Automatic debiasing of neural networks via moment-constrained learning

Christian L. Hines
The Alan Turing Institute
London, UK
chines@turing.ac.uk

Oliver J. Hines
QuantCo
Berlin, Germany
oliver.hines@quantco.com

Abstract

Causal and nonparametric estimands in economics and biostatistics can often be viewed as the mean of a linear functional applied to an unknown outcome regression function. Naively learning the regression function and taking a sample mean of the target functional results in biased estimators, and a rich debiasing literature has developed where one additionally learns the so-called Riesz representer (RR) of the target estimand (targeted learning, double ML, automatic debiasing etc.). Learning the RR via its derived functional form can be challenging, e.g. due to extreme inverse probability weights or the need to learn conditional density functions. Such challenges have motivated recent advances in automatic debiasing (AD), where the RR is instead learned directly via minimization of a bespoke loss. In this paper we propose moment-constrained learning as an alternative RR learning approach that addresses some shortcomings in AD, constraining the predicted moments and improving the robustness of RR estimates to optimization hyperparamters. Though our approach is not tied to a particular class of learner, we illustrate it using neural networks, and evaluate on the problems of average treatment/derivative effect estimation using semi-synthetic data. Our numerical experiments show improved performance versus state of the art benchmarks.

1 Introduction

Several problems in causal inference, economics, and biostatistics can be viewed as inferring the average moment estimand $\Psi \equiv \mathbb{E}[m(\mu, W)]$, where $W = (Y, Z)$ consists of an outcome Y and inputs $Z = (A, X)$, with $\mu(z) \equiv \mathbb{E}[Y|Z = z]$, and $m(f, W)$ is a known functional that is linear in f. Examples of this setup include average treatment, policy and derivative effects, which are outlined in more detail below. It is well known that for such estimands, naively plugging in regression estimates $\hat{\mu}$ and taking the sample mean of $m(\hat{\mu}, W)$, given i.i.d. observations of W, generally leads to biased estimates which converge to Ψ at less than the parametric $\sqrt{n}$ rate.

These biases arise because the bias-variance trade off of the regression estimator is controlled by a generic loss (e.g. mean squared error, cross-entropy) that does not adequately control for biases in the downstream estimation task. In particular, the true regression function μ satisfies $\mathbb{E}[\alpha(Z)\{Y - \mu(Z)\}] = 0$ for any function α, but the same is not true of the empirical mean $\mathbb{E}_n[\alpha(Z)\{Y - \hat{\mu}(Z)\}]$, which may converge to zero slower than the $\sqrt{n}$ rate. Biases for average moment estimands take this form for an estimand-specific function α. Specifically, the 'plug-in bias' is characterized by the Riesz representer (RR) α, which is an unknown function such that $\Psi = \langle \mu, \alpha \rangle$, where $\langle f, g \rangle \equiv \mathbb{E}[f(Z)g(Z)]$ denotes an inner product over a Hilbert space $\mathcal{H}$ equipped with norm $||f|| \equiv \langle f, f \rangle^{1/2}$ and it is assumed that $\mu \in \mathcal{H}$. Existence of a unique $\alpha \in \mathcal{H}$ follows by Riesz's representation theorem since $f \mapsto h(f) \equiv \mathbb{E}[m(f, W)]$ is a bounded linear map.

Preprint. Under review.

Example 1: Average treatment effect (ATE). *The ATE [36] is obtained by letting $A \in \{0, 1\}$ be a binary treatment, and $\Psi = \mathbb{E}[\mu(1, X) - \mu(0, X)]$. Letting $p(x) \equiv \mathbb{E}[A|X = x]$, and assuming $p(x) \in (0, 1)$, the ATE has the RR $\alpha(z) = \{a - p(x)\}/[p(x)\{1 - p(x)\}]$.*

Example 2: Average policy effect (APE) *Using the setup from Example 1, the APE [15, 44, 3] is $\Psi = \mathbb{E}[\pi(X)\{\mu(1, X) - \mu(0, X)\} + \mu(0, X)]$, where $x \mapsto \pi(x) \in \{0, 1\}$ is a known policy. The APE has the RR $\alpha(z) = [\pi(x)\{a - p(x)\} + p(x)\{1 - a\}]/[p(x)\{1 - p(x)\}]$.*

Example 3: Average derivative effect (ADE) *Assuming requisite derivatives exist, the ADE [19, 33, 25, 37] is $\Psi = \mathbb{E}[\mu'(A, X)]$ where A is a continuous treatment and μ' denotes the derivative of μ w.r.t. a. Letting $p(a|x)$ denote the conditional density of A given X, and assuming $p(a|x) > 0$, the ADE has the RR $\alpha(z) = p'(a|x)/p(a|x)$ where $p'(a|x)$ is the derivative of $p(a|x)$ w.r.t. a.*

Example 4: Incremental policy effect (IPE) *Using the setup from Example 3, the IPE [3] is $\Psi = \mathbb{E}[\pi(X)\mu'(A, X)]$, where $x \mapsto \pi(x) \in [-1, 1]$ is a known policy. The IPE has the RR $\alpha(z) = \pi(x)p'(a|x)/p(a|x)$.*

Following semiparametric efficiency results [35, 32], a rich literature has developed in recent years that compensates for plug-in biases either by estimating the RR then shifting the naive estimator (double machine learning [9]), or retrospectively modifying the estimates $\hat{\mu}$ such that the estimated plug-in bias is negligible (targeted learning [48]). Both approaches are celebrated for delivering efficient estimators that converge at $\sqrt{n}$ rate even when learners for the conditional mean outcome and the RR converge at a slower e.g. $n^{1/4}$ rate, as may be the case when machine learning (ML) methods are used.

As exemplified above, however, the RR can be a complicated function of the data distribution, making learning the RR using its derived form challenging. Consider the RR of the ATE and APE, which can be estimated after learning the propensity score p. Since p appears in the denominator of the RR, the resulting estimates may be overly sensitive to the error $\hat{p}(x) - p(x)$ when $\hat{p}(x)$ is close to 0 or 1. Similarly, RR estimators for the ADE typically use kernel estimators (with a differentiable kernel) and are overly sensitive to the choice of bandwidth [7].

To overcome such issues, recent work has sought to learn the RR directly from the data, without using knowledge of its functional form. Initial approaches for binary treatments used balancing weights rather than propensity scores to estimate plug-in biases [49, 2]. These approaches have been generalized through the adversarial RR learner of [13] that builds on the (also adversarial) augmented minimax linear estimator [23] and similar estimators for conditional moment models [14]. More recently, automatic debiasing (AD) [10, 11] has been proposed to bypass the need to solve a computationally challenging adversarial learning problem by constructing a simple loss that is equivalent to minimizing the mean squared error in the RR. AD generalizes similar approaches using approximately sparse linear regression [12] and reproducing kernel Hilbert spaces [41].

Despite the success of AD, there are several areas for improvement which we address in our work. First, the AD loss is unbounded, and includes a negative average moment term that can lead to extreme moment predictions in the final RR estimator. In practice, early stopping using an external validation set is recommend to avoid such issues, however the resulting learners may be overly sensitive to e.g. early stopping and learning rate hyperparameters. Second, oftentimes the RR admits known inner products which are ignored by the AD loss. For instance, we know *a priori* that for the ADE/ATE $h(a) = \mathbb{E}[A\alpha(Z)] = 1$. Methods which estimate the RR using its derived functional form approximately encode such identities, but this is not the case when the RR is learned by AD.

Contributions: We propose average moment estimators based on a new decomposition of the RR in terms of the moment-constrained function $\beta_\perp(z)$. Specifically, $\beta_\perp$ minimizes the mean squared error in predicting a known function $\beta(z)$ subject to $h(\beta_\perp) = 0$, where β is chosen such that one knows *a priori* that $h(\beta) \neq 0$. Furthermore we propose an approach to learning $\beta_\perp$ and debiased estimators of Ψ based on initial ML estimates $\hat{\mu}$ and $\hat{\beta}_\perp$.

The advantage of learning the RR via $\beta_\perp$ rather than the AD loss is that the resulting RR estimates better control for extreme out-of-sample RR predictions since constants of proportionality in the RR are estimated using the estimation sample rather than the training sample. Moreover, our proposed estimator for $\beta_\perp$ is robust to overfitting issues that may arise when using the AD loss and thus is less sensitive to the tuning of optimization/model hyperparameters. Our proposal remains 'automatic' in the sense of not requiring the functional form of the RR to be derived. However, unlike AD, which only requires knowledge of the moment function m, we require users to construct a known function β with $h(\beta) \neq 0$. Constructing such functions is easy as we demonstrate for Examples 1 to 4 above.

