A. Supplementary Materials *for*

Hierarchically Regularized Entropy Balancing

Yiqing Xu (Stanford) Eddie Yang (UCSD)

Table of Contents

A.1. Identifying the ATT

Suppose the researcher's goal is to estimate the ATT in a cross-sectional setting. Under the Neyman-Rubin potential outcomes framework, let Y_{1i} and Y_{0i} be the potential outcomes for units $i = 1, 2, ..., N$ under the treatment and control conditions, respectively. Let $D_i \in \{0, 1\}$ be the treatment assignment for unit i; $D_i = 1$ when i belongs to the treatment group $\mathcal{T}$ and $D_i = 0$ when i belongs to the control group $\mathcal{C}$. n_1 and n_0 are the numbers of treated units and control units, respectively. Let $G \in \mathbb{R}^J$ be J pretreatment covariates. To identify the ATT, we make the following assumptions:

Assumption 1 (Strong ignorability): The untreated potential outcome is independent of the treatment assignment conditional on the observed covariates G, i.e., $Y_{0i} \perp\!\!\!\perp D_i | G_i$.

Assumption 2 (Positivity): $0 < Pr(D_i = 1|G_i) < 1$ for all i.

Assumption 3 (Linearity in series expansion of covariates): There exists a series expansion of the covariates, $X = f(G)$ and $f(\cdot) : \mathbb{R}^J \to \mathbb{R}^T$, such that *either* the conditional expectation of Y_{0i} *or* the logit of the propensity score $\pi(X_i)$ is linear in X_i, i.e., $\mathbb{E}[Y_{0i}|G_i = g] = X_i'\theta$ or $logit(\pi(X_i)) = X_i'\theta$, for some $\theta \in \mathbb{R}^T$.

Define

$$\hat{\tau}_{hbal} = \frac{1}{n_1} \sum_{D_i=1} Y_i - \sum_{D_i=0} w_i^{hbal} Y_i.$$

Theorem 1 in Zhao and Percival (2016) shows that $\hat{\tau}_{hbal}$ is consistent for the ATT when exact balance is achieved. Essentially, this is equivalent to using entropy balance to achieve exact balance on the serially expanded covariates.

If exact balance is infeasible,

$$\hat{\tau}_{hbal+} = \frac{1}{n_1} \sum_{D_i=1} (Y_i - \hat{g}_0(G_i)) - \sum_{D_i=0} w^{hbal} (Y_i - \hat{g}_0(G_i)),$$

in which $\hat{g}_0(G_i) = X_i'\hat{\beta}$, is consistent for the ATT under Assumptions 1-3.

A.2. Algorithm

A.2.1. Newton's Method for Problem (2)

Given that the original *ebal* optimization problem in (1) is globally convex and twice differentiable (Hainmueller, 2012), and since the added regularization term in (2) is also convex and twice differentiable, the objective function of (2) is thus also convex and twice differentiable. Therefore, we can proceed to solve *hbal*'s optimization problem by Newton's method.

Specifically, let A be a vector of length T such that its t^{th} element corresponds to the value of the tuning parameters α associated with λ_t. Further let $M = \{m_1, \ldots, m_T\}'$ be the moments of the treated units and $W = \{w_1, \ldots, w_{n_0}\}'$ be a vector of weights for the control units, where n_0 denotes the number of control units. The gradient of (2) with respect to the Lagrangian multipliers Z^+ is given by $\frac{\delta L^d}{\delta Z^+} = M - X'W + 2A \circ Z^+$, where $\circ$ denotes the Hadamard product.[A1]

Furthermore, the Hessian is given by $\frac{\delta^2 L^d}{\delta Z^+ \delta Z^+} = X'[D(W) - WW']X + 2D(A)$, where $D(W)$ is a n_0-dimensional diagonal matrix with W in the diagonal and $D(A)$ is a J-dimensional diagonal matrix with A in the diagonal. Using Newton's method, the solution Z^+ is searched iteratively by $Z^{new} = Z^{old} - l \nabla_Z^2 L^{d-1} \nabla_Z L^d$, where l is a scalar that denotes the step length.

A.2.2. Derivation of Gradient and Hessian of the Optimization Problem of (2)

Given A, X, $Z^+ = \{\lambda_1 \ldots \lambda_T\}'$, $M = \{m_1, \ldots, m_T\}'$, and $W = \{w_1, \ldots, w_{n_0}\}'$, further let $Q = \{q_1 \ldots q_{n_0}\}'$, we can rewrite the dual problem

$$\min_{Z^+} L^d = \log\left(\sum_{i \in \mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T} \lambda_t X_{it}\right)\right) + \sum_{t=1}^{T} \lambda_t m_t + \sum_{k=1}^{K} \alpha_k \sum_{l=1}^{L} \|\lambda_l\|_2$$

as

$$\min_{Z^+} L^d = \log(Q' \exp(-XZ^+)) + M'Z^+ + A \circ Z^{+\prime} Z^+ \tag{A1}$$

