Predictive State Propensity Subclassification (PSPS): A causal inference method for optimal data-driven propensity score stratification

Joseph Kelly* josephkelly@google.com

Jing Kong jingkong@google.com

Georg M. Goerg † georg@evolutioniq.com

June 5, 2020

Abstract

We introduce Predictive State Propensity Subclassification (PSPS), a novel estimation method for undertaking causal inference from observational studies. PSPS combines propensity and outcome models into one encompassing probabilistic framework, which can be jointly estimated using maximum likelihood or Bayesian inference. The methodology applies to both discrete and continuous treatments and can estimate unit-level and population-level average treatment effects. We give a detailed overview on the TensorFlow implementation for likelihood optimization and show via large-scale simulations that it outperforms several state of the art methods – both in terms of bias for average treatment effects (ATEs) and root mean square error (RMSE) for unit-level treatment effects (UTEs). Finally we illustrate the methodology and algorithms on standard datasets in the causal inference literature.

1 Introduction

In this work we are developing a novel framework for estimating the effect of an intervention on a measurable outcome of interest. For example, medical researchers are interested in finding out the effect on blood pressure of a new medication or advertisers would like to estimate the effect an advertisement may have on the chance of purchasing the advertised product. In an ideal world a randomized experiment would be run to help estimate these effects.

Randomization ensures balance in expectation on important confounders – variables that effect both the treatment assignment and outcome metric – across the different treatment arms allowing for unbiased estimates of the quantity of interest (the causal effect). Randomized experiments, however, are not always feasible. Time, cost, legal, or ethical reasons are among

*Corresponding author.

†This work was completed while the author was at Google.

many of the reasons that may prevent a randomized experiment from taking place. In that case the decision maker has to rely on observational data – otherwise known as a natural experiment – to try and estimate the causal effects.

The lack of randomization leading to covariate imbalances between the treatment arms in natural experiments make identification and estimation of causal effects difficult. In a randomized experiment an experimenter may block or stratify on certain covariates in order to ensure they are balanced between different treatment assignments for a particular randomization (Bernstein, 1927). Causal inference methods for natural experiments do not have the benefit of randomization and use other means to help mitigate the imbalances. As such they are often viewed as tools to help fix broken randomized experiments (Rubin, 2008).

The identification of causal effects and the assumptions needed to link potential outcomes under different treatments to the observed data is well established and will not be examined here in detail (Rubin, 2005). Although we too will make these assumptions the focus will be on creating an estimator with good performance that adjusts for the observed imbalances on the distribution of covariates between different levels of treatment. The idea being that the estimator is trying to re-create balance between different treatment levels that did not occur due to the lack of randomization.

Existing methods that aim to do this are often based on either matching methods (Stuart, 2010) or inverse propensity score weighting coupled with some outcome modeling. The latter include entropy balancing (Hainmueller, 2012) and the double robust estimator (Chan et al., 2010) among others. Cochran (1968) proposed *subclassification*, which addresses covariate imbalance by creating strata of observations such that within each strata there is approximate covariate balance between treated and control units.

In this work we introduce Predictive State Propensity Subclassification (PSPS), a novel causal inference method leveraging recent advances in multi-task learning (Ruder, 2017). PSPS follows the Rubin Causal Model (RCM) framework (Imbens and Rubin, 2010) and is in the same vein as matching, inverse-propensity score weighting and traditional subclassification that aim to fix the broken design of the experiment. The key contribution of our work is to incorporate predictive state smoothing (PRESS) (Goerg, 2017, 2018) for the propensity model, which yields a principled data-driven way to obtain the *strata* or *blocks* used in traditional approaches based on pre-treatment covariates.

This paper is organized as follows: Section 2 introduces PSPS with details on parameter estimation via maximum likelihood and particular emphasis on the trade-off between optimizing propensity and outcome models. Section 3 describes how to obtain treatment effect estimates from a trained PSPS model. In Section 4 we show via simulations of standard datasets in the literature (Kang and Schafer, 2007, Dehejia and Wahba, 1999, Radcliffe, 2007) that PSPS has excellent statistical properties, with practically zero bias and low variance compared to several state-of-the-art causal inference methods. Section 5 summarizes the proposed methodology and empirical findings.

2 Methodology

For the remainder of this work, let T be the treatment variable, Y the observed outcome, and $\mathbf{X}$ is a collection of pre-treatment covariates. The dimensionality and variable type of T, Y and $\mathbf{X}$ are left intentionally vague as the core underlying concepts we present hold for multivariate outcomes and multi-level or continuous treatments.

Under a randomized-block experiment $P(T \mid \mathbf{X})$ is known by construction and the properties of randomization meet the ignorable treatment assignment (Rosenbaum and Rubin, 1983) allowing for the identification and estimation of causal effects. In a natural experiment the distribution of the treatment assignment is typically not known and as such the ignorable treatment assignment is usually made on a leap of faith after adjusting for any observed imbalances.

To deal with potential self-selection bias and imbalances between treated and control units, Rosenbaum and Rubin (1985) proposed *subclassification* on the propensity score, which stratifies units by a coarsened value of the fitted propensity scores $\widehat{P(T \mid \mathbf{X})}$, – usually accomplished by binning; it is also often accompanied by a model-based adjustment to deal with any remaining imbalances. The idea being that observations within each bin have approximately the same propensity score and thus are more robust to any misspecification in the outcome models of each bin. The (predicted) outcome value for treated and controls can then be compared within each strata to estimate the effect of treatment. As separate outcome models or adjustments are made in each strata this approach typically relies less on extrapolation and modeling assumptions.

One common problem with this approach is that there are little to no guidelines on how to actually create the subclasses and most rely on heuristics. Common approaches are to a) bin on quantiles of the propensity score, b) have equally sized bins, or c) try to at least make the largest bin size moderately small. Imbens and Rubin (2015) propose creating bins following a recursive algorithm which tests whether propensity score means are equal between treated and control via a t-test. If the test statistic is above some pre-specified threshold then the strata is split at the median and the process is repeated on the sub-strata. This continues until the test statistic is not significant or the pre-specified minimum number of treated and controls are met. The end result, however, is inevitably an ad-hoc approach.

2.1 Predictive State Propensity Subclassification (PSPS)

PSPS follows the same experimental design as traditional subclassification with the exception of how subclasses are created and how subsequent analyses take place. The core causal assumptions such as ignorability of the treatment assignment are the same. We rely on these established results (Imbens and Rubin, 2015) and move straight to the model on the observed data bypassing the setup of potential outcomes and application of relevant assumptions that are necessary in order to identify and estimate causal effects.

Statistically, treatment assignment and application are confounded. In essence all causal methods aim to break this confounding between assignment and the application effect of treatment on the outcome. In PSPS this dual role of treatment is made explicit by decomposing the joint distribution

$$\begin{aligned} p(Y,T \mid \mathbf{X}) &= p_{\theta_Y}(Y \mid T, \mathbf{X}) \cdot p_{\theta_T}(T \mid \mathbf{X}) \\ &= p_{\theta_Y}(Y \mid T_{apply}, \mathbf{X}) \cdot p_{\theta_T}(T_{assign} \mid \mathbf{X}), \end{aligned} \tag{1}$$

where θ_Y and θ_T parameterize the conditional distributions of Y and T, respectively. Eq. (1) makes it explicit that T plays two roles: i) once as the output of a propensity model (treatment assignment, T_{assign}), and ii) as an input to an outcome model (treatment application, T_{apply}). Since assignment and application are confounded we modulate the model on the observed outcome $Y \mid T_{apply}, \mathbf{X}$ with the fact that there was self-selection to treatment, $T_{assign} \mid \mathbf{X}$. This distinction also further helps to illustrate that at the most granular level we are truly interested in