Using multi-tasking neural networks, we evaluate our estimators on two semi-synthetic datasets, comparing them with RieszNet [11] for ADE/ATE estimation, and DragonNet [40], Reproducing Kernel Hilbert Space (RKHS) Embedding [41], Neural Net (NN) Embedding [46] for ATE estimation. To ensure a fair comparison we re-implement RieszNet and DragonNet learners and estimators, with code available at `https://github.com/anonymized`. Our repository includes all reproduction code and proofs are provided in the supplementary material.

2 Estimation

2.1 Debiased estimation

Given a sample of n i.i.d. observations, estimators of Ψ are typically based on initial estimates of the regression function $\hat{\mu}$, the RR $\hat{\alpha}$, and the empirical distribution $\mathbb{E}_n[.] = n^{-1}\sum_{i=1}^{n}(.)_i$. Letting $G_n[.] \equiv \sqrt{n}(\mathbb{E}_n[.] - \mathbb{E}[.])$ be an empirical process operator, an estimator ψ of Ψ is said to be regular asymptotically linear (RAL) if $\sqrt{n}(\psi - \Psi) = G_n[\varphi(W)] + o_p(1)$ for some finite variance function $\varphi(W)$. RAL estimators are unbiased since, by the central limit theorem, $\sqrt{n}(\psi - \Psi)$ converges to a mean zero normal distribution with variance $\text{var}[\varphi^2(W)]$. Moreover, results from nonparametric efficiency theory [32] imply that $\text{var}[\varphi^2(W)]$ is minimized when $\varphi(W) = m(\mu, W) + \alpha(Z)\{Y - \mu(Z)\}$ is the uncentered influence curve [18, 24] of Ψ, also called the pseudo-outcome [27, 21] (see [22, 26] for pedagogical reviews). Thus, one can construct standard errors for ψ by approximating φ with some $\hat{\varphi}$ and taking a sample variance. To consider specific estimators ψ we use the identity

$$\sqrt{n}(\psi - \Psi) = G_n\left[\varphi(W)\right] - \underbrace{\sqrt{n}\mathbb{E}_n\left[\hat{\varphi}(W) - \psi\right]}_{\text{plug-in bias}} + \underbrace{\sqrt{n}\mathbb{E}[\hat{\varphi}(W) - \Psi]}_{\text{first-order remainder}} - \underbrace{G_n\left[\varphi(W) - \hat{\varphi}(W)\right]}_{\text{second-order remainder}} \quad (1)$$

which holds for any ψ and pair of (measurable) functions $\varphi, \hat{\varphi}$. The second-order remainder above is usually not a concern and is $o_p(1)$ under weak assumptions, e.g. when $\mathbb{E}[\{\varphi(W) - \hat{\varphi}(W)\}^2] = o_p(1)$ and $\hat{\varphi}$ is obtained from an independent sample. In practice, this motivates estimators which apply some form of sample-splitting/cross-fitting to estimate $\hat{\varphi}$ and evaluate the estimator [48, 9].

Letting $h_n(f) \equiv \mathbb{E}_n[m(f, W)]$, a naive (Direct) estimator is $\hat{\Psi} \equiv h_n(\hat{\mu})$, which does not use the RR estimates $\hat{\alpha}$. To examine the bias properties of the naive estimator, consider (1) when $\psi = \hat{\Psi}$ and $\hat{\varphi}(W) = m(\hat{\mu}, W) + \hat{\alpha}(Z)\{Y - \hat{\mu}(Z)\}$. Following [13], the first-order remainder reduces to the 'mixed bias', $-\sqrt{n}\langle\hat{\mu} - \mu, \hat{\alpha} - \alpha\rangle$, the square of which can be bound by Cauchy-Schwarz as $n\langle\hat{\mu} - \mu, \hat{\alpha} - \alpha\rangle^2 \leq n||\hat{\mu} - \mu||^2||\hat{\alpha} - \alpha||^2$. The first-order remainder will therefore be $o_p(1)$ when $\hat{\mu}$ or $\hat{\alpha}$ converge to their true counterparts at sufficiently fast rates with sample size. Moreover, one can trade off accuracy in $\hat{\mu}$ and $\hat{\alpha}$, a property known as rate double robustness.

The plug-in bias $\sqrt{n}\mathbb{E}_n[\hat{\varphi}(W) - \hat{\Psi}] = \sqrt{n}\mathbb{E}_n[\hat{\alpha}(Z)\{Y - \hat{\mu}(Z)\}]$, however, is generally not $o_p(1)$, and hence $\hat{\Psi}$ is not RAL. The main challenge in obtaining RAL estimators is therefore removing plug-in biases and there are three main strategies for doing so. One-step debiased estimators simply add the plug-in bias to both sides of (1), resulting in the (double robust) RAL estimator, $\hat{\Psi}^{(\text{DR})} \equiv \psi + \mathbb{E}_n[\hat{\varphi}(W) - \psi] = \mathbb{E}_n[\hat{\varphi}(W)]$. Estimating equations estimators obtain an estimator as the solution ψ to $\mathbb{E}_n[\hat{\varphi}(W) - \psi] = 0$, which is solved by $\psi = \hat{\Psi}^{(\text{DR})}$ since $\hat{\varphi}(W)$ does not depend on ψ.

Finally, targeted maximum likelihood estimators (TMLEs) are similar, in a sense, to estimating equations estimators, but instead replace $\hat{\mu}$ with an alternative $\hat{\mu}^*$. These estimators return $\hat{\Psi}^{(\text{TMLE})} = h_n(\hat{\mu}^*)$ where $\hat{\mu}^*$ solves the score equation

$$\mathbb{E}_n[\hat{\alpha}(Z)\{Y - \hat{\mu}^*(Z)\}] = 0. \quad (2)$$

TMLEs of this form are also double robust, with the mixed bias condition $\langle\hat{\mu}^* - \mu, \hat{\alpha} - \alpha\rangle = o_p(n^{-1/2})$. Targeting can be achieved in many ways, for example by first defining the linear parametric submodel $\hat{\mu}_t(z) = \hat{\mu}(z) + t\hat{\alpha}(z)$ where $t \in \mathbb{R}$ is a univariate indexing parameter. An optimal t^* can then be obtained by minimizing $\mathbb{E}_n[\{Y - \hat{\mu}_t(Z)\}^2]$ over t, thereby improving the fit of the

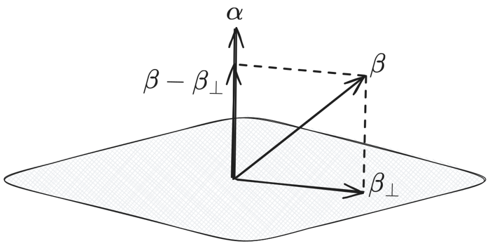

Figure 1: Illustration of moment-constrained function. The plane represents the space of zero average moment functions, i.e. f such that $h(f) = \langle f, \alpha \rangle = 0$. The non-zero function $\beta - \beta_\perp$ is orthogonal to the plane, hence it is a scalar multiple of the Riesz representer α.

outcome learner in the estimation sample, and ensuring that $\hat{\mu}^* = \hat{\mu}_{t^*}$ is a solution to (2). For the linear parametric submodel above, the TMLE $\hat{\Psi}^{(\text{TMLE})} = h_n(\hat{\mu}_{t^*})$ reduces to

$$\hat{\Psi}^{(\text{TMLE})} = h_n(\hat{\mu}) + \left(\frac{h_n(\hat{\alpha})}{\mathbb{E}_n[\hat{\alpha}^2(Z)]}\right)\mathbb{E}_n[\hat{\alpha}(Z)\{Y - \hat{\mu}(Z)\}]. \tag{3}$$

Comparing $\hat{\Psi}^{(\text{TMLE})}$ with $\hat{\Psi}^{(\text{DR})}$, we see that the TMLE introduces a correction term $h_n(\hat{\alpha})/\mathbb{E}_n[\hat{\alpha}^2(Z)]$, which is an empirical approximation to the population value $h(\alpha)/||\alpha||^2 = 1$. Variations of the TMLE method often include canonical GLM link functions in the parametric submodel definition, and maximize the associated GLM log-likelihood (hence the name TMLE), see e.g. [43] for submodel proposals. GLM variations of this type may be used e.g. when Y is binary and a cross-entropy outcome loss is preferred. Finally, to motivate new RR learning methodologies, we remark that $\hat{\Psi}^{(\text{TMLE})}$, and its score equation in (2), are invariant to constants of proportionality in $\hat{\alpha}$, thus one might consider RR learners that are agnostic to such constants.