We can also write the equation for solution weights w_i^*:

$$w_i^* = \frac{q_i \exp\left(-\sum_{t=1}^{T} \lambda_t X_{it}\right)}{\sum_{i \in \mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T} \lambda_t X_{it}\right)}$$

as

$$w_i^* = \frac{Q \exp(-XZ^+)}{Q' \exp(-XZ^+)}$$

Differentiating (A1) with respect to Z^+ gives the gradient

[A1] For details on the derivation of the gradient and the Hessian, see Section A.2.2 in SM.

$$
\begin{aligned}
\frac{\delta L^d}{\delta Z^+} &= \frac{\delta}{\delta Z^+}\log(Q'\exp(-XZ^+)) + M'Z^+ + A\circ Z^{+\prime}Z^+ \\
&= \frac{1}{Q'\exp(-XZ^+)}\frac{\delta}{\delta Z^+}\left[Q'\exp(-XZ^+)\right] + M + 2A\circ Z^+ \\
&= \frac{1}{Q'\exp(-XZ^+)}\frac{\delta}{\delta Z^+}\left(\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)\right) + M + 2A\circ Z^+ \\
&= \frac{1}{Q'\exp(-XZ^+)}\begin{bmatrix} \sum(X_{i1}q_i\exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right) \\ \sum(X_{i2}q_i\exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right) \\ \vdots \\ \sum(X_{iT}q_i\exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right) \end{bmatrix} + M + 2A\circ Z^+ \\
&= \frac{1}{Q'\exp(-XZ^+)}\left(-X'(Q\exp(-XZ^+))\right) + M + 2A\circ Z^+ \\
&= M - X'W + 2A\circ Z^+.
\end{aligned}
$$

Given the gradient, we can derive the Hessian by the following:

$$
\begin{aligned}
\frac{\delta^2 L^d}{(\delta Z^+)^2} &= \frac{\delta}{\delta Z^{+\prime}}\left(M - X'W + 2A\circ Z^+\right) \\
&= -\frac{\delta}{\delta Z^{+\prime}}\left(X'W\right) + 2D(A) \\
&= -\frac{\delta}{\delta Z^{+\prime}}\left(X'\frac{Q\exp(-XZ^+)}{Q'\exp(-XZ^+)}\right) + 2D(A) \\
&= -\frac{\delta}{\delta Z^{+\prime}}\begin{bmatrix} \frac{c_{11}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)+\ldots c_{n_0 1}q_{n_0}exp\left(-\sum_{t=1}^{T}\lambda_t X_{n_0 t}\right)}{\sum_{i\in\mathcal{C}} q_i\exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)} \\ \vdots \\ \frac{c_{1T}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)+\ldots c_{n_0 T}q_{n_0}exp\left(-\sum_{t=1}^{T}\lambda_t X_{n_0 t}\right)}{\sum_{i\in\mathcal{C}} q_i\exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)} \end{bmatrix} + 2D(A) \\
&= X'\begin{bmatrix} w_1 & & \\ & \ddots & \\ & & w_{n_0} \end{bmatrix}X - X'\begin{bmatrix} w_1 \\ \vdots \\ w_{n_0} \end{bmatrix}\left[w_1\ldots w_{n_0}\right]X + 2D(A) \\
&= X'[D(W) - WW']X + 2D(A).
\end{aligned}
$$

Note that $w_i = \frac{q_i exp\left(-\sum_{t=1}^{T}\lambda_t X_{n_0 t}\right)}{\sum_{i\in\mathcal{C}} q_i\exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)}$ and where

$$\frac{\delta}{\delta Z^{+\prime}}\begin{bmatrix} \frac{c_{11}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)+\ldots c_{n_0 1}q_{n_0} exp\left(-\sum_{t=1}^{T}\lambda_t X_{n_0 t}\right)}{\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)} \\ \vdots \\ \frac{c_{1T}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)+\ldots c_{n_0 T}q_{n_0} exp\left(-\sum_{t=1}^{T}\lambda_t X_{n_0 t}\right)}{\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\sum c_{i1}^2 q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)}{\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)} & \cdots & \frac{\sum c_{i1}c_{iT} q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)}{\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)} \\ \vdots & \ddots & \vdots \\ \frac{\sum c_{iT}c_{i1} q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)}{\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)} & \cdots & \frac{\sum c_{iT}^2 q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)}{\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)} \end{bmatrix} -$$

$$\begin{bmatrix} \frac{\left[\sum c_{i1}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)\right]^2}{\left[\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)\right]^2} & \cdots & \frac{\left[\sum c_{i1}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)\right]\left[\sum c_{iT}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)\right]}{\left[\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)\right]^2} \\ \vdots & \ddots & \vdots \\ \frac{\left[\sum c_{iT}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)\right]\left[\sum c_{i1}q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)\right]}{\left[\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)\right]^2} & \cdots & \frac{\sum c_{iT}^2 q_1 exp\left(-\sum_{t=1}^{T}\lambda_t X_{1t}\right)}{\left[\sum_{i\in\mathcal{C}} q_i \exp\left(-\sum_{t=1}^{T}\lambda_t X_{it}\right)\right]^2} \end{bmatrix}.$$

A.2.3. Alternative Distance Metrics

Here we consider two alternative distance metrics to Kullback (1959) entropy divergence. In particular, we consider empirical likelihood (EL), defined as $\log(w)$, and chi-square distance (CD), defined as $\frac{(w-q)^2}{q}$ for base weights q. Both metrics have received much scholarly attention as a measure of distance between the estimated and the ideal weights (See e.g. Owen 1988; Deville and Särndal 1992; Qin and Lawless 1994). Although these distance metrics are asymptotically equivalent (Little and Wu, 1991; Imbens, Johnson and Spady, 1995), we use simulations to consider their finite sample performance in our case.