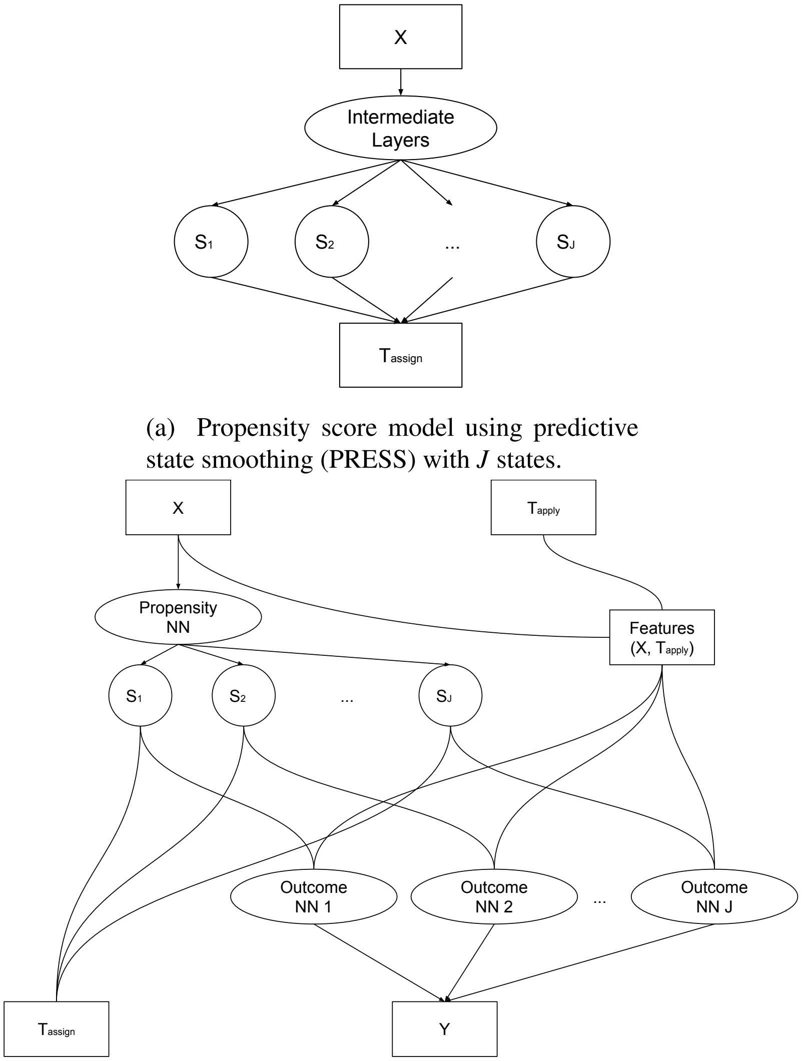
$$p_{\theta_Y}(Y_i \mid T_{apply} = t_i, \mathbf{X} = \mathbf{x}_i), \tag{2}$$

that is the effect of *applying* treatment t on unit i with observed pre-treatment covariates $\mathbf{x}_i$.

2.2 Predictive States

PSPS puts the goal of constructing these subclasses front and center and estimates the optimal stratification directly from the observed data. It relies on Predictive State Smoothing (PRESS) (Goerg, 2017, 2018) to estimate minimal sufficient subclasses. PRESS models the conditional predictive distribution $p(T \mid \mathbf{X})$ as a mixture over J latent predictive states, $\{s_1(\mathbf{X}), \ldots, s_J(\mathbf{X})\}$ (Figure 1a). Each state represents an equivalence classes over distribution space such that observations with the same conditional distribution are in the same predictive state. That is $\mathbf{X}_{i_1}$ and $\mathbf{X}_{i_2}$ have the same state s_j if and only if

$$p(T \mid \mathbf{X}_{i_1}) \equiv p(T \mid \mathbf{X}_{i_2}) =: p(T \mid s_j). \tag{3}$$



(a) Propensity score model using predictive state smoothing (PRESS) with J states.

(b) PSPS as a multi-input multi-output neural network. Note that $T_{apply} \equiv T_{assign} \equiv T$ but the dual role in (1) is made explit here.

Figure 1: Overview of the components of a PSPS model: propensity model, $P(T \mid \mathbf{X})$, an outcome model in a given state, $P(Y \mid \mathbf{X}, S = S_j)$, and the full joint model $P(Y, T \mid \mathbf{X})$, where $\mathbf{X}$ are covariates, T is the treatment, and Y is the outcome.

The key insight here is that predictive states are exactly the strata $\{s_1, \ldots, s_J\}$ that would have ideally been blocked on when assigning treatment: knowing the predictive state of observation i – which is unobserved in a natural experiment –would be sufficient to knowing the propensity score distribution of all observations in that state.

The states are also minimal sufficient statistics for prediction with the property that they make T conditionally independent of $\mathbf{X}$ given s_j.[1] For the propensity model this yields

$$p(T \mid \mathbf{X}) = \sum_{i=1}^{J} p(S = s_j \mid \mathbf{X}) \cdot p(T \mid S = s_j, \mathbf{X}) \tag{4}$$

$$= \sum_{i=1}^{J} w_j(\mathbf{X}) \cdot p(T \mid S = s_j) \tag{5}$$

where (5) follows from the construction of predictive states as T and $\mathbf{X}$ become conditionally independent given s_j.

Let $w_j(\mathbf{X}) := p(S = s_j \mid \mathbf{X})$ denote the probability that $\mathbf{X}$ belongs to state s_j. We use the weight-vector notation $\mathbf{w} = (w_1, \ldots, w_J)$ as the representation of the state space mapping; by definition $\sum_{j=1}^{J} w_j = 1$ and $w_j \geq 0$. It is easy to see that any two observations i_1 and i_2 – with potentially different covariates $\mathbf{X}_{i_1}$ and $\mathbf{X}_{i_2}$ – that are mapped to the same predictive state with same probabilities, i.e., $\mathbf{w}_{i_1} = \mathbf{w}_{i_2}$, necessarily have the same distribution of treatment since

$$p(T_i \mid \mathbf{X}_i) = \sum_{j=1}^{J} w_{i,j} \cdot p(T_i \mid s_j). \tag{6}$$

Put in other words, if i_1 and i_2 have different observed covariate values, $\mathbf{x}_{i_1}$ and $\mathbf{x}_{i_2}$, but are in the same state, s_j, then they have the same propensity score:

$$P(T_i \mid \mathbf{X} = \mathbf{x}_{i_1}, S = s_j) = P(T_i \mid S_j) = P(T_j \mid \mathbf{X} = \mathbf{x}_{i_2}, S = s_j). \tag{7}$$

This is exactly the goal of the original subclassification method by Imbens and Rubin (2015): group observations that have similar propensity scores.

Another way of thinking about it is that each state is a matched group where we are matching on an optimal distance between units i_1 and i_2 with covariates $\mathbf{X}_{i_1}$ and $\mathbf{X}_{i_2}$ measured as the distance between propensity scores $P(T_{i_1} \mid \mathbf{X}_{i_1})$ and $P(T_{i_2} \mid \mathbf{X}_{i_2})$.

After constructing the subclasses, PSPS follows the traditional subclassification philosophy of fitting an outcome model in each subclass (Figure 1b). This has the effect of adjusting for covariates if the PRESS propensity model is misspecified. Even when it is correctly specified, more accurate outcome models can further reduce the variability in causal effect estimates (Rosenbaum, 2002).

[1] We refer to the original PRESS papers for details.