2.2 Debiased estimation with moment constraints

Our main contribution is to propose average moment estimators based on the identity

$$\alpha(z) = \frac{h(\beta)}{||\beta - \beta_\perp||^2}\{\beta(z) - \beta_\perp(z)\} \tag{4}$$

where $\beta \in \mathcal{H}$ is a known function with $h(\beta) \neq 0$ and $\perp$ denotes projection on to orthogonal complement set $\mathcal{C}^\perp \equiv \{f \in \mathcal{H} \mid h(f) = 0\}$. Specifically, $\beta_\perp = \arg\min_{f \in \mathcal{C}^\perp} ||\beta - f||$. A geometric illustration of this result is provided in Figure 1.

Proof of (4): *Note that $\mathcal{C}^\perp = \{f \in \mathcal{H} \mid \langle f, \alpha \rangle = 0\}$. By Hilbert's projection theorem, $\beta_\perp$ exists, with*

$$\beta_\perp(z) \equiv \beta(z) - \frac{\langle \beta, \alpha \rangle}{||\alpha||^2}\alpha(z) \qquad \Longleftrightarrow \qquad \alpha(z) = \frac{||\alpha||^2}{h(\beta)}\{\beta(z) - \beta_\perp(z)\}$$

where we use $\langle \beta, \alpha \rangle = h(\beta)$. Taking the norm of both sides and solving for $||\alpha|| \neq 0$ gives $||\alpha|| = |h(\beta)|/||\beta - \beta_\perp||$ which completes the proof.

The identity in (4) offers new avenues for debiased estimation of Ψ via learning $\beta_\perp$ and μ. In practice, there are usually candidates for β, where $h(\beta)$ is known *a priori*. For instance, for the ATE and ADE, $h(\beta) = 1$ when $\beta(z) = a$. For the APE, $h(\beta) = 1$ when $\beta(z) = a + 1 - \pi(x)$. For the IPE, $h(\beta) = 1$ when $\beta(z) = a/\pi(x)$, or if there is concern that $\pi(X)$ can be zero, then one can let $\beta(z) = a$, and estimate $h(\beta) = \mathbb{E}[\pi(X)]$. For full generality, we develop estimators for the setting where $h(\beta)$ must be estimated, but our results simplify slightly when $h(\beta)$ is known.

Example ATE: *Denoting $\beta(z) = \beta(a, x) = a$, the known RR and* (4) *imply*

$$\beta_\perp(a, x) = p(x) + \{a - p(x)\}\left(1 - \frac{1}{p(x)\{1 - p(x)\}}\mathbb{E}\left[\frac{1}{p(X)\{1 - p(X)\}}\right]^{-1}\right).$$

It is insightful to compare $\beta_\perp$, which minimizes $\mathbb{E}[\{A-f(A,X)\}^2]$ given $\mathbb{E}[f(1,X)] = \mathbb{E}[f(0,X)]$, with $p(x)$, which minimizes the same mean squared error, under the stronger constraint $f(1,X) = f(0,X)$. We notice that $p(x)$ lies on the interval $(0,1)$, but the same is not true of $\beta_\perp$, which has weaker restrictions on its outputs: $\beta_\perp(1,x) < 1$ and $\beta_\perp(0,x) > 0$. Also p and $\beta_\perp$ are related by the identity $p(x) = \mathbb{E}[\beta_\perp(A,X)|X = x]$.

In Section 2.3 below, we propose methods for learning $\beta_\perp$, however, we first show how the standard debiased estimators from Section 2.1 look when debiasing is achieved using initial estimates $\hat{\beta}_\perp$ instead of $\hat{\alpha}$. Specifically, we consider

$$\hat{\alpha}(z) = \frac{h_n(\beta - \hat{\beta}_\perp)\{\beta(z) - \hat{\beta}_\perp(z)\}}{\mathbb{E}_n[\{\beta(Z) - \hat{\beta}_\perp(Z)\}^2]}. \tag{5}$$

Under this parameterization, the one-step debiased (and estimating equations) estimator becomes

$$\hat{\Psi}^{(\mathrm{DR})} = h_n(\hat{\mu}) + \frac{h_n(\beta - \hat{\beta}_\perp)\mathbb{E}_n[\{\beta(Z) - \hat{\beta}_\perp(Z)\}\{Y - \hat{\mu}(Z)\}]}{\mathbb{E}_n[\{\beta(Z) - \hat{\beta}_\perp(Z)\}^2]}.$$

Since the the TMLE score equation in (2) only requires estimating the RR up to constants of proportionality, TMLEs can be derived using parametric submodels where $\hat{\alpha}$ is replaced with $\beta - \hat{\beta}_\perp$, e.g. by using the linear submodel $\hat{\mu}_t(z) = \hat{\mu}(z) + t\{\beta(z) - \hat{\beta}_\perp(z)\}$. For the linear parametric submodel, the TMLE recovers the one-step debiased estimator $\hat{\Psi}^{(\mathrm{TMLE})} = \hat{\Psi}^{(\mathrm{DR})}$. This equality arises because constants of proportionality in the RR are estimated using the estimation sample, rather than through an RR learner. This is already the case for TMLEs, but is introduced to one-step debiased estimators through our choice of RR learner.

Finally, the estimators in Section 2.1 rely on the mixed bias assumption $\langle \hat{\mu} - \mu, \hat{\alpha} - \alpha \rangle = o_p(n^{-1/2})$ to control the first-order remainder. In the moment-constrained learning setting, this assumption can be replaced by $\langle \hat{\mu}^* - \mu, \hat{\beta}_\perp - \beta_\perp \rangle = o_p(n^{-1/2})$, where $\hat{\mu}^*$ is the conditional mean estimator which is used to construct $\hat{\Psi}^{(\mathrm{TMLE})} = h(\hat{\mu}^*)$.

2.3 Moment-constrained learning

We propose learners for the moment-constrained function $\beta_\perp$ using the property that $\beta_\perp$ is the function $f \in \mathcal{H}$ that solves

$$\begin{aligned} \text{minimize:} \quad & \mathbb{E}\left[\{\beta(Z) - f(Z)\}^2\right] \\ \text{subject to:} \quad & h(f) = 0 \end{aligned} \tag{6}$$

Similar constrained learning problems have been studied in the context of ML with fairness constraints [31]. E.g. [47, 1] minimize a classification loss, while ensuring that predictions are uncorrelated with specific sensitive attributes (race, sex etc.). Letting $\lambda \in \mathbb{R}$ be a Lagrange multiplier, (6) is characterized by the Lagrangian

$$\mathcal{L}(f, \lambda) \equiv \mathbb{E}\left[\{\beta(Z) - f(Z)\}^2 + \lambda m(f, W)\right] \tag{7}$$

and a solution is obtained by finding $f^*, \lambda^* = \arg\max_{\lambda \in \mathbb{R}} \arg\min_{f \in \mathcal{H}} \mathcal{L}(f, \lambda)$. Naively, therefore, one might learn f by performing gradient descent over parameters indexing f and gradient ascent on λ, as in the basic differential multiplier method (BDMM) of [34].