For EL, the objective function is changed to

$$\min_{Z^+} L^d = \sum_{D_i=0} \log(w_i) + \sum_{t=1}^{T} \lambda_t \left(\sum_{D_i=0} w_i X_i - m_t \right) + \sum_{k=1}^{K} \alpha_k r_k \tag{A2}$$

with the solution weights given by $w = \frac{1}{XZ^+}$, for covariate matrix X and vector of Lagrangian multipliers Z^+

For CD, the objective function is changed to

$$\min_{Z^+} L^d = \sum_{D_i=0} \frac{(w_i - q_i)^2}{q_i} + \sum_{t=1}^{T} \lambda_t \left(\sum_{D_i=0} w_i X_i - m_t \right) + \sum_{k=1}^{K} \alpha_k r_k \tag{A3}$$

with the solution weights given by $w = Q(1 + XZ^{+})$.

Same as *hbal*, we derive the first and second derivatives of equations A2 and A3 and use Newton's method to obtain the solution weights. Through simulations (Table A1), we find that both EL and CD are less robust than entropy distance in finite samples. In particular, CD may yield negative weights (500 out of 500 simulations) and EL are prone to convergence failure (177 out of 500 simulations), when the number of constraints is large (as is our case). Both phenomena have been pointed out by prior work (for robustness of EL, see Imbens, Johnson and Spady (1995); for negative weights in CD, see Deville and Särndal (1992)).

TABLE A1. COMPARISON OF *hbal* WITH ALTERNATIVE DISTANCE METRICS

	CD	CD+	EL	EL+	hbal	hbal+
Outcome Design 1						
Bias	-33.0	-	-1.6	-0.9	-1.3	-1.4
MSE	3178.1	-	9.3	1.5	1.2	1.5
Outcome Design 2						
Bias	-205.7	-	0.4	-0.9	0.1	-1.4
MSE	102888.1	-	29.5	1.5	2.6	1.5
Outcome Design 3						
Bias	62.5	-	-5.0	-2.8	-2.7	-1.2
MSE	38645.9	-	37.2	2.4	2.0	2.3

Note: MSE stands for mean squared error, i.e., $\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$. Results are averaged over 500 simulations. Results for bias and MSE are $\times 100$ for better presentation. CD+, EL+ stand for CD and EL coupled with a linear outcome model respectively. Because CD produced negative weights for all of the simulations and doubly robust estimators do not support negative weights, there are no entries for CD+.

A.2.4. L1 Regularization

We also consider using the L1 instead of L2 norm of the Lagrangian multipliers as the penalty term in the objective function. Because L1 norm is not differentiable at zero, we resort to a derivative-free optimization scheme (Powell, 1994) in both cross-validation and obtaining the solutions weights. In simulations, we find that using L1 penalty causes the algorithm to take up about 28x computation time than L2 penalty. Additionally, results obtained from using L1 penalty have larger bias and variance than those from using L2 penalty, likely because solutions from the derivative-free optimization scheme are less accurate.

TABLE A2. COMPARISON OF *hbal* WITH L1 REGULARIZATION

	hbal-L1	hbal-L1+	hbal	hbal+
Outcome Design 1				
Bias	-11.0	-0.7	-0.8	-0.6
MSE	5.2	1.4	1.1	1.4
Outcome Design 2				
Bias	-10.9	-0.7	1.1	-0.6
MSE	4.3	1.4	2.3	1.4
Outcome Design 3				
Bias	-9.6	-2.8	-2.1	-0.6
MSE	2.9	2.2	2.1	2.3

Note: MSE stands for mean squared error, i.e., $\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$. Results are averaged over 500 simulations. Results for bias and MSE are $\times 100$ for better presentation. *hbal-L1* and *hbal-L1+* stand for *hbal* with l1 regularization and *hbal+* with l1 regularization respectively.