2.3 Combining Outcome Models Across Predictive States

To replicate the behavior of estimating the causal effect within each subclass we use the predictive states from the propensity model to factorize the conditional outcome distribution, $P(Y \mid \mathbf{X}, T)$. In particular, by averaging over predictive states the first term in (1) becomes

$$\begin{aligned} p(Y_i \mid T_i, \mathbf{X}_i) &= \sum_{j=1}^{J} p(S = s_j \mid \mathbf{X}_i, T_i) \times p(Y_i \mid T_i, \mathbf{X}_i, S = s_j) && (8) \\ &= \sum_{j=1}^{J} \frac{p(T_i \mid S = s_j)}{p(T_i \mid \mathbf{X}_i)} \cdot P(S = s_j \mid \mathbf{X}_i) \times p_{\theta_Y^{(j)}}(Y_i \mid T_i, \mathbf{X}_i), && (9) \\ &= \sum_{j=1}^{J} (\rho_{i,j} \cdot w_{i,j}) \times p_{\theta_Y^{(j)}}(Y_i \mid T_i, \mathbf{X}_i) && (10) \end{aligned}$$

where the lift ratio $\rho_{i,j} \in [0, \infty)$ quantifies how much the state predictive distribution, $p(T_i \mid s_j)$, differs from the point prediction of the propensity score for feature $\mathbf{x}_i$, $p(T_i \mid \mathbf{X}_i = \mathbf{x}_i)$, and $\theta_Y^{(j)}$ parameterizes the outcome model in state j. Since $\rho_{i,j} \cdot w_{i,j} = P(S = s_j \mid, \mathbf{X}_i, T_i)$ is a proper probability distribution for each i, the decomposition in (10) is in fact a mixture over state-conditional outcome models, $p(Y \mid \mathbf{X}, T)$, where model j contributes with weight $v_{i,j} := \rho_{i,j} \cdot w_{i,j}$ to the final estimate for unit i. That means observation i is only partially contributing to state j and also explains why, in general, outcome model parameter estimates will be different for every state.[2]

Note that if each unit i is mapped exactly to one state with probability 1 (i.e., a deterministic mapping), then also $\rho_{i,j} = 1$ for exactly one j and 0 otherwise. In that case,

$$p(Y_i \mid T_i, \mathbf{X}_i) = \sum_{j=1}^{J} p_{\theta_Y^{(j)}}(Y_i \mid T_i, \mathbf{X}_i, S = s_j) \mathbb{1}\left(S = s_j\right)_{\mathbf{X}_i}, \tag{11}$$

which is equivalent to the traditional procedure of binning propensity scores and fitting an outcome model in each bin.

2.4 Parameter Estimation via Multi-Task Learning

Since PSPS is a fully probabilistic framework we can use maximum likelihood estimators (MLE) or Bayesian methods for inference. For the scope of this work we focus on MLE and leave Bayesian inference for future work.

[2] In general, each state could be equipped with a different model class, e.g., a Random Forest in state j_1, a linear model in state j_2, and a deep net in j_3. In practice, however, they are usually of the same model family – with different (estimated) parameters per state.

Decomposing the joint distribution in (1) makes parameter estimation much more tractable as it allows us to specify propensity model and outcome models separately, yet combine them in a joint log-likelihood function as

$$\ell(\theta_Y, \theta_T; Y, \mathbf{X}, T) = \ell(\theta_Y; Y, \mathbf{X}, T) + \ell(\theta_T; \mathbf{X}, T), \tag{12}$$

where $\ell(\tau; Z) := \log p_\tau(Z)$ is the log-likelihood of τ for data Z. This decomposition naturally leads to framing parameter estimation using a multi-task learning problem (Ruder, 2017) with a negative log-likelihood loss in a multi-output neural net:

$$\begin{aligned} L(Y, T, \hat{Y}(\theta_Y, \theta_T), \hat{T}(\theta_T)) &= L(Y, \hat{Y}(\theta_Y, \theta_T)) + L(T, \hat{T}(\theta_T)), \\ \text{joint loss} &= \text{outcome model loss} + \text{propensity model loss}, \end{aligned} \tag{13}$$

where $\hat{T}(\theta_T)$ is the predicted probability of treatment (propensity score) and $\hat{Y}(\theta_Y, \theta_T)$ is the predicted outcome. Therefore, the MLE can be obtained by solving

$$\widehat{(\theta_Y, \theta_T)}_{MLE} = \arg\min_{\theta_Y, \theta_T} L(Y, T, \hat{Y}(\theta_Y, \theta_T), \hat{T}(\theta_T)). \tag{14}$$

2.5 Separation of Design and Analysis

In observational studies, practitioners often face a choice between only fitting an outcome model ($P(Y \mid T, \mathbf{X})$) or a two-step procedure of fitting a propensity model first ($P(T \mid \mathbf{X})$) and an adjusted/weighted outcome model later. We now show that PSPS includes both approaches as special cases.

First note that the log-likelihood decomposition in (12) can be seen as trying to find an optimal outcome model, but with a penalty for bad propensity models. Taking this view, it is natural to introduce a penalty parameter, $\lambda \in [0, \infty]$,

$$L(Y, \hat{Y}(\theta_Y, \theta_T)) + \lambda \cdot L(T, \hat{T}(\theta_T)), \tag{15}$$

which reduces to the original joint model in (13) when $\lambda = 1$. This has a nice interpretation in that we are just fitting a single outcome model $Y \mid T, \mathbf{X}$, with a penalty $L(T, \hat{T}(\theta_T))$ that adjusts for imbalances in the covariate distribution between treated and controls. This is also aligned with recent work in the deep learning literature on representation learning for causal inference (Johansson et al., 2016).

When $\lambda = 0$ the second term vanishes from (15) and the optimizer only focuses on outcome models, weighted by a random mixture of predictive state weights from each state, which – in expectation – is just a simple overall outcome model. On the other extreme, for $\lambda \to \infty$ optimizing (15) approximates the standard two-step approach that separates design and analysis stages of an experiment: since $\lambda \to \infty$ the optimization first tries to obtain optimal propensity score, and only then it will further tune the outcome model parameters to improve total loss (at the given optimal propensity score loss).

2.6 Parameter Estimation

In this section we describe estimation for randomized or natural experiments with binary treatment and a continuous univariate outcome. It is this model that we evaluate through various simulation studies in Section 4.

For binary treatment the negative log-likelihood for observation i equals

$$L(T_i, \hat{T}_i(\theta_T)) = -\left[T_i \cdot \log(\hat{T}_i(\theta_T)) + (1 - T_i) \cdot \log(1 - \hat{T}_i(\theta_T))\right]. \tag{16}$$

For univariate continuous outcomes a Normal log-likelihood, $N(y \mid x, \sigma_\varepsilon^2)$, is a natural candidate leading to a weighted sum of outcome losses for each state:

$$L(Y_i, \hat{Y}_i(\theta_Y); \theta_T, \mathbf{X}_i) = -\log(\sigma_\varepsilon) + \frac{1}{2\sigma_\varepsilon^2} \sum_{j=1}^{J} v_{i,j}(\mathbf{X}_i, \theta_T) \cdot (Y_i - \hat{Y}_i(\theta_{Y,j}))^2, \tag{17}$$

where J is the number of propensity states, $v_{i,j}(\mathbf{X}_i, \theta_T)$ is the probability of unit i belonging to state j, $\hat{Y}_i(\theta_Y)$ is the outcome model prediction for Y_i, and $\theta_Y = \sigma_\varepsilon \cup \{\theta_j \mid j = 1, \ldots, J\}$ are all free parameters of the outcome model, with $\sigma_\varepsilon \in \theta_Y$ as the unknown standard deviation of residual, $Y_i - \hat{Y}_i$. The total loss across N observations equals

$$\sum_{i=1}^{N} L(Y_i, \hat{Y}_i(\theta_Y); \theta_T, \mathbf{X}_i) - \sum_{i=1}^{N} \left[T_i \cdot \log(\hat{T}_i(\theta_T; \mathbf{X}_i)) + (1 - T_i) \cdot \log(1 - \hat{T}_i(\theta_T; \mathbf{X}_i))\right]. \tag{18}$$

It is important to point out that the PRESS propensity model parameters, θ_T, do not only enter the loss via the propensity model loss, but also affect the outcome loss through the weighting over the J states, $v_{i,j}(\mathbf{X}_i, \theta_T)$.

The estimation procedure to optimize (18) can be described as follows:

1. Train PRESS model $T \sim \mathbf{X}$ with J predictive states.

2. Map observation i to their state with probability $w_{i,j} = \hat{p}(s_j \mid \mathbf{X}_i)$; compute adjusted weights by further conditioning on T, $v_{i,j} = \hat{p}(s_j \mid \mathbf{X}_i, T_i)$.

3. In each state, $s_j, j = 1, \ldots, J$ fit an outcome model for $Y \mid \mathbf{X}, T$, where observation i is weighted by $v_{i,j}$ from 2. The outcome models could be a neural net, a regression, or even

just the sample mean of the response; $Y \mid \mathbf{X}, T = 1$ and $Y \mid \mathbf{X}, T = 0$ can also be modeled separately.