In our numerical experiments, we consider the setting where $f = f_w$ is the output of a multilayer perceptron (MLP) with weights w. We observe that application of BDMM to a sample analogue of $\mathcal{L}(f_w, \lambda)$ leads to empirical constraint violations that oscillate around zero, as the number of ascent/descent iterations increases (shown in Figure 4 of the supplement). Similar behavior is documented elsewhere for adversarial function learners [38, 30]. Instead, stable constraint violations were achieved by effectively replacing the constraint in (6) with the equivalent constraint $|h(f)| \leq 0$, yielding the Lagrangian

$$\tilde{\mathcal{L}}(f, \tilde{\lambda}) \equiv \mathbb{E}\left[\{\beta(Z) - f(Z)\}^2\right] + \tilde{\lambda}|h(f)| \tag{8}$$

with empirical MLP analogue

$$\tilde{\mathcal{L}}_n(f_w, \tilde{\lambda}) \equiv \mathbb{E}_n\left[\{\beta(Z) - f_w(Z)\}^2\right] + \tilde{\lambda}|h_n(f_w)|. \tag{9}$$

In this formulation, $\tilde{\lambda} \geq 0$ penalizes the sample average moment of f_w in a similar way to the smoothing parameters in conventional Lasso/ridge regression. The key difference between these classical methods, however, is that the penalty $|h_n(f_w)|$ also depends on the observed data, and not only on the weights w. In practice we set $\tilde{\lambda}$ to a constant value during training, and minimize over w using gradient descent, though one might alternatively consider methods e.g. where $\tilde{\lambda}$ increases monotonically with the number of descent iterations (epochs).

2.4 Comparison with Automatic debiasing (AD)

AD [10, 11] is an RR learning method based on the identity

$$\begin{aligned}\alpha &= \underset{\hat{\alpha}\in\mathcal{H}}{\arg\min}\,||\alpha-\hat{\alpha}||^2\\ &= \underset{\hat{\alpha}\in\mathcal{H}}{\arg\min}\,||\hat{\alpha}||^2 - 2\langle\hat{\alpha},\alpha\rangle\\ &= \underset{\hat{\alpha}\in\mathcal{H}}{\arg\min}\,\mathbb{E}\left[\hat{\alpha}(Z)^2 - 2m(\hat{\alpha},W)\right].\end{aligned}$$

The AD RR learner minimizes a sample analogue of this expectation. We connect AD to our proposal as follows. Consider that $\hat{\alpha}\in\mathcal{H}$ can be written as $\hat{\alpha}(z) = 2\lambda^{-1}\{\beta(z) - f(z)\}$ where $\lambda \neq 0$ is constant and $f \in \mathcal{H}$. Thus, the AD population minimization becomes

$$\begin{aligned}\beta_\lambda &\equiv \underset{f\in\mathcal{H}}{\arg\min}\,\mathbb{E}\left[4\lambda^{-2}\{\beta(Z)-f(Z)\}^2 - 4\lambda^{-1}m(\beta - f, W)\right]\\ &= \underset{f\in\mathcal{H}}{\arg\min}\,\mathbb{E}\left[\{\beta(Z)-f(Z)\}^2 + \lambda m(f,W)\right].\end{aligned}$$

with $\alpha(z) = 2\lambda^{-1}\{\beta(z) - \beta_\lambda(z)\}$. The objective above is the Lagrangian $\mathcal{L}(f,\lambda)$ in (7). In this construction λ is unrestricted, therefore provided that $\lambda_\perp \equiv \arg\max_{\lambda\in\mathbb{R}} \min_{f\in\mathcal{H}} \mathcal{L}(f,\lambda)$ is finite and non-zero, one can write $\alpha(z) = 2\lambda_\perp^{-1}\{\beta(z) - \beta_{\lambda_\perp}(z)\}$. Appealing to the primal problem in (6), we see that $\beta_{\lambda_\perp} = \beta_\perp$ and, $\lambda_\perp = 2||\beta - \beta_\perp||^2/h(\beta)$ as in (4).

Moment-constrained learning, therefore, reinterprets the AD loss as a Lagrangian when one is agnostic to constants of proportionality in the RR. When estimating Ψ, these constants are estimated using the estimation sample rather than by the RR learner directly. Moreover, by connecting the RR to the primal problem, we are able to propose alternative constrained learning methods that may have better empirical performance, e.g. using the Lagrangian $\tilde{\mathcal{L}}$ in (8).

3 MADNet: Moment-constrained Automatic Debiasing Networks

Multi-headed MLPs, as illustrated through the schematic in Figure 2, are emerging as a popular architecture for estimating average moment estimands using deep learning. Initial efforts focused on the binary treatment setting, such as the multi-headed MLP outcome learner TARNet (Treatment Agnostic Representation Network) [39]. TARNet takes inputs X and produces two scalar outputs representing $\mu(1, X)$ and $\mu(0, X)$ respectively. During training, an outcome prediction error (e.g. mean-squared error) is minimized, with predictions of $\mu(A, X)$ obtained from one of the scalar outputs according to whether an observation is treated/untreated. The resulting outcome learner can be used to obtain plug-in estimates for e.g. the ATE/APE, optionally with debiasing by a separate RR learner as described in Section 2.1.

DragonNet [40] also focuses on binary treatments, extending TARNet by introducing a third scalar output MLP of zero depth, which is used to estimate the propensity score $p(X)$, and hence the RR of the ATE/APE. The authors reason that the propensity score MLP should have zero depth so that the shared MLP learns representations of X that are predictive of the RR, since, for ATE estimation it is sufficient to learn the outcome conditional on A and $p(X)$ only. This approach is generalized by Chernozhukov et al. [11, Lemma 3.1], where it is shown that: for estimation of Ψ it is sufficient to learn the outcome conditional on the RR only, i.e $E[Y|\alpha(Z) = \alpha(z)]$. Similarly, for estimation of Ψ, we show that it is sufficient to learn the outcome conditional on $\beta(Z) - \beta_\perp(Z)$ only. Moreover, we show

$$\mu(Z) = \mu_\perp(Z) + \frac{\Psi}{h(\beta)}\{\beta(Z) - \beta_\perp(Z)\} \tag{10}$$

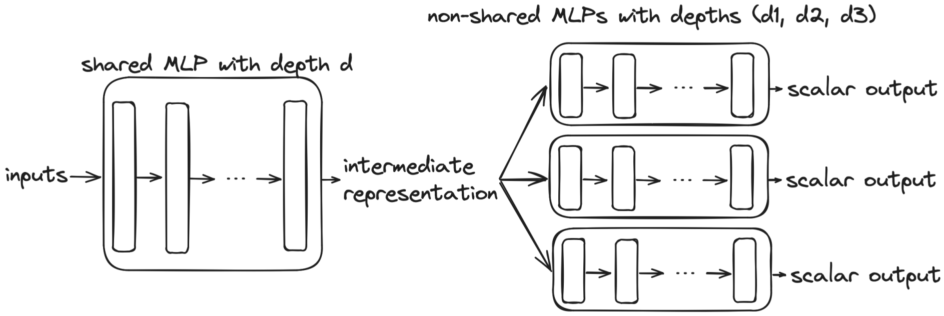

Figure 2: Multi-headed MLP architecture with three outputs. Typically the intermediate representation has the same width as the internal layers of the shared MLP, and the non-shared MLPs have internal layers with half the width of the shared MLP. During training, a single loss is used based on all scalar outputs, and MLP weights are learned using back-propagation over the entire multi-headed MLP.

where $\mu_\perp$ is the projection of μ on $\mathcal{C}^\perp$, i.e. $\mu_\perp = \arg\min_{f \in \mathcal{C}^\perp} \mathbb{E}[\{Y - f(Z)\}^2]$. This result further highlights the role of the RR when learning the outcome for average moment estimation.

RieszNet is similar in structure to DragonNet, except with input $Z = (A, X)$ rather than X. Both also use a multi-tasking loss to learn μ and α simultaneously. We propose MADNet which uses the same network structure as RieszNet and a multi-tasking loss to learn μ and $\beta_\perp$ simultaneously. Specifically we consider the loss

$$\tilde{\mathcal{L}}_n(f_{w,1}, \tilde{\lambda}) + \rho \, \text{REGLoss}_n(f_{w,2}) \tag{11}$$

where $\rho \geq 0$ is a hyperparameter, REGLoss_n is a regression loss, e.g. the mean-squared error in the outcome prediction $\text{REGLoss}_n(f) = \mathbb{E}_n[\{Y - f(Z)\}^2]$, and $f_w(z) = (f_{w,1}(z), f_{w,2}(z))$ represents two outputs from a multi-headed MLP. Note that e.g. $f_{w,1}$ depends only on the weights of the shared MLP and the first non-shared MLP, but we write it as a function of all multi-headed MLP weights w for convenience. Like RieszNet, for ATE/ADE estimation we replace $f_{w,2}(z)$ in (11) with $\tilde{a} f_{w,2}(z) + (1 - \tilde{a}) f_{w,3}$, where $\tilde{a}$ represents the min-max normalized treatment a scaled on to the interval $[0, 1]$, i.e. $\tilde{a}_i = \{a_i - \min_n(a_i)\}/\{\max_n(a_i) - \min_n(a_i)\}$.