A.3. Implementation Details

A.3.1. Series Expansion: An Example

Here we provide a toy example to illustrate the series expansion and penalty-searching steps in Section 2. Consider two continuous covariates G_1 and G_2 (for binary variables, we only consider linear terms and interactions). To balance higher moments and interaction of these covariates, we simply include their higher-order terms and interaction as additional covariates. For up to the third order polynomials, we have the new covariate set $\{G_1, G_2, G_1^2, G_2^2, G_1G_2, G_1^3, G_2^3, G_1^2G_2, G_1G_2^2\}$. Now we break the expanded covariate set into groups. We have linear terms $k_1 = \{G_1, G_2\}$; two-way interaction term $k_2 = \{G_1G_2\}$; square terms $k_3 = \{G_1^2, G_2^2\}$; interaction between linear and square terms $k_4 = \{G_1^2G_2, G_1G_2^2\}$; and cubic terms $k_5 = \{G_1^3, G_2^3\}$.

We keep balance constraints on the linear terms k_1 unpenalized, setting $\alpha_1 = 0$. To search for optimal penalties on k_2, k_3, k_4, k_5, we use grid search over a set of α values and V-fold cross-validation to search for the α values that minimize out-of-sample covariate imbalance. Specifically, given four fixed folds and for each set of $\{\alpha_2, \alpha_3, \alpha_4, \alpha_5\}$ values, we calculate the mean absolute error (MAE) between the first covariate moments of the held-out treated units and those of the reweighted held-out control units. The set of α values with the lowest MAE are then selected to calculate the final solution weights. The optimization procedure

is implemented in R with the nloptr package, using the constrained optimization by linear approximations algorithm (Powell, 1994).

A.3.2. Parameter Sharing

To speed up computation of the algorithm, we use the solution Lagrangian multipliers from the previous round of cross-validation as the starting values for the next round of cross-validation. We also use the solution Lagrangian multipliers for the previous α value as the starting point in the first round of cross-validation for the current α value. In our experiment, compared with initializing Lagrangian multipliers values as zero or random, the parameter-sharing scheme significantly speeds up convergence and thus substantially reduces computation time.

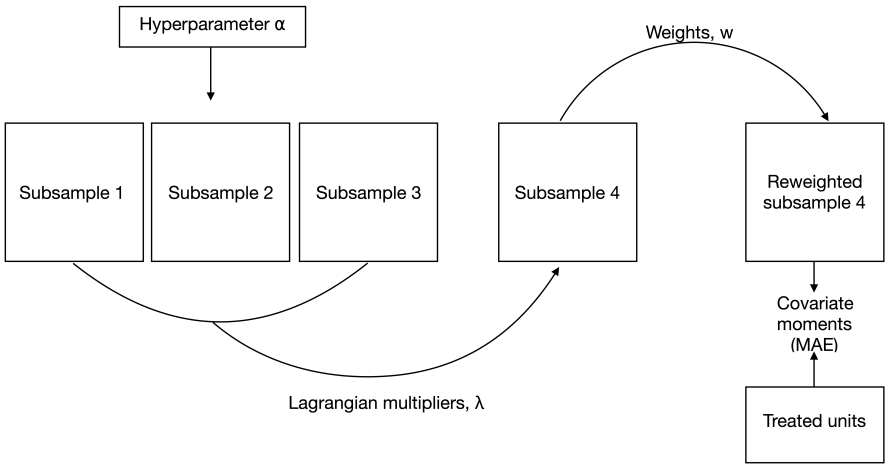
A.3.3. Selecting Tuning Parameters

To search for tuning parameters α, we combine grid search with a V-fold cross-validation procedure that minimizes mean absolute error (MAE) of covariate balance between the held-out sample of the control units and the treated units. In other words, we select tuning parameters that minimize the mean absolute differences across covariates (including their higher-order terms and interactions) between the weighted control units from the held-out sample and the treated units. The intuition for this cross-validation scheme is that, for a given set of tuning parameters, if the resulting coefficients (Lagrangian multipliers) closely approximate the true coefficients, then we should be able to use these coefficients to achieve good covariate balance on the held-out sample (because it comes from the same data generating process).

Specifically, we first subset the control group into V subsamples. We then use $(V-1)$ samples to find solution Lagrangian multipliers based on the objective function in Problem (2). With these Lagrangian multipliers, we construct a set of weights on the held-out V^{th} sample, and assess the out-of-sample covariate balance between the treatment group and the reweighted V^{th} sample. This process is then repeated for different values of the tuning parameters. The tuning parameters with the smallest MAE on the held-out sample is chosen. A graphical representation of the procedure for four-fold cross-validation is shown in Figure A1. Obtaining the penalties following this procedure thus avoids overfitting, i.e., to achieve exact balance on features that differ between the treatment and control groups only because of random noises in the data.

In order to use MAE as the metric to select tuning parameters α using cross-validation, we need to make sure that all covariates are on the same scale so that their imbalance (balance) has equal weight on the choice of α. To achieve this, we standardize all covariates

FIGURE A1. TUNING PARAMETER SELECTION SCHEME



by subtracting their respective mean and diving by their respective standard deviation, before running *hbal*. We then balance on the standardized moments while using cross-validation to select the optimal α.

