and X_1, X_3. In other words, if we make σ^2_{mcol} very small, the operator norm of $\theta(f)$, or alternatively the norm of the Riesz representer α_0 can become very large — in causal inference language, we have poor overlap.

For all simulations, we use the following parameters: $\sigma_{\text{iv}} = 0.06, \sigma_{\text{c}} = 0.08, \sigma_{\text{out}} = 0.04, \rho = -8$. We then consider a medium overlap setting with $\sigma_{\text{mcol}} = 0.4$ and a poor overlap setting with $\sigma_{\text{mcol}} = 0.05$. Finally, we approximate the derivative numerically using symmetric differencing,

$$\theta(f) \approx \mathbb{E}\left[\frac{f(D + h, X) - f(D - h, X)}{2h}\right],$$

where we pick $h = 0.1$. For both settings, we get a ground-truth estimate of $\theta_0 = 0.7$ by simulating ten million samples and then calculating the derivative w.r.t. the true structural function.

F NPIV With and Without Covariates

For many NPIV methods, while it is without loss of generality to write $D = (D, X)$ and $Z = (Z, X)$, the case with covariates represents a major gap in difficulty. See Appendix C of Xu et al. (2020) for a discussion. Fortunately, in 2SML we do not encounter this difficulty. Our first-stage representation learning step finds a basis $\phi(Z, X)$ that best predicts Y. In the second stage, the projection P_ϕ kills information about X that is not relevant for predicting Y. We know that this extraneous information in X is not used in $\mathbb{E}[f_0(D, X)|Z, X]$ because of the NPIV moment condition. This is another advantage of our formulation.