Note though that since the entire PSPS model can be expressed as a single computational graph using standard neural network architectures, we can estimate all parameters and predictive states jointly rather than relying on the – statistically and computationally – inefficient iterative procedure above.

3 Estimating Causal Effects

Once the predictive state mapping and all outcome models have been estimated, PSPS provides unit-level treatment effect (UTE) estimates as the aggregation of UTE within each state, weighted by the size of each state relative to the entire sample:

$$\begin{aligned}\hat{\Delta}_i &= \sum_{j=1}^{J} \hat{v}_{i,j} \cdot \hat{\Delta}_i(j) \\ &= \sum_{j=1}^{J} \mathrm{P}(s_j \mid t_i, \mathbf{x}_i; \hat{\theta}_T) \cdot \left(\mathbb{E}(y \mid \mathbf{x}_i, t_i = 1; \hat{\theta}_j) - \mathbb{E}(y \mid \mathbf{x}_i, t_i = 0; \hat{\theta}_j)\right),\end{aligned} \tag{19}$$

where $\hat{\Delta}_i(j)$ is the estimated treatment effect for unit i from outcome model j. The overall UTE estimate is a weighted average, where the weights are determined by the probability of unit i being in state j given treatment and pre-treatment covariates.

Estimates for sample, population and conditional treatment effects can be obtained by simply aggregating over the appropriate (sub-)sample of observations. For example,

$$\hat{\Delta}_{\mathrm{ATE}} = \frac{1}{N} \sum_{i=1}^{N} \hat{\Delta}_i \tag{20}$$

and

$$\hat{\Delta}_{\mathrm{ATT}} = \frac{1}{\sum_{i=1}^{N} T_i} \sum_{i=1}^{N} T_i \hat{\Delta}_i \,. \tag{21}$$

3.1 Deterministic Assignment to States

Without any restrictions on $w_{i,j}(\mathbf{X})$ the method described above is not exactly the same as traditional subclassification, since observations are mapped to a state with a certain probability rather than deterministically. In that sense PSPS is a soft-thresholding generalization of the classic subclassification approach – akin Gaussian mixture modeling being the soft-thresholding generalization of K-means clustering.

If practitioners want to obtain such a deterministic (hard-clustering) strata assignment, we suggest to add an entropy based penalty regularization of the form

$$\lambda_e \cdot \sum_{i=1}^{N} \left(- \sum_{j=1}^{J} w_{i,j} \log_2 w_{i,j} \right) = \lambda_e \cdot \sum_{i=1}^{N} \text{entropy}_i. \tag{22}$$

Adding this penalty to the total loss in (18) allows us to control with how much certainty observations are mapped to predictive states. For sufficiently large λ_e the penalty is minimized for deterministic mappings, $w_{i,j} = 0$ for all but one j for each i, i.e., it effectively assigns observations to exactly one state with probability 1.

We want to emphasize though that without any prior knowledge of such a deterministic mapping we suggest to keep the soft-thresholding property of PRESS, as from a pure likelihood point of view it leads to better models. Similarly to how a Gaussian mixture model usually gives better quality results compared to a hard-thresholding K-means solution. In practice a small λ_e can be beneficial to make $\mathbf{w}_j$ slightly more sparse.

3.2 Trimming

A nice consequence of using predictive states in the propensity model is that we can easily assess balance within each predictive state between treated and control and then drop states from the inference where balance is deemed inadequate. A simple suggestion is to ensure that there are enough treated and control units in each state with respect to the complexity of the outcome model being assumed. Other measures may include assessing balance of individual covariates or their products via methods such as t-tests, etc.

3.3 TensorFlow Implementation

Modeling the propensity and outcome models jointly – rather than traditional two-step procedures – has the benefit that the model can be expressed as a single computational graph in deep learning frameworks. In particular, our implementation for MLE optimization and treatment effect inference runs entirely within a TensorFlow graph computation (Abadi et al., 2016), which in turn can be easily added into larger graphs that rely on causal inference as part of their computation.

3.4 Model Complexity and Parameter Tuning

Recall that PSPS is based on a single PRESS model that approximates $P(T \mid \mathbf{X})$ using J states; PSPS then trains outcome models in each of the J states. There are no restrictions on the complexity of the function spaces to approximate $P(Y \mid \mathbf{X}, T)$ and $P(T \mid \mathbf{X})$, respectively. One thus faces a trade-off to approximate $p(T \mid \mathbf{X})$ well (increase J, larger model class) but at the same time not let J grow too much as the full joint model otherwise contains too many parameters $(\theta_1, \ldots, \theta_J)$ to estimate consistently and efficiently with limited amount of data.

While only a model selection / tuning algorithm can find a value for J and guarantee reasonable model complexity choices for the dataset at hand, we can leverage theory on doubly robust methodology (Robins and Rotnitzky, 1995) to give a generally applicable recommendation for this trade-off. Double robustness states that as long as at least one of the propensity or the outcome model is correctly specified, the causal effects can be estimated consistently – even if one of the models is not misspecified. From that point of view, it is clearly preferable to get *one* propensity model right, rather than trying to specify *all* J outcome models correctly. This aligns with our findings in simulations that a more complex propensity model (e.g., a neural net with several layers before the predictive states) combined with simple (e.g., linear) outcome models is sufficient to get unbiased, low-variance causal effect estimators.

In practice we found that shallow neural networks with standard activation functions, such as *selu* or *relu*, work well for a wide range of datasets (see Goodfellow et al., 2016, for details and further references).

If the sample size is large enough, we do recommend to use non-linear function spaces for outcome models as well, as this can even further reduce the variance in treatment effect estimates.

3.5 Uncertainty Estimates

We use standard bootstrap procedures (Efron and Tibshirani, 1986) to obtain empirical confidence intervals for Δ_i and Δ_{ATE} in (19) and (20), respectively. See Figure 2b, 3b, and 4b for interval width and coverage metrics.

4 Simulations

Here we compare PSPS (with $J = 10$) to several state of the art causal inference methods. In particular we consider linear outcome models, doubly robust inverse propensity weighted (DRIPW) models (Chan et al., 2010) – both linear and Random Forest (Breiman, 2001) versions –, entropy balancing (Hainmueller, 2012), and a neural net outcome model (PRESS from (Goerg, 2017)).[3]

[3] For clarification, the PRESS neural net model is not related to the PRESS setup that PSPS uses for the propensity model. It is simply an outcome neural net for $P(Y \mid Z)$ with $Z = (\mathbf{X}, T)$ as the covariates, which happens to use the PRESS structure. The states from this outcome-only model are not related to the predictive states of the

All of the above models – with the exception of entropy balancing – can not only estimate the average treatment effect (ATE), but also unit-level treatment effects (UTE) (this is equivalent to the conditional average treatment effect (CATE) in our setup). To compare the different methods we conduct a simulation study and examine the distribution of the biases for ATE and RMSE for UTE (where appropriate) across the different simulations for each of the different methods.

The results are graphed in Figures 2, 3 and 4 where each point in the graph is a run from a single simulated dataset. For estimating ATE we plot the distribution of biases for each of the simulated datasets for each method for each simulation scenario. By plotting the full distribution we can not only compare the location of the biases but also their spread, where a method with the same average or median bias but smaller variance is preferred. We chose to examine the distribution of RMSE for estimating UTEs as it is a single measure summarizing the performance for a single dataset. This was chosen in lieu of the average bias as by construction the sample mean of the estimated UTEs for each dataset in the simulation is the estimated ATE for that dataset (see equation 20) and so the average bias is equivalent to the bias for the ATE.