Convergence rates: The standard theory in Section 2.1 controls first-order remainders by requiring estimators to converge to their true counterparts at sufficiently fast rates. For neural network learners, recent convergence rate results have been obtained using the theory of critical radii [45, 17, 13]. In particular, results are provided for MLPs with Rectified Linear Unit (ReLU) activation functions [16], describing L_2 convergence rates in terms of the number of training observations and the network width/depth. Similar results exist for AD learners with moments satisfying the mean-squared continuity property $\mathbb{E}[\{m(u, W) - m(v, W)\}^2] \leq M||u - v||^2$ for $M \geq 1$ and $u, v \in \mathcal{A}_n$, where $\mathcal{A}_n$ is a function set described by Chernozhukov et al. [10]. Due to the connection of AD with moment-constrained learning discussed above, we expect similar theoretical guarantees to hold in the current context for mean-squared continuous moments.

Sample-splitting: To obtain valid inference from a single sample, the standard theory in Section 2.1 relies on cross-fitting to control the second-order remainder terms in (1). Cross-fitting is used to bypass Donsker class assumptions [4, 48, 9], which restrict the complexity of the initial estimators and are usually not satisfied by ML algorithms. Recent work has sought to bypass Donsker conditions by instead relying on entropic arguments [42] or leave-one-out stability [8]. In practice DragonNet and RieszNet do not use sample splitting due to the associated computational burden. Instead, implementations of both algorithms split the data into training and validation sets, with the validation set used to control early stopping of the training algorithm. For estimation, the full dataset is used (training + validation). We also use this strategy for moment-constrained learning.

4 Numerical experiments

We consider ATE and ADE estimation in the following semi-synthetic data scenarios.

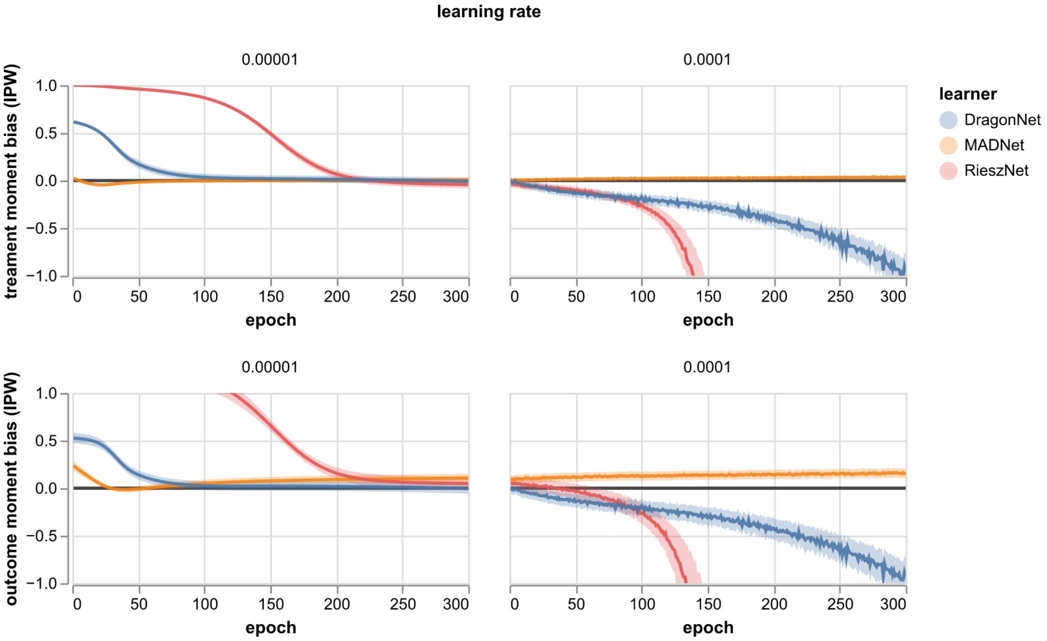

Figure 3: Top row: mean and standard error of the treatment moment bias $\mathbb{E}_n[A\hat{\alpha}(Z)] - \mathbb{E}[A\alpha(Z)]$ with $\mathbb{E}[A\alpha(Z)] = 1$ known *a priori*. Bottom row: mean and standard error of the outcome moment bias $\hat{\Psi}^{(\text{IPW})} - \Psi = \mathbb{E}_n[Y\hat{\alpha}(Z)] - \Psi$ with the synthetic outcome moment obtained as $h_n(\mu)$. Both plots use 20 datasets of IHDP data whereby predictions are made on a 20% validation set and the outcome moment is scaled by the standard deviation of the outcome. The zero bias line is accentuated for clarity.

IHDP: (Infant Health and Development Program). IHDP is a randomized experiment on the effects of home visits by specialists (binary treatment, A) on infant cognition scores (Y), given 25 baseline covariates. The data consists of n = 747 infants. Synthetic outcomes are drawn from a normal distribution given (A, X), as described by [20]. We consider 1000 synthetic IHDP datasets in total.

BHP: (Blundell, Horowitz and Parey) [5]. BHP consists of 3,640 household level observations from the 2001 (U.S.) National Household Travel Survey, with the goal of estimating the price elasticity of gasoline consumption, given 18 confounding variables. Price elasticity can be defined through the ADE of log price (A) on the log quantity of gasoline sold (Y). Following the experiments of [11], we draw synthetic treatments from a normal distribution, with conditional mean and variance obtained from random forest predictions of the mean and variance of the true log price. Conditionally normal synthetic outcomes are then generated, given (A, X), with a mean function that is cubic in treatment. The results in Table 1 are evaluated over 200 random seeds.

The first experimental research question is as follows: to what extent do RR predictions from moment-constrained auto-debiasing learners satisfy oracle RR properties that are known *a priori*? To answer this, we consider Inverse Probability Weighted (IPW) estimators of $\mathbb{E}[A\alpha(Z)] = 1$, and Ψ. The corresponding IPW estimators, $\mathbb{E}_n[A\hat{\alpha}(Z)]$ and $\mathbb{E}_n[Y\hat{\alpha}(Z)]$, do not depend on the outcome model, thus are a convenient way of comparing RR learners. Figure 3 shows, using IHDP data, how the mean error evolves over gradient descent iterations (epochs) and for two different learning rates. These plots show that the MADNet RR estimator converges rapidly to a stable optimum, and is therefore more robust to changes in learning rate and early stopping hyperparameters.

Next we compare absolute errors of MADNet estimators versus several alternatives: DragonNet [40], Reproducing Kernel Hilbert Space (RKHS) Embedding [41], Neural Net (NN) Embedding [46], and RieszNet [11], with only the latter applying to ADE estimation in the BHP scenario. Results in Table 1 show that the Double Robust MADNet estimator has improved empirical performance across all scenarios.

Table 1: Absolute error (mean ± standard error) of the ATE and ADE estimates across the semi-synthetic data scenarios described in Section 4. The lowest mean absolute error (MAE) is shown in bold for each dataset. The RieszNet IHDP benchmark values deviate from those reported in Chernozhukov et al. [11]. We report values obtained by running `RieszNet_IHDP.ipynb` from `https://github.com/victor5as/RieszLearning` (the code repository that accompanies [11]) without modification. MAEs for the BHP scenario are not reported in [11], and instead we use our re-implementation of RieszNet (Table 2 in the supplement contains full reproduction results). Values for DragonNet and RKHS/NN embedding are retrieved from Xu and Gretton [46, Table 1].

Estimator	IHDP	BHP	Citation
DragonNet (DR)	0.146 ± 0.010	–	[40]
RKHS Embedding	0.166 ± 0.003	–	[41]
NN Embedding	0.117 ± 0.002	–	[46]
RieszNet (Direct)	0.128 ± 0.004	0.692 ± 0.040	[11]
RieszNet (IPW)	0.789 ± 0.036	0.449 ± 0.025	[11]
RieszNet (DR)	0.114 ± 0.003	0.428 ± 0.023	[11]
MADNet (Direct)	0.504 ± 0.016	0.471 ± 0.026	Proposed
MADNet (IPW)	0.719 ± 0.039	0.474 ± 0.026	Proposed
MADNet (DR)	**0.094 ± 0.002**	**0.391 ± 0.019**	Proposed

Our implementation is built on a JAX+Equinox computational stack [6, 28]. Detailed implementation notes are provided in the supplement, as well as DragonNet/RieszNet results obtained using our implementation (Table 2 of the supplement). We highlight, however, two main difference in our implementation: (i) for ADE estimation we use automatic differentiation of NN outputs, rather than a finite difference approximation; (ii) the original RieszNet uses a complicated learning rate / early stopping scheme to circumvent the stability issues in Figure 3; we use a slightly simpler scheme. Overall we found that replication via re-implementation of the RieszNet results was challenging, possibly due to the aforementioned stability issues in the AD loss.