A.3.4. Prescreening Moment Conditions

When the number of covariates is large, a series expansion (e.g., to the third order) will produce an enormous number of additional variables, which makes approximate balancing a challenging task. For example, in an application with 10 continuous covariates, a full series expansion to the third order will create close to 300 covariates. Moreover, many of these variables may be irrelevant for treatment assignment or predicting the outcome. To further reduce the dimensionality of the optimization problem and improve efficiency, we provide an option to prescreen the expanded covariates using the double-selection method (Belloni, Chernozhukov and Hansen, 2014). Specifically, we first fit two lasso regressions on the outcome and the treatment variables respectively using the expanded covariates and then select the union of the selected covariates from the two lasso regressions. In addition, when a higher-order term is selected, we make sure to include its lower-order compositions even if they are not selected.

A.3.5. A Summary of the Procedure

We provide a sketch of the full procedure below.

Procedure: Hierarchically Regularized Entropy Balancing

1. Perform a series expansion of the covariates (e.g., up to the third degree);
2. Select the tuning parameters using a cross-validation method;
3. Reweight control units to achieve approximate covariate balance using hierarchical regularization;
4. Obtain an ATT estimate.

A.4. Additional Simulation Results

A.4.1. Full Monte Carlo Results - Point Estimation

TABLE A3. CONTROL TO TREATMENT 1:1

	N = 600			N = 900			N = 1200			N = 1500		
	Bias	MSE	Time	Bias	MSE	Time	Bias	MSE	Time	Bias	MSE	Time
Outcome Design 1												
Raw	-14.04	5.75	0.00	-12.24	3.90	0.00	-14.95	4.01	0.00	-13.94	3.34	0.00
CBPS	5.68	1.17	0.26	6.65	1.08	0.36	5.38	0.73	0.44	5.59	0.68	0.56
PSW	0.02	1.14	0.00	0.18	0.88	0.00	-0.72	0.58	0.00	-0.64	0.50	0.00
CEM	-2.81	11.03	0.01	0.90	4.18	0.02	-2.15	3.65	0.02	-1.69	1.98	0.03
ebal	0.18	0.80	0.00	0.59	0.58	0.00	-0.46	0.42	0.00	-0.39	0.35	0.00
ebal*	-11.28	5.16	0.02	-10.52	4.41	0.03	-10.38	4.53	0.04	-9.99	3.70	0.05
kbal	-1.89	4.80	2.72	-2.99	4.27	5.24	-1.74	3.21	9.12	-2.90	2.30	14.36
hbal	-2.28	1.88	0.45	-2.30	1.39	0.55	-3.16	1.19	0.64	-2.19	0.95	0.72
hbal+	-0.06	1.33	0.48	0.38	0.91	0.59	-0.59	0.72	0.68	-0.54	0.57	0.77
Outcome Design 2												
Raw	-16.56	5.34	0.00	-13.73	3.97	0.00	-16.13	3.87	0.00	-15.45	3.63	0.00
CBPS	6.30	2.61	0.26	7.91	2.40	0.36	6.11	1.42	0.44	6.19	1.48	0.56
PSW	4.95	2.43	0.00	6.53	2.15	0.00	4.80	1.22	0.00	4.74	1.29	0.00
CEM	-2.77	11.30	0.01	-0.32	4.12	0.02	-2.12	3.71	0.02	-2.00	1.98	0.03
ebal	8.90	3.28	0.00	10.10	2.99	0.00	8.24	1.85	0.00	8.09	1.89	0.00
ebal*	-2.44	3.76	0.02	-1.25	3.50	0.03	-2.60	3.51	0.04	-3.42	2.77	0.05
kbal	-8.95	5.87	2.72	-10.29	5.31	5.24	-9.80	4.60	9.12	-11.40	3.84	14.36
hbal	0.32	2.97	0.45	-0.31	2.41	0.55	-2.13	1.74	0.64	-1.45	1.39	0.72
hbal+	-0.06	1.33	0.48	0.38	0.91	0.59	-0.59	0.72	0.68	-0.54	0.57	0.77
Outcome Design 3												
Raw	-8.24	2.24	0.00	-7.94	1.65	0.00	-8.61	1.51	0.00	-8.90	1.49	0.00
CBPS	-6.98	2.19	0.26	-6.29	1.52	0.36	-7.24	1.37	0.44	-7.55	1.35	0.56
PSW	-6.74	2.29	0.00	-6.11	1.57	0.00	-7.08	1.40	0.00	-7.42	1.37	0.00
CEM	-3.00	11.82	0.01	0.64	4.75	0.02	-1.68	3.51	0.02	-1.09	2.18	0.03
ebal	-6.36	2.32	0.00	-5.72	1.59	0.00	-6.87	1.41	0.00	-7.01	1.36	0.00
ebal*	-5.55	4.64	0.02	-7.35	4.93	0.03	-7.13	4.96	0.04	-6.87	3.97	0.05
kbal	1.63	6.46	2.72	0.82	5.20	5.24	1.07	4.46	9.12	-0.22	3.06	14.36
hbal	-3.89	2.30	0.45	-2.59	1.76	0.55	-2.43	1.32	0.64	-2.13	1.25	0.72
hbal+	0.28	2.03	0.48	0.97	1.47	0.59	0.16	1.09	0.68	-0.18	0.90	0.77

Note: MSE stands for mean squared error, i.e., $\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$. Results are averaged over 500 simulations. Results for bias and MSE are $\times 100$ for better presentation. Time is measured in seconds. Raw, CBPS, PSW, CEM stand for difference in means, covariate balancing propensity score, inverse propensity score weighting, and coarsened exact matching, respectively. *ebal ebal**, *kbal*, *hbal*, *hbal+* stand for entropy balancing on the first moments of the covariates, entropy balancing on the serially expanded covariate set, kernel balancing, hierarchically regularized entropy balancing, and hierarchically regularized entropy balancing coupled with an outcome model, respectively.