4.1 Kang & Schafer

Kang and Schafer (2007) construct a simulation based on one of their real case studies. Here four features are drawn independently from a standard Normal, $(z_1, z_2, z_3, z_4) \sim N(0, I_4)$, and the outcome and true propensity scores are generated as

$$y = 210 + 27.4z_1 + 13.7z_2 + 13.7z_3 + 13.7z_4 + \varepsilon \tag{23}$$

and

$$\pi = \text{expit}(-z_1 + 0.5z_2 - 0.25z_3 - 0.1z_4) \tag{24}$$

respectively, where $\varepsilon \sim N(0,1)$. The true features (z_1, z_2, z_3, z_4) are then transformed as

$$\begin{aligned} x_1 &= \exp(z_1/2) \\ x_2 &= x_2/(1+\exp(z_1)) + 10 \\ x_3 &= (z_1 z_3/25 + 0.6)^3 \\ x_4 &= (z_2 + z_4 + 20)^2 \end{aligned} \tag{25}$$

such that causal methods can be evaluated against correctly and incorrectly specified models by either using the original or transformed features. This can help demonstrate doubly robustness

propensity model in PSPS.

but also can help test feature generation and exploration that some methods utilize such as PSPS, PRESS and DRIPWRandomForest.

Figure 2a summarizes the results on bias & RMSE. In the "Original" (linear) case, DoublyRobustIPW, EntropyBalancing and PSPS are all unbiased; the linear outcome model is slightly biased but has the best RMSE for UTE estimates alongside the PSPS model. It is not surprising that both parametric linear causal models (DoublyRobustIPW and EntropyBalancing) perform best in terms of bias as they can be considered best-case oracle scenarios. Interestingly, PSPS is on par with the parametric & linear models even though it is a semi-parametric, non-linear method. PSPS especially shows its strength compared to the misspecified linear models in the "Transformed" case, where Entropy Balancing does significantly worse in terms of bias, and DoublyRobustIPW has the worst UTE estimates amongst all models (EntropyBalancing does not even provide UTE estimates). PSPS on the contrary is only slightly biased, but has the smallest RMSE for UTE effects amongst all models. Given that in real world settings, cause and effect are rarely – if ever – related in a linear way, this comes to show that PSPS is a great choice for estimating causal effects in real world – non-linear – scenarios. Figure 2b shows that only PSPS, EntropyBalancing, and DoublyRobustIPW have close to proper confidence interval coverage with small widths. However, all methods except for DoublyRobustIPW are completely off in terms of coverage for the "Transformed" case. Achieving correct coverage in the non-linear case remains a task for future work.

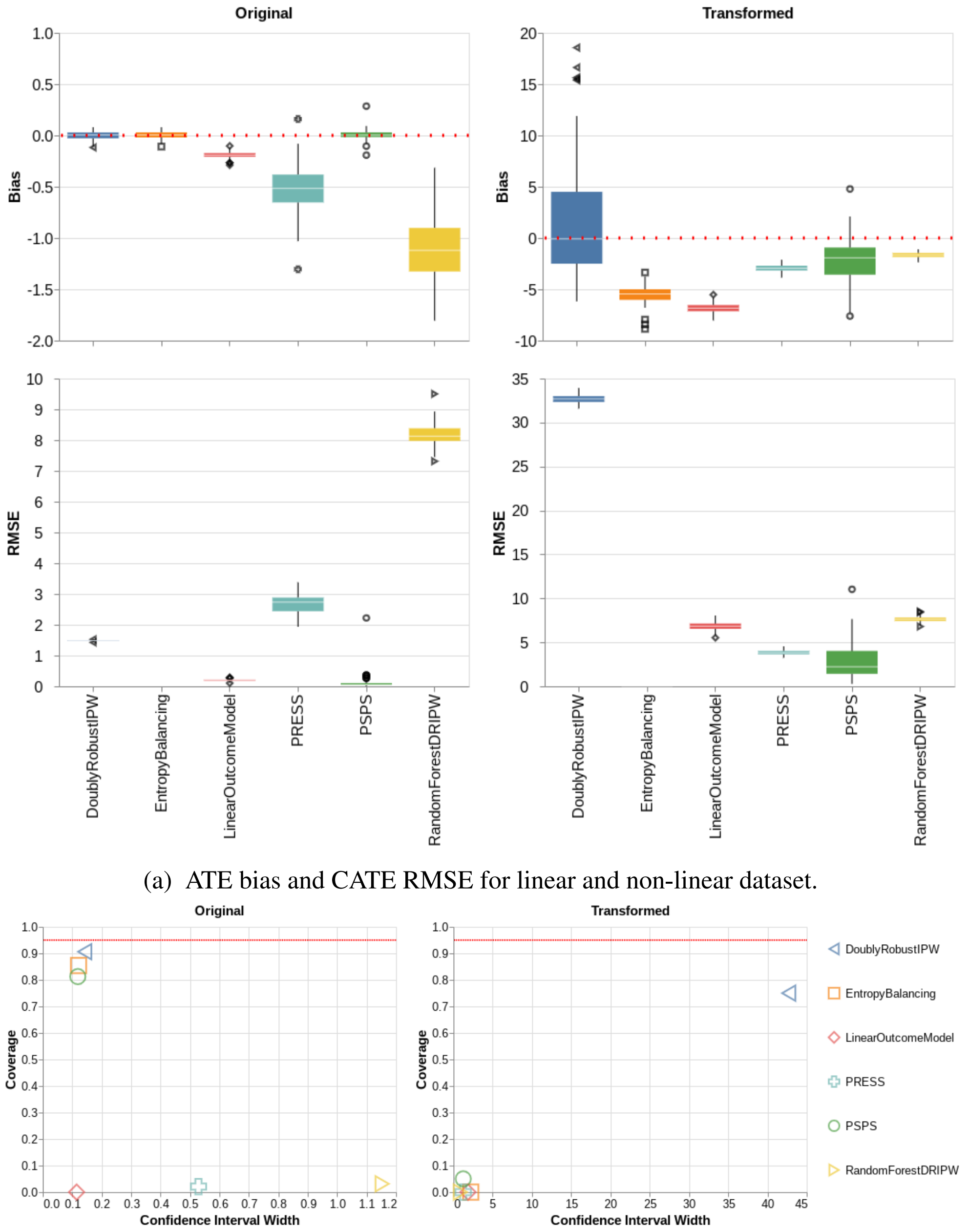
4.2 Lunceford & Davidian

Lunceford and Davidian (2004) provide another simulation where the data still consists of $\{(\mathbf{X}_i, \mathbf{Z}_i, T_i, Y_i), i = 1, \cdots, n\}$, but all of them are observed. Both $\mathbf{X}_i$ and $\mathbf{Z}_i$ are three dimensional vectors. The propensity score is only related to $\mathbf{X}$ – not $\mathbf{Z}$ – through

$$\operatorname{logit}(P(T_i = 1)) = \beta_0 + \sum_{j=1}^{d} \beta_j \mathbf{X}_{i,j}. \tag{26}$$

The parameter β controls the association between $\mathbf{X}$ and T. We follow the original work and use three settings for "no", "moderate", and "strong" association:

$$\begin{aligned} \beta^{no} &= (0, 0, 0, 0)^T, \\ \beta^{moderate} &= (0, 0.3, -0.3, 0.3)^T, \\ \beta^{strong} &= (0, 0.6, -0.6, 0.6)^T. \end{aligned} \tag{27}$$



(a) ATE bias and CATE RMSE for linear and non-linear dataset.