5 Conclusion

We present a new algorithm for estimating average moment estimands, inspired by recent work on automatic debiasing (AD). Our approach leverages functions for which the average moment is known *a priori* to be non-zero. We contend that constructing such functions is significantly simpler than deriving the functional form of the RR. Our proposal is 'automatic' in the sense of not requiring complicated estimand-specific machinery. Moreover, rather than learning the full RR, as in conventional AD, we instead learn a moment-constrained function that is sufficient for debiasing the naive average moment estimator. We propose a Lagrange-type penalization method for moment-constrained learning and apply this method using multi-tasking neural networks.

There are several directions which one might extend our work. First, our set up considers estimands that represent the average moment of a regression functions, but extensions of AD to so-called generalized regressions (e.g. quantile functions) have been considered by other authors [10]. Similar extensions for moment-constrained learning should also be possible in these settings. Second, we consider neural network learners, but similar extensions for gradient boosted trees / random forests should also be possible. Third, our numerical experiments consider two common estimands for binary and continuous treatments, but further empirical comparisons for other average moment estimands are needed. Finally, our approach to learning moment-constrained functions may be applied to other problems with stochastic constraints e.g. constraints related to fairness of ML predictions.

References

[1] Akhtar, Z., Bedi, A., and Rajawat, K. (2021). Conservative Stochastic Optimization with Expectation Constraints. *IEEE Transactions on Signal Processing*, 69:3190–3205.

[2] Athey, S., Imbens, G. W., and Wager, S. (2018). Approximate residual balancing: debiased inference of average treatment effects in high dimensions. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 80(4):597–623.

[3] Athey, S. and Wager, S. (2021). Policy Learning With Observational Data. *Econometrica*, 89(1):133–161.

[4] Bickel, P. J. (1982). On Adaptive Estimation. *The Annals of Statistics*, 10(3).

[5] Blundell, R., Horowitz, J., and Parey, M. (2017). Nonparametric estimation of a nonseparable demand function under the slutsky inequality restriction. *Review of Economics and Statistics*, 99(2):291–304.

[6] Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., and Zhang, Q. (2018). JAX: composable transformations of Python+NumPy programs (v0.3.13).

[7] Cattaneo, M. D., Crump, R. K., and Jansson, M. (2013). Generalized jackknife estimators of weighted average derivatives. *Journal of the American Statistical Association*, 108(504):1243–1256.

[8] Chen, Q., Syrgkanis, V., and Austern, M. (2022). Debiased Machine Learning without Sample-Splitting for Stable Estimators. *Advances in Neural Information Processing Systems*, 35:1–48.

[9] Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W. K., and Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. *Econometrics Journal*, 21(1):C1–C68.

[10] Chernozhukov, V., Newey, W. K., Quintas-Martinez, V., and Syrgkanis, V. (2021). Automatic Debiased Machine Learning via Neural Nets for Generalized Linear Regression. *arXiv (2104.14737)*.

[11] Chernozhukov, V., Newey, W. K., Quintas-Martínez, V., and Syrgkanis, V. (2022a). RieszNet and ForestRiesz: Automatic Debiased Machine Learning with Neural Nets and Random Forests. *Proceedings of Machine Learning Research*, 162:3901–3914.

[12] Chernozhukov, V., Newey, W. K., and Singh, R. (2022b). Debiased machine learning of global and local parameters using regularized Riesz representers. *The Econometrics Journal*, 25(3):576–601.

[13] Chernozhukov, V., Newey, W. K., Singh, R., and Syrgkanis, V. (2020). Adversarial Estimation of Riesz Representers. *arXiv (2101.00009)*.

[14] Dikkala, N., Lewis, G., Mackey, L., and Syrgkanis, V. (2020). Minimax estimation of conditional moment models. *Advances in Neural Information Processing Systems*, 2020-December(NeurIPS).

[15] Dudik, M., Langford, J., and Li, H. (2011). Doubly robust policy evaluation and learning. *Proceedings of the 28th International Conference on Machine Learning, ICML 2011*, pages 1097–1104.

[16] Farrell, M. H., Liang, T., and Misra, S. (2021). Deep Neural Networks for Estimation and Inference. *Econometrica*, 89(1):181–213.

[17] Foster, D. J. and Syrgkanis, V. (2023). Orthogonal Statistical Learning. *Annals of Statistics*, 51(3):879–908.

[18] Hampel, F. R. (1974). The Influence Curve and its Role in Robust Estimation. *Journal of the American Statistical Association*, 69(346):383–393.

[19] Härdle, W. and Stoker, T. M. (1989). Investigating smooth multiple regression by the method of average derivatives. *Journal of the American Statistical Association*, 84(408):986–995.

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

[21] Hines, O. J., Diaz-Ordaz, K., and Vansteelandt, S. (2023). Variable importance measures for heterogeneous causal effects. *arXiv (2204.06030)*.

[22] Hines, O. J., Dukes, O., Diaz-Ordaz, K., and Vansteelandt, S. (2022). Demystifying Statistical Learning Based on Efficient Influence Functions. *The American Statistician*, 76(3):292–304.

[23] Hirshberg, D. A. and Wager, S. (2021). Augmented minimax linear estimation. *The Annals of Statistics*, 49(6).

[24] Ichimura, H. and Newey, W. K. (2022). The influence function of semiparametric estimators. *Quantitative Economics*, 13(1):29–61.

[25] Imbens, G. W. and Newey, W. K. (2009). Identification and Estimation of Triangular Simultaneous Equations Models Without Additivity. *Econometrica*, 77(5):1481–1512.

[26] Kennedy, E. H. (2022). Semiparametric doubly robust targeted double machine learning: a review. *arXiv (2203.06469)*.

[27] Kennedy, E. H. (2023). Towards optimal doubly robust estimation of heterogeneous causal effects. *Electronic Journal of Statistics*, 17(2):3008–3049.

[28] Kidger, P. and Garcia, C. (2021). Equinox: neural networks in JAX via callable PyTrees and filtered transformations. *Differentiable Programming workshop at Neural Information Processing Systems 2021*.

[29] Loshchilov, I. and Hutter, F. (2019). Decoupled Weight Decay Regularization.

[30] Mokhtari, A., Ozdaglar, A., and Pattathil, S. (2020). A Unified Analysis of Extra-gradient and Optimistic Gradient Methods for Saddle Point Problems: Proximal Point Approach. *Proceedings of Machine Learning Research*, 108:1497–1507.

[31] Nabi, R., Hejazi, N. S., van der Laan, M. J., and Benkeser, D. (2024). Statistical learning for constrained functional parameters in infinite-dimensional models with applications in fair machine learning. *arXiv (2404 . 09847)*.

[32] Newey, W. K. (1994). The Asymptotic Variance of Semiparametric Estimators. *Econometrica*, 62(6):1349.

[33] Newey, W. K. and Stoker, T. M. (1993). Efficiency of Weighted Average Derivative Estimators and Index Models. *Econometrica*, 61(5):1199.

[34] Platt, J. and Barr, A. (1987). Constrained differential optimization. In Anderson, D., editor, *Neural Information Processing Systems*, volume 0. American Institute of Physics.

[35] Robins, J. M., Rotnitzky, A., and Zhao, L. P. (1994). Estimation of Regression Coefficients When Some Regressors Are Not Always Observed. *Journal of the American Statistical Association*, 89(427):846.

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

[37] Rothenhäusler, D. and Yu, B. (2019). Incremental causal effects. *arXiv (1907.13258)*.

[38] Schäfer, F. and Anandkumar, A. (2019). Competitive gradient descent. *Advances in Neural Information Processing Systems*, 32(NeurIPS).