TABLE A4. CONTROL TO TREATMENT 3:1

	N = 600			N = 900			N = 1200			N = 1500		
	Bias	MSE	Time	Bias	MSE	Time	Bias	MSE	Time	Bias	MSE	Time
Outcome Design 1												
Raw	-14.98	7.48	0.00	-14.68	5.76	0.00	-14.02	4.51	0.00	-13.25	3.92	0.00
CBPS	1.87	1.14	0.20	2.00	0.75	0.27	2.18	0.62	0.35	2.34	0.49	0.43
PSW	0.91	1.39	0.00	1.31	0.89	0.00	1.34	0.69	0.00	1.61	0.57	0.00
CEM	0.16	16.03	0.01	-1.85	6.85	0.02	-0.81	5.51	0.02	-0.52	3.04	0.03
ebal	-0.43	1.08	0.00	0.05	0.65	0.00	0.21	0.56	0.00	0.27	0.43	0.00
ebal*	-12.01	6.12	0.03	-10.14	4.47	0.04	-8.37	4.13	0.05	-5.93	3.55	0.07
kbal	-2.39	4.87	4.33	-3.10	3.60	8.64	-1.78	2.88	14.08	-1.54	2.03	21.08
hbal	-1.34	1.47	0.61	-0.79	1.01	0.71	-0.37	0.72	0.83	0.02	0.59	0.98
hbal+	-0.95	1.71	0.64	-0.36	1.08	0.75	0.32	0.97	0.88	0.14	0.64	1.03
Outcome Design 2												
Raw	-15.94	6.05	0.00	-15.36	4.77	0.00	-14.11	3.78	0.00	-14.49	3.56	0.00
CBPS	6.75	3.46	0.20	8.53	2.97	0.27	8.91	2.43	0.35	8.57	2.07	0.43
PSW	5.49	3.15	0.00	7.03	2.64	0.00	7.43	2.08	0.00	7.07	1.76	0.00
CEM	-0.27	15.98	0.01	-1.79	6.73	0.02	-0.91	5.33	0.02	-0.63	3.10	0.03
ebal	7.43	3.55	0.00	9.29	3.09	0.00	9.61	2.56	0.00	9.22	2.19	0.00
ebal*	-2.66	4.82	0.03	-1.59	3.80	0.04	-1.30	3.38	0.05	-0.93	3.21	0.07
kbal	-12.02	6.72	4.33	-13.68	5.35	8.64	-12.69	4.60	14.08	-13.02	3.78	21.08
hbal	0.75	2.86	0.61	1.58	2.11	0.71	1.18	1.48	0.83	0.99	1.15	0.98
hbal+	-0.95	1.71	0.64	-0.36	1.08	0.75	0.32	0.97	0.88	0.14	0.64	1.03
Outcome Design 3												
Raw	-8.72	3.15	0.00	-8.50	2.03	0.00	-8.67	1.86	0.00	-8.43	1.58	0.00
CBPS	-6.70	3.03	0.20	-6.80	1.89	0.27	-6.85	1.73	0.35	-6.59	1.37	0.43
PSW	-6.75	3.05	0.00	-6.84	1.89	0.00	-6.91	1.73	0.00	-6.71	1.39	0.00
CEM	-1.41	17.35	0.01	-0.33	6.93	0.02	-0.37	6.08	0.02	-0.44	3.36	0.03
ebal	-6.50	3.07	0.00	-6.64	1.90	0.00	-6.69	1.75	0.00	-6.41	1.37	0.00
ebal*	-6.02	5.61	0.03	-5.38	4.80	0.04	-4.93	4.77	0.05	-4.63	4.49	0.07
kbal	4.14	6.87	4.33	2.58	4.43	8.64	2.03	3.71	14.08	1.44	2.80	21.08
hbal	-3.39	2.64	0.61	-2.14	1.45	0.71	-1.36	1.20	0.83	-0.91	0.97	0.98
hbal+	-1.04	2.63	0.64	-0.28	1.83	0.75	0.68	1.40	0.88	0.26	0.96	1.03

Note: MSE stands for mean squared error, i.e., $\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$. Results are averaged over 500 simulations. Results for bias and MSE are $\times 100$ for better presentation. Time is measured in seconds. Raw, CBPS, PSW, CEM stand for difference in means, covariate balancing propensity score, inverse propensity score weighting, and coarsened exact matching, respectively. *ebal ebal**, *kbal*, *hbal*, *hbal+* stand for entropy balancing on the first moments of the covariates, entropy balancing on the serially expanded covariate set, kernel balancing, hierarchically regularized entropy balancing, and hierarchically regularized entropy balancing coupled with an outcome model, respectively.