(b) 95% bootstrap confidence interval coverage and interval width

Figure 2: Model comparison for Kang & Schafer dataset (Section 4.1)

The response Y is generated according to

$$Y_i = \nu_0 + \sum_{j=1}^{3} \nu_j \mathbf{X}_{ij} + \nu_4 T_i + \sum_{j=1}^{3} \xi_j Z_{ij} + \varepsilon_i, \tag{28}$$

where $\varepsilon_i \sim N(0,1)$ and $\nu = (0,-1,1,-1,2)^T$. Similarly to β, ξ controls the association between $\mathbf{Z}$ and Y:

$$\begin{aligned} \xi^{no} &= (0,0,0)^T, \\ \xi^{moderate} &= (-0.5,0.5,0.5)^T, \\ \xi^{strong} &= (-1,1,1)^T. \end{aligned} \tag{29}$$

The joint distribution of $(\mathbf{X}_i, \mathbf{Z}_i)$ is given by $\mathbf{X}_{i3} \sim \text{Bernoulli}(0.2)$, then generate $\mathbf{Z}_{i3}$ as Bernoulli with

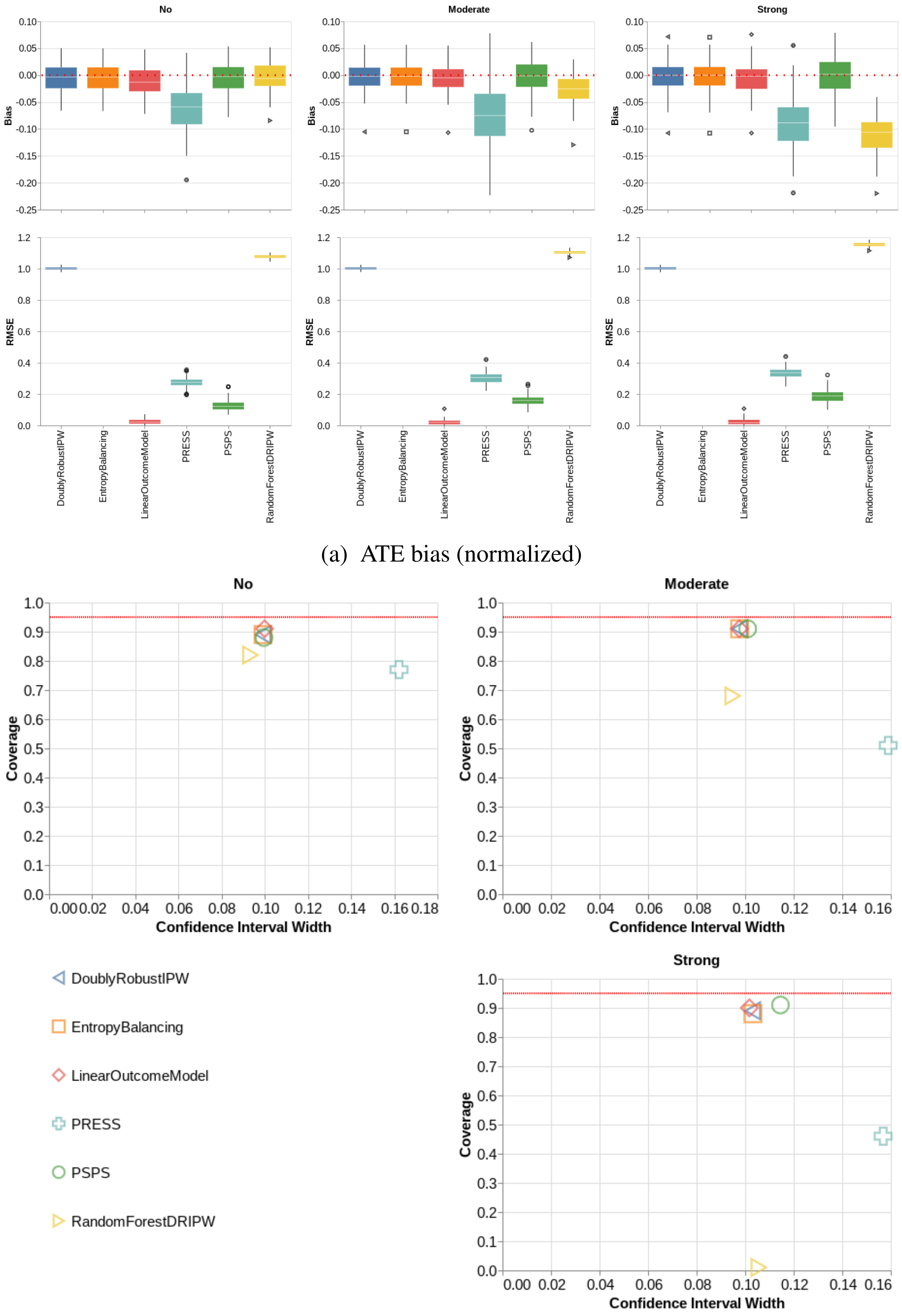
$$P(\mathbf{Z}_{i3} = 1 \mid \mathbf{X}_{i3}) = 0.75 \cdot \mathbf{X}_{i3} + 0.25 \cdot (1 - \mathbf{X}_{i3}). \tag{30}$$

Conditional on $\mathbf{X}_{i3}$, $(\mathbf{X}_{i1}, \mathbf{Z}_{i1}, \mathbf{X}_{i2}, \mathbf{Z}_{i2})$ is then generated as multivariate normal $N(a_{\mathbf{X}_{i3}}, B_{\mathbf{X}_{i3}})$, where $a_1 = (1,1,-1,-1)^T, a_0 = (-1,-1,1,1)^T$ and

$$B_0 = B_1 = \begin{pmatrix} 1 & 0.5 & -0.5 & -0.5 \\ 0.5 & 1 & -0.5 & -0.5 \\ -0.5 & -0.5 & 1 & 0.5 \\ -0.5 & -0.5 & 0.5 & 1 \end{pmatrix}. \tag{31}$$

This data generation process yields a true ATE of 2.0. The challenge comes from the inter- and intra-dependence of $\mathbf{X}$ and $\mathbf{Z}$, and that $\mathbf{Z}$ is only involved in the outcome model.

In Figure 3 we see that PSPS, linear DRIPW, entropy balancing and DRIPW perform equally well in all cases. As in the Kang & Schafer case this is quite compelling for PSPS as it is on par with the parametric linear models, while being able to model non-linearities as in a semi-parametric fashion. The neural net outcome model underestimates ATE in all cases – most likely due to a regression to the mean property (shrinkage) of PRESS' inherent smoothing. The bias of a Random Forest outcome model increases for stronger associations. Interestingly, both a linear and Random Forest DRIPW model cannot estimate unit-level effects – compared to competing models that all have low RMSE.



(a) ATE bias (normalized)

(b) 95% bootstrap confidence interval coverage and interval width

Figure 3: Model comparison for Lunceford & Davidian dataset (Section 4.2)

4.3 Radcliffe & Surrey

This data generating process is taken from Radcliffe (2007) and Radcliffe and Surry (2011). Here treatment is randomly assigned to each unit with probability p. For each unit, covariates x_1 and x_2 are drawn as independent uniform random integers in $U[0,B]$ for an upper bound B (here: $B = 99$). Unobserved variables u_1 and u_2 are then drawn uniformly from $U(0,x_i)$, $i = 1,2$ respectively. Unit-level treatment effects and outcome regressions are then set as