[39] Shalit, U., Johansson, F. D., and Sontag, D. (2017). Estimating individual treatment effect: Generalization bounds and algorithms. *34th International Conference on Machine Learning, ICML 2017*, 6:4709–4718.

[40] Shi, C., Blei, D. M., and Veitch, V. (2019). Adapting neural networks for the estimation of treatment effects. *Advances in Neural Information Processing Systems*, 32:1–14.

[41] Singh, R., Xu, L., and Gretton, A. (2023). Kernel methods for causal functions: dose, heterogeneous and incremental response curves. *Biometrika*.

[42] van de Geer, S., Bühlmann, P., Ritov, Y., and Dezeure, R. (2014). On asymptotically optimal confidence regions and tests for high-dimensional models. *Annals of Statistics*, 42(3):1166–1202.

[43] van der Laan, M. J. and Gruber, S. (2016). One-Step Targeted Minimum Loss-based Estimation Based on Universal Least Favorable One-Dimensional Submodels. *International Journal of Biostatistics*, 12(1):351–378.

[44] van der Laan, M. J. and Luedtke, A. R. (2014). Targeted Learning of the Mean Outcome under an Optimal Dynamic Treatment Rule. *Journal of Causal Inference*, 3(1):61–95.

[45] Wainwright, M. J. (2019). *High-Dimensional Statistics*. Cambridge University Press.

[46] Xu, L. and Gretton, A. (2023). A neural mean embedding approach for back-door and front-door adjustment. In *The Eleventh International Conference on Learning Representations*.

[47] Zafar, M. B., Valera, I., Rodriguez, M. G., and Gummadi, K. P. (2017). Fairness constraints: Mechanisms for fair classification. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017*, 54.

[48] Zheng, W. and van der Laan, M. J. (2011). Cross-validated targeted minimum-loss-based estimation. In *Targeted Learning*, pages 459–474. Springer New York, New York, NY.

[49] Zubizarreta, J. R. (2015). Stable Weights that Balance Covariates for Estimation With Incomplete Outcome Data. *Journal of the American Statistical Association*, 110(511):910–922.

Supplement to: Automatic debiasing of neural networks via moment-constrained learning

Equation and reference numbering in this document continues from that of the main manuscript.

A Notes on the numerical experiments

A.1 MADNet architecture

To ensure a fair evaluation, our proposed MADNet architecture emulates that of the RieszNet [11] (see Figure 2 for a schematic of the multi-headed architecture). MADNet uses a shared network of width 200 and depth 3 followed by three branches: 2 outcome networks (one per binary treatment) each of width 100 and depth 2 and another of depth zero, i.e. a linear combination of the final shared representation layer that is our $\hat{\beta}_{\perp}$ prediction. The constraint weight hyperparameter was set to $\tilde{\lambda} = 5$, the weight mixing parameter was set to $\rho = 1$, and Exponential Linear Unit (ELU) activation functions were used throughout. Finally, outcomes Y were scaled by their sample standard deviation prior to training, with predictions rescaled to the original scale using the same constant standard deviation estimate.

A.2 MADNet training details

Numerical experiments were run on an Apple M2 Max chip with 32GB of RAM. The MADNet training procedure was also borrowed from Chernozhukov et al. [11, Appendix A1], which itself was borrowed from Shi et al. [40]. Minor modifications are outlined below. The dataset was split into a training dataset (80%) and validation dataset (20%), with estimation performed on the entire dataset. The training followed a two stage procedure outlined below.

ATE benchmarks

1. Fast training: batch size: 64, learning rate: 0.0001, maximum number of epochs: 100, optimizer: Adam, early stopping patience: 2, L2 weight decay: 0.001
2. Fine-tuning: batch size: 64, learning rate: 0.00001, maximum number of epochs: 600, optimizer: Adam, early stopping patience: 40, L2 weight decay: 0.001

ADE benchmarks

1. Fast-training: batch size: 64, learning rate: 0.001, maximum number of epochs: 100, optimizer: Adam, early stopping patience: 2, L2 weight decay: 0.001
2. Fine-tuning: batch size: 64, learning rate: 0.0001, maximum number of epochs: 300, optimizer: Adam, early stopping patience: 20, L2 weight decay: 0.001

The differences between the original implementations and ours are:

- For ADE moment estimation, RieszNet uses a finite difference approximation to differentiate the forward pass with respect to the treatment a. However our implementation uses automatic differentiation provided by JAX. One of the advantages of JAX is that the ADE can be straightforwardly expressed as `jax.grad(f)(a, x)`.
- On top of the early stopping callback, the original RieszNet and DragonNet implementations additionally use a learning rate plateau schedule that halves the learning rate when the validation loss metric has stopped improving over a short patience of epochs (shorter than the stopping patience). Whilst we implement the same two-stage training with early stopping, we use a constant learning rate in each of the fast-training and fine-tuning phases.
- L2 regularization is implemented differently between RieszNet and DragonNet. DragonNet use a regularizer to apply a penalty on the layer's kernel whilst RieszNet uses an additive L2 regularization term in their loss function [11, Equation 5]. However, recent work shows that L2 regularization and weight decay regularization are not equivalent for adaptive gradient algorithms, such as Adam [29]. For this reason, we use Adam with weight decay regularization (provided by `optax.adamw`).

Table 2: Full reproduction results for our own implementation of each learner/estimator. Here + SRR, refers to estimator which use the outcome model $\tilde{g}$ described in [11].

		Mean Absolute Error (MAE)	Median Absolute Error	Standard Error in MAE
Dataset	Estimator			
BHP	RieszNet (DR + SRR)	0.428	0.355	0.023
	RieszNet (DR)	0.428	0.353	0.023
	RieszNet (Direct + SRR)	0.724	0.617	0.042
	RieszNet (Direct)	0.692	0.585	0.040
	RieszNet (IPW)	0.449	0.384	0.025
	MADNet (DR)	0.391	0.346	0.019
	MADNet (Direct)	0.471	0.424	0.026
	MADNet (IPW)	0.474	0.404	0.026
IHDP	DragonNet (DR + SRR)	0.101	0.085	0.003
	DragonNet (DR)	0.100	0.084	0.002
	DragonNet (Direct + SRR)	0.124	0.098	0.004
	DragonNet (Direct)	0.123	0.098	0.004
	DragonNet (IPW + SRR)	0.262	0.233	0.006
	DragonNet (IPW)	0.262	0.233	0.006
	RieszNet (DR + SRR)	0.109	0.088	0.003
	RieszNet (DR)	0.109	0.089	0.003
	RieszNet (Direct + SRR)	0.126	0.102	0.004
	RieszNet (Direct)	0.118	0.099	0.003
	RieszNet (IPW)	0.665	0.300	0.036
	MADNet (DR)	0.094	0.076	0.002
	MADNet (Direct)	0.504	0.367	0.016
	MADNet (IPW)	0.719	0.277	0.039

A.3 Naive Lagrangian optimization

We consider the basic differential multiplier method (BDMM), as described by Platt et al. [34]. The authors introduce a so-called damping term $\delta \geq 0$ to the Lagrangian in (7) to obtain the Lagrangian

$$\mathcal{L}_\delta(f, \lambda) \equiv \mathbb{E}\left[\{\beta(Z) - f(Z)\}^2\right] + \lambda h(f) + \delta h^2(f),$$

with (7) recovered by setting $\delta = 0$. Note that when the moment constraint is satisfied, i.e.$h(f) = 0$, then $\mathcal{L}_\delta$ does not depend on δ. In Figure 4 we see how Naively performing gradient ascent on λ and gradient descent over f results in oscillatory behavior. Similar behavior is also observed in the literature on adversarial learning, see e.g. [38, 30].

B Short notes and proofs

B.1 Proof of orthogonality representation

Claim:

$$\mu(z) = \mu_\perp(z) + \frac{\Psi}{h(\beta)}\{\beta(z) - \beta_\perp(z)\}. \tag{12}$$

Proof: Note that $\mathcal{C}^\perp = \{f \in \mathcal{H} \mid \langle f, \alpha \rangle = 0\}$ then by Hilbert's projection theorem, $\mu_\perp$ exists, with

$$\mu_\perp(z) \equiv \mu(z) - \frac{\langle \mu, \alpha \rangle}{||\alpha||^2}\alpha(z)$$

Using $\langle \mu, \alpha \rangle = \Psi$ and applying (4) completes the proof of (12).