TABLE A5. CONTROL TO TREATMENT 5:1

	N = 600			N = 900			N = 1200			N = 1500		
	Bias	MSE	Time	Bias	MSE	Time	Bias	MSE	Time	Bias	MSE	Time
Outcome Design 1												
Raw	-14.40	8.52	0.00	-13.68	6.54	0.00	-13.23	5.65	0.00	-13.62	4.42	0.00
CBPS	0.01	1.26	0.20	0.42	0.85	0.26	0.52	0.66	0.34	-0.25	0.53	0.42
PSW	1.37	1.51	0.00	1.56	1.00	0.00	1.53	0.78	0.00	1.08	0.58	0.00
CEM	-0.26	26.64	0.01	-2.69	9.96	0.02	-1.80	7.07	0.02	-0.78	4.46	0.03
ebal	-0.52	1.23	0.00	0.10	0.86	0.00	0.17	0.65	0.00	-0.53	0.53	0.00
ebal*	-12.29	7.29	0.03	-9.43	5.13	0.04	-8.96	4.64	0.06	-7.64	3.91	0.08
kbal	-4.26	5.25	4.98	-3.21	3.80	10.02	-1.67	2.66	16.05	-3.23	2.32	24.34
hbal	-0.77	1.47	0.77	-0.63	1.23	0.84	0.00	0.75	1.01	-0.78	0.64	1.17
hbal+	-1.27	2.04	0.80	0.22	1.51	0.88	0.03	1.07	1.05	-0.66	0.90	1.22
Outcome Design 2												
Raw	-16.75	7.56	0.00	-15.14	5.29	0.00	-14.12	4.39	0.00	-16.31	4.58	0.00
CBPS	6.15	4.63	0.20	8.23	3.35	0.26	9.07	2.78	0.34	6.65	2.14	0.42
PSW	5.00	4.38	0.00	6.91	3.05	0.00	7.78	2.50	0.00	5.38	1.90	0.00
CEM	-2.55	26.83	0.01	-3.19	9.83	0.02	-3.33	7.13	0.02	-0.80	4.47	0.03
ebal	6.51	4.70	0.00	8.57	3.41	0.00	9.42	2.86	0.00	6.98	2.19	0.00
ebal*	-3.73	5.71	0.03	-2.40	3.91	0.04	-3.28	3.96	0.06	-3.25	3.39	0.08
kbal	-14.86	7.55	4.98	-14.43	5.79	10.02	-13.60	4.81	16.05	-15.50	4.88	24.34
hbal	1.16	3.68	0.77	1.21	2.33	0.84	1.70	1.76	1.01	-1.01	1.38	1.17
hbal+	-1.27	2.04	0.80	0.22	1.51	0.88	0.03	1.07	1.05	-0.66	0.90	1.22
Outcome Design 3												
Raw	-8.10	3.53	0.00	-8.80	2.52	0.00	-8.65	2.24	0.00	-8.65	1.82	0.00
CBPS	-5.66	3.54	0.20	-6.67	2.43	0.26	-6.67	2.01	0.34	-6.66	1.63	0.42
PSW	-5.66	3.49	0.00	-6.81	2.43	0.00	-6.83	2.02	0.00	-6.78	1.64	0.00
CEM	-1.98	26.33	0.01	-2.85	11.34	0.02	-1.84	7.55	0.02	0.11	4.67	0.03
ebal	-5.54	3.56	0.00	-6.60	2.44	0.00	-6.59	2.01	0.00	-6.57	1.63	0.00
ebal*	-5.58	6.56	0.03	-4.83	5.58	0.04	-6.06	5.25	0.06	-5.93	4.81	0.08
kbal	3.50	6.80	4.98	3.17	4.52	10.02	2.72	3.50	16.05	0.39	2.84	24.34
hbal	-2.05	3.15	0.77	-2.04	1.93	0.84	-1.25	1.52	1.01	-1.06	1.08	1.17
hbal+	-0.72	3.49	0.80	0.28	2.30	0.88	0.16	1.76	1.05	-0.45	1.24	1.22

Note: MSE stands for mean squared error, i.e., $\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$. Results are averaged over 500 simulations. Results for bias and MSE are $\times 100$ for better presentation. Time is measured in seconds. Raw, CBPS, PSW, CEM stand for difference in means, covariate balancing propensity score, inverse propensity score weighting, and coarsened exact matching, respectively. *ebal ebal**, *kbal*, *hbal*, *hbal+* stand for entropy balancing on the first moments of the covariates, entropy balancing on the serially expanded covariate set, kernel balancing, hierarchically regularized entropy balancing, and hierarchically regularized entropy balancing coupled with an outcome model, respectively.