$$ute_i = u_{2,i} \cdot s + c \tag{32}$$

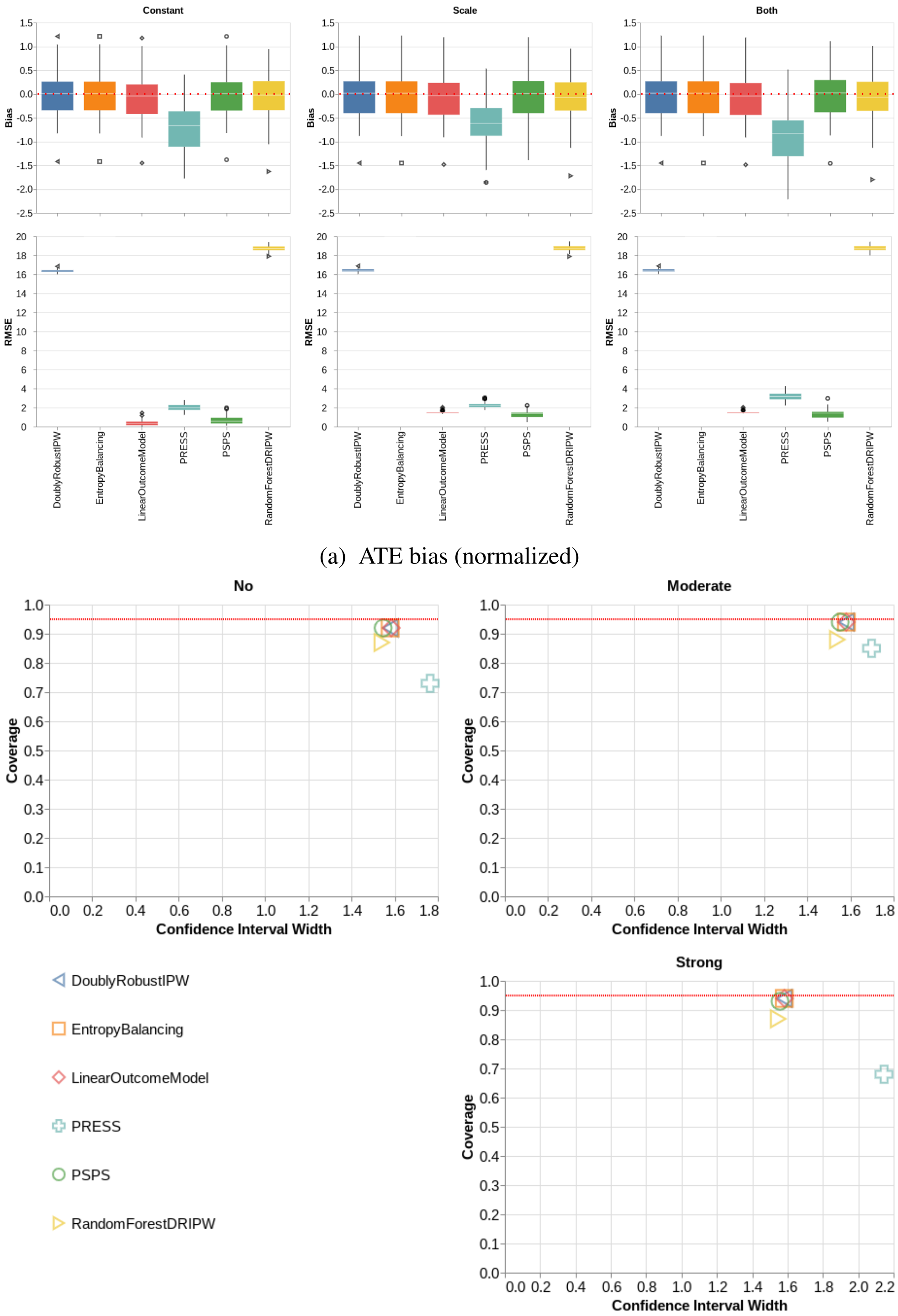
$$y_i = u_{1,i} + T_i \cdot ute_i, \tag{33}$$

where T_i is the realization of a Bernoulli draw with probability p (here: $p = 0.5$). This specification yields a true ATE of

$$ATE = s \cdot \mathbb{E}(u_2) + c, \text{ where } \mathbb{E}(u_2) = B/4. \tag{34}$$

In simulations we explore whether the scaling factor s or shift c affect the ability to recover the true treatment effect. As for Lunceford & Davidian we consider three scenarios: $const = \{s = 0, c = 3\}$, $scale = \{s = 0.1, c = 0\}$, and $both = \{s = 0.1, c = 3\}$.

Figure 4 shows that – except for the outcome PRESS neural net – all models give unbiased ATE estimates; and again, linear and Random Forest DRIPW cannot estimate unit-level effects at all. Results do not vary much between “const”, “scale”, or “both” except for RandomForest-DRIPW, which performs worse as the association grows stronger.



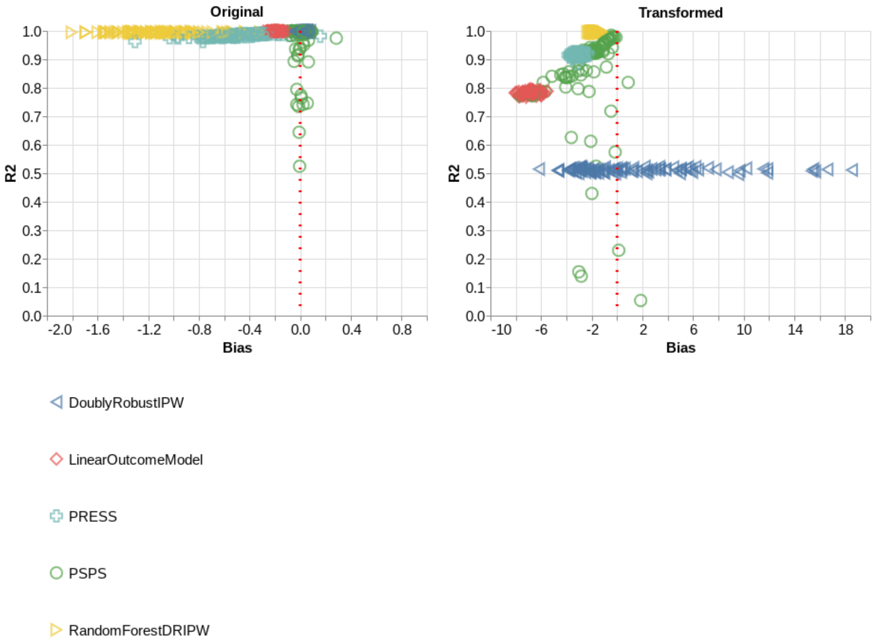
(a) ATE bias (normalized)

(b) 95% bootstrap confidence interval coverage and interval width

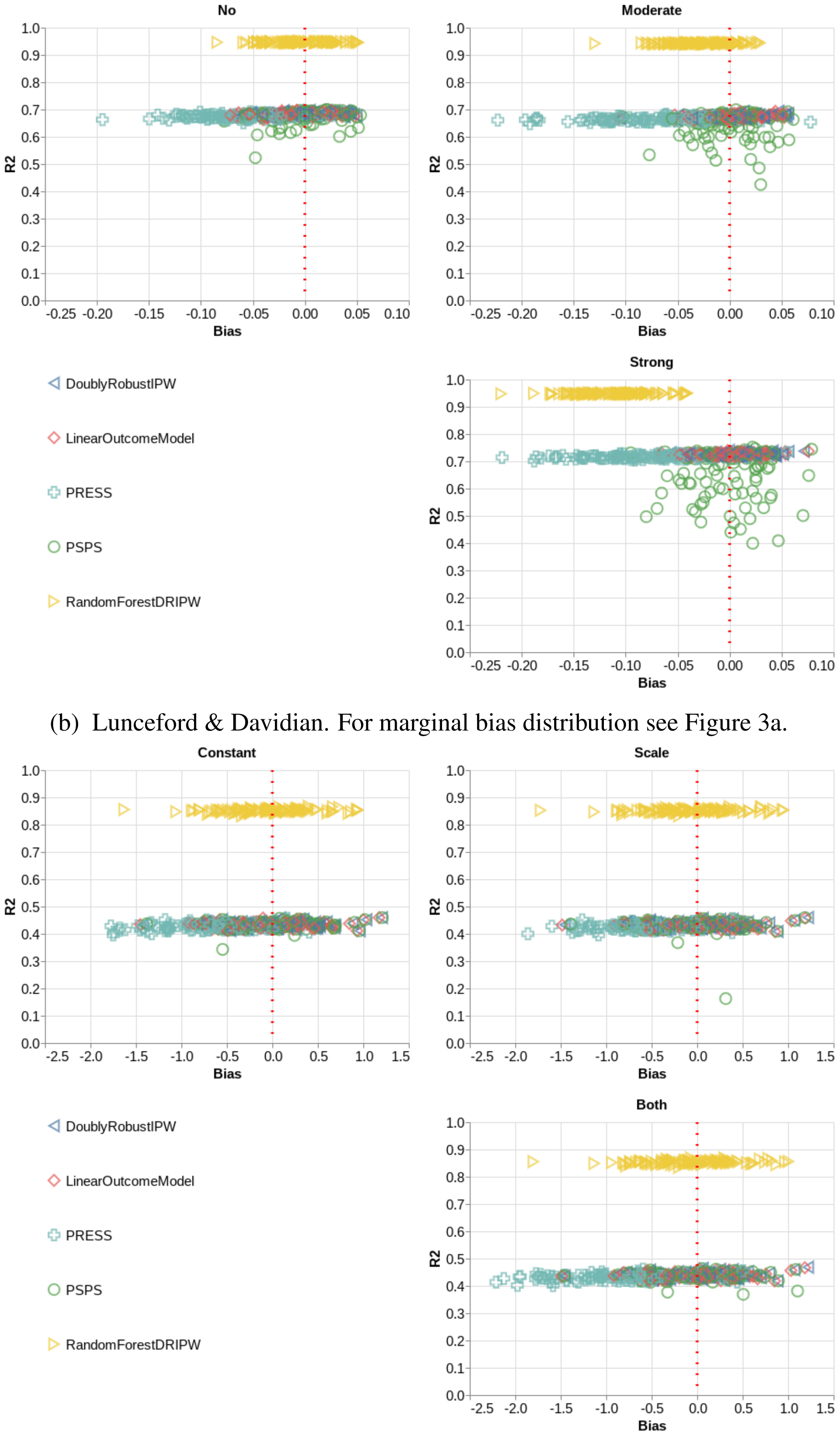
Figure 4: Model comparison for Radcliffe & Surrey dataset (Section 4.3)