B.2 Sufficiency of learning conditional on the unscaled RR

Claim: $\Psi = h(\eta)$, where

$$\eta(z) \equiv \mathbb{E}[Y|\beta(Z) - \beta_\perp(Z) = \beta(z) - \beta_\perp(z)].$$

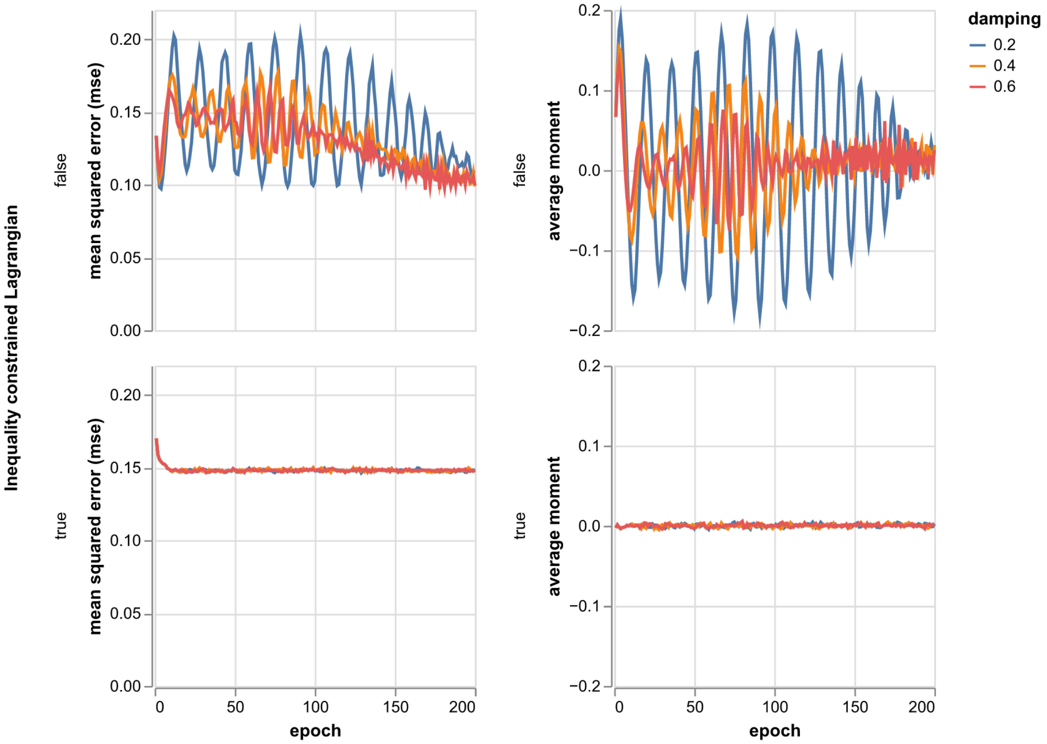

Figure 4: Top row: Low damping coefficients in the basic differential multiplier method (BDMM) [34] lead to oscillatory behavior around the saddle point solution when the optimisation problem is formulated as an equality constrained Lagrangian. Bottom row: Using the inequality constrained Lagrangian approach described in the main paper results in more stable training and constraint satisfaction. A single dataset from the IHDP data is used to showcase this behavior over 200 epochs.

Proof:

$$\begin{aligned}\Psi &= \mathbb{E}[Y\alpha(Z)]\\ &= \frac{h(\beta)\mathbb{E}[Y\{\beta(Z)-\beta_\perp(Z)\}]}{||\beta-\beta_\perp||}\\ &= \frac{h(\beta)\mathbb{E}[\eta(Z)\{\beta(Z)-\beta_\perp(Z)\}]}{||\beta-\beta_\perp||}\\ &= \mathbb{E}[\eta(Z)\alpha(Z)]\end{aligned}$$

where in the third step we apply the law of iterated expectation.

B.3 First-order remainder under the standard theory

Claim: $\mathbb{E}[\hat{\varphi}(W)-\Psi] = -\langle\hat{\mu}-\mu, \hat{\alpha}-\alpha\rangle$.

Proof:

$$\begin{aligned}&\mathbb{E}[m(\hat{\mu},W)+\hat{\alpha}(Z)\{Y-\hat{\mu}(Z)\}-\Psi]\\ =&\mathbb{E}[m(\hat{\mu},W)+\hat{\alpha}(Z)\{\mu(Z)-\hat{\mu}(Z)\}-m(\mu,W)]\\ =&\mathbb{E}[m(\hat{\mu}-\mu,W)-\hat{\alpha}(Z)\{\hat{\mu}(Z)-\mu(Z)\}]\\ =&\langle\hat{\mu}-\mu,\alpha\rangle-\langle\hat{\mu}-\mu,\hat{\alpha}\rangle\\ =&-\langle\hat{\mu}-\mu,\hat{\alpha}-\alpha\rangle.\end{aligned}$$

B.4 Second-order remainder under the standard theory

Claim: If $\hat{\mu}$ and $\hat{\alpha}$ are independent consistent estimators, and there exists a constant k such that $\alpha^2(Z) < k$ and $\text{var}(Y|Z) < k$ almost surely, then $G_n\{\hat{\varphi}(W) - \varphi(W)\} = o_p(1)$.

Proof:

$$\begin{aligned} G_n[\hat{\varphi}(W) - \varphi(W)] = &+ G_n[m(\hat{\mu} - \mu, W)] \\ &+ G_n\left[\{\hat{\alpha}(Z) - \alpha(Z)\}\{\hat{\mu}(Z) - \mu(Z)\}\right] \\ &- G_n\left[\alpha(Z)\{\hat{\mu}(Z) - \mu(Z)\}\right] \\ &+ G_n\left[\{\hat{\alpha}(Z) - \alpha(Z)\}\{Y - \mu(Z)\}\right] \end{aligned}$$

By the central limit theorem, these empirical processes are $o_p(1)$ when the following expressions are $o_p(1)$

$$\begin{aligned} &\mathbb{E}\left[m^2(\hat{\mu} - \mu, W)\right] \\ &\mathbb{E}\left[\{\hat{\alpha}(Z) - \alpha(Z)\}^2\{\hat{\mu}(Z) - \mu(Z)\}^2\right] \\ &\mathbb{E}\left[\alpha^2(Z)\{\hat{\mu}(Z) - \mu(Z)\}^2\right] \\ &\mathbb{E}\left[\{\hat{\alpha}(Z) - \alpha(Z)\}^2\{Y - \mu(Z)\}^2\right] \end{aligned}$$

The first two terms are $o_p(1)$ by consistency of $\hat{\alpha}$ and $\hat{\mu}$, for the final two terms

$$\begin{aligned} \mathbb{E}\left[\alpha^2(Z)\{\hat{\mu}(Z) - \mu(Z)\}^2\right] &< k||\hat{\mu} - \mu||^2 \\ \mathbb{E}\left[\{\hat{\alpha}(Z) - \alpha(Z)\}^2\text{var}(Y|Z)\right] &< k||\hat{\alpha} - \alpha||^2 \end{aligned}$$

hence, these are also $o_p(1)$ by consistency.

Remark: The requirement for estimator independence can be relaxed if one makes Donsker class assumptions instead.

B.5 First-order remainder for moment-constrained learning TMLE estimators

Since TMLE estimators are agnostic to proportionality constants in the RR, we consider (1) for the uncentred influence function

$$\hat{\varphi}^*(W) = m(\hat{\mu}^*, W) + \frac{h(\beta)}{||\beta - \beta_\perp||^2}\{\beta(Z) - \hat{\beta}_\perp(Z)\}\{Y - \hat{\mu}^*(Z)\},$$

and $\psi = \hat{\Psi}^{(\text{TMLE})} = h_n(\hat{\mu}^*)$. Claim:

$$\mathbb{E}[\hat{\varphi}^*(W) - \hat{\Psi}^{(\text{TMLE})}] = \frac{h(\beta)}{||\beta - \beta_\perp||^2}\langle\hat{\mu}^* - \mu, \hat{\beta}_\perp - \beta_\perp\rangle.$$

Proof: This result follows immediately from (4) is applied to the result in Section B.3 with $\hat{\mu}$ replaced with $\hat{\mu}^*$ and

$$\hat{\alpha}(Z) = \frac{h(\beta)}{||\beta - \beta_\perp||^2}\{\beta(Z) - \hat{\beta}_\perp(Z)\}.$$