4.4 Balancing the Outcome and Treatment Models

In Figure 5 we plot the R-squared of the underlying outcome model versus the achieved bias of the ATE for each dataset (with the exception of EntropyBalancing which does not assume a separate predictive outcome model). We see that those methods with a higher R-squared do not necessarily achieve a better bias reduction for estimating the Average Treatment Effect. This is demonstrated in Figure 5a where we see that PSPS often achieves zero bias with a lower R-squared than other methods. It is also evident in Figure 5b where we see that even though RandomForestDRIPW had the largest R-squared value it did not achieve lowest bias. These results highlight the fact that in causal inference the objective is to estimate the treatment effects and not to achieve greatest predictive accuracy on the outcome. They show that by not just focusing on the predictive quality of an outcome model but by correcting imbalances among the treatment groups a better estimate of the average treatment effect can be achieved.



(a) Kang & Schafer. For marginal bias distribution see Figure 2a.



(b) Lunceford & Davidian. For marginal bias distribution see Figure 3a.

(c) Radcliffe & Surrey. For marginal bias distribution see Figure 4a.

Figure 5: R-squared of the Outcome Model vs. ATE Bias

5 Discussion

We introduce *predictive state propensity subclassification* (PSPS), a novel method for causal inference. It follows the general Rubin Causal Model (RCM) framework by adjusting natural experiments through propensity models following a subclassification approach. However, rather than using ad-hoc procedures to bin observations by propensity score, PSPS estimates outcome models, propensity models, and optimal strata simultaneously as part of a joint probabilistic model. This in turn yields efficient parameter estimation procedures that can be easily implemented in deep learning frameworks like TensorFlow. We show via simulations on standard causal inference datasets that PSPS is either unbiased or modestly biased for a wide range of data generating processes, small confidence intervals width, yet with accurate nominal coverage outperforming state-of-the-art causal inference algorithms.

Acknowledgments

We want to thank Penny Chu, Tony Fagan, and Jim Koehler for their ongoing support during this project. We are especially grateful to Mike Hankin, Tianqi Liu, Susanna Makela, Xiaojing Wang, and Mike Wurm for fruitful discussions on the general methodology and their contributions to run all large-scale simulations and evaluations we presented in this work.

References

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. J. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Józefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. G. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. A. Tucker, V. Vanhoucke, V. Vasudevan, F. B. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467, 2016. URL http://arxiv.org/abs/1603.04467.

S. Bernstein. Sur l'extension du théorème limite du calcul des probabilités aux sommes de quantités dépendantes. *Mathematische Annalen*, 97(1):1–59, 1927.

L. Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, Oct. 2001. ISSN 0885-6125. doi: 10.1023/A:1010933404324. URL https://doi.org/10.1023/A:1010933404324.

D. Chan, R. Ge, O. Gershony, T. Hesterberg, and D. Lambert. Evaluating online ad campaigns in a pipeline: Causal models at scale. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 7–16, New York, NY, USA, 2010. ACM. ISBN 978-1-4503-0055-1. doi: 10.1145/1835804.1835809. URL http://doi.acm.org/10.1145/1835804.1835809.

W. G. Cochran. The effectiveness of adjustment by subclassification in removing bias in observational studies. *Biometrics*, pages 295–313, 1968.

R. H. Dehejia and S. Wahba. Causal effects in nonexperimental studies: Reevaluating the evaluation of training programs. *Journal of the American Statistical Association*, 94(448):1053–1062, 1999. doi: 10.1080/01621459.1999.10473858. URL https://www.tandfonline.com/doi/abs/10.1080/01621459.1999.10473858.

B. Efron and R. Tibshirani. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statist. Sci.*, 1(1):54–75, 02 1986. doi: 10.1214/ss/1177013815. URL https://doi.org/10.1214/ss/1177013815.

G. M. Goerg. Predictive State Smoothing (PRESS): Scalable non-parametric regression for high-dimensional data with variable selection. Technical report, Google, 2017. URL https://ai.google/research/pubs/pub46141.

G. M. Goerg. Classification using Predictive State Smoothing (PRESS): A scalable kernel classifier for high-dimensional features with variable selection. Technical report, Google, 2018. URL https://ai.google/research/pubs/pub46767.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

J. Hainmueller. Entropy balancing for causal effects: A multivariate reweighting method to produce balanced samples in observational studies. *Political Analysis*, 20(1):25–46, 2012.

G. W. Imbens and D. B. Rubin. Rubin causal model. In *Microeconometrics*, pages 229–241. Springer, 2010.

G. W. Imbens and D. B. Rubin. *Causal Inference for Statistics, Social, and Biomedical Sciences: An Introduction*. Cambridge University Press, 2015. doi: 10.1017/CBO9781139025751.

F. D. Johansson, U. Shalit, and D. Sontag. Learning representations for counterfactual inference. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 3020–3029. JMLR.org, 2016. URL `http://dl.acm.org/citation.cfm?id=3045390.3045708`.

J. D. Y. Kang and J. L. Schafer. Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data. *Statist. Sci.*, 22(4):523–539, 11 2007. doi: 10.1214/07-STS227. URL `https://doi.org/10.1214/07-STS227`.

J. K. Lunceford and M. Davidian. Stratification and weighting via the propensity score in estimation of causal treatment effects: a comparative study. *Statistics in medicine*, 23(19):2937–2960, 2004.

N. Radcliffe. Using control groups to target on predicted lift: Building and assessing uplift model. *Direct Marketing Analytics Journal*, pages 14–21, 2007.

N. J. Radcliffe and P. D. Surry. Real-world uplift modelling with significance-based uplift trees. Technical report, Stochastic Solutions, 2011.

J. M. Robins and A. Rotnitzky. Semiparametric efficiency in multivariate regression models with missing data. *Journal of the American Statistical Association*, 90(429):122–129, 1995.

P. R. Rosenbaum. Covariance adjustment in randomized experiments and observational studies. *Statist. Sci.*, 17(3):286–327, 08 2002. doi: 10.1214/ss/1042727942. URL `https://doi.org/10.1214/ss/1042727942`.

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

P. R. Rosenbaum and D. B. Rubin. Constructing a control group using multivariate matched sampling methods that incorporate the propensity score. *The American Statistician*, 39(1):33–38, 1985.

D. B. Rubin. Causal inference using potential outcomes. *Journal of the American Statistical Association*, 100(469):322–331, 2005. doi: 10.1198/016214504000001880. URL `https://doi.org/10.1198/016214504000001880`.

D. B. Rubin. For objective causal inference, design trumps analysis. *Ann. Appl. Stat.*, 2(3):808–840, 09 2008. doi: 10.1214/08-AOAS187. URL `https://doi.org/10.1214/08-AOAS187`.

S. Ruder. An overview of multi-task learning in deep neural networks. *ArXiv*, abs/1706.05098, 2017.

E. A. Stuart. Matching methods for causal inference: A review and a look forward. *Statistical science: a review journal of the Institute of Mathematical Statistics*, 25(1):1, 2010.