A.4.2. Additional Result: Hierarchical vs. Nonhierarchical Regularization

We show the advantage of hierarchical regularization by comparing *hbal* and *ebal* with non-hierarchical regularization. For nonhierarchical regularization, we simply treat all higher-order terms as one group and use the same search scheme to select the tuning parameter α. We use the same data-generating process as in the main simulation with sample size of 900 and control to treatment ratio of 5 : 1. Results below are averaged from 500 random samples.

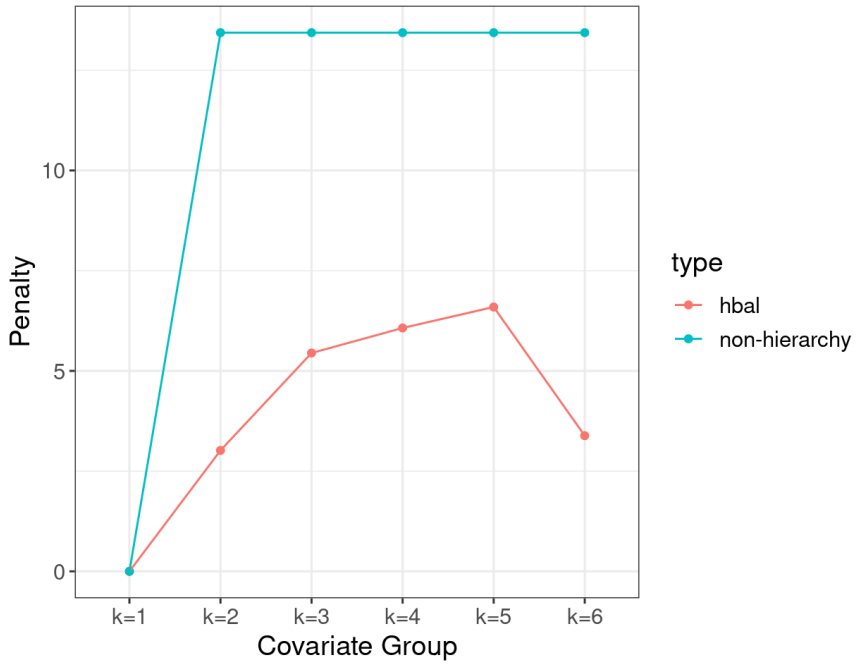
As Table A6 shows, for nonlinear outcome designs 2 and 3, *hbal* is able to achieve lower bias than *ebal* with nonhierarchical regularization. By checking the penalties assigned to each group of covariates as shown in Figure A2, we can see that nonhierarchical regularization assigns a uniform penalty of 13.43 for all higher-order terms while *hbal* assigns the smallest penalty for the two-way interactions (k = 2) that are important for treatment assignment and larger penalties for other higher-order terms.

TABLE A6. COMPARISON OF *hbal* WITH NON-HIERARCHICAL REGULARIZATION

	nonh-ierarchical	hbal	hbal+
Outcome Design 1			
Bias	-0.6	-0.8	-0.6
MSE	0.9	1.1	1.4
Outcome Design 2			
Bias	4.0	1.1	-0.6
MSE	2.7	2.3	1.4
Outcome Design 3			
Bias	-4.2	-2.1	-0.6
MSE	2.3	2.1	2.3

Note: MSE stands for mean squared error, i.e., $\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$. Results are averaged over 500 simulations. Results for bias and MSE are $\times 100$ for better presentation.

FIGURE A2. PENALTIES FOR DIFFERENT GROUPS OF COVARIATES



Note: linear terms ($k = 1$); two-way interactions ($k = 2$), squared terms ($k = 3$), three-way interactions ($k = 4$), interactions between square and level terms ($k = 5$), and cubic terms ($k = 6$). Penalties averaged across 500 simulations.

A.4.3. Additional Result: Additional Comparisons

In addition to the methods used in the simulation section of the main text, here we include *hbal*'s comparison with additional methods. In particular, we consider augmented inverse propensity score weighting with an outcome model using level terms (IPWDR1) and serially expanded covariates (IPWDR2), as well as ridge regression (with no shrinkage on the treatment variable) with optimal cross-validated penalty (ridge1) and largest penalty that results in one standard deviation of the minimum cross-validation error (ridge2).

Across outcome designs, we can see in Table A7 that *hbal* and *hbal+* generally are on par with or achieve the best results in comparison with the other methods.

TABLE A7. COMPARISON OF *hbal* WITH ADDTIONAL METHODS

	IPWDR1	IPWDR2	ridge1	ridge2	hbal	hbal+
Outcome Design 1						
Bias	0.1	-2.7	-2.6	-4.7	-0.5	-0.3
MSE	1.2	14.1	1.2	1.4	1.1	1.3
Outcome Design 2						
Bias	13.3	-2.8	-1.5	-2.8	1.5	-0.3
MSE	6.1	14.1	1.2	1.3	2.5	1.3
Outcome Design 3						
Bias	-5.7	2.2	-2.3	-4.2	-1.6	0.2
MSE	3.0	19.3	1.6	1.7	1.9	2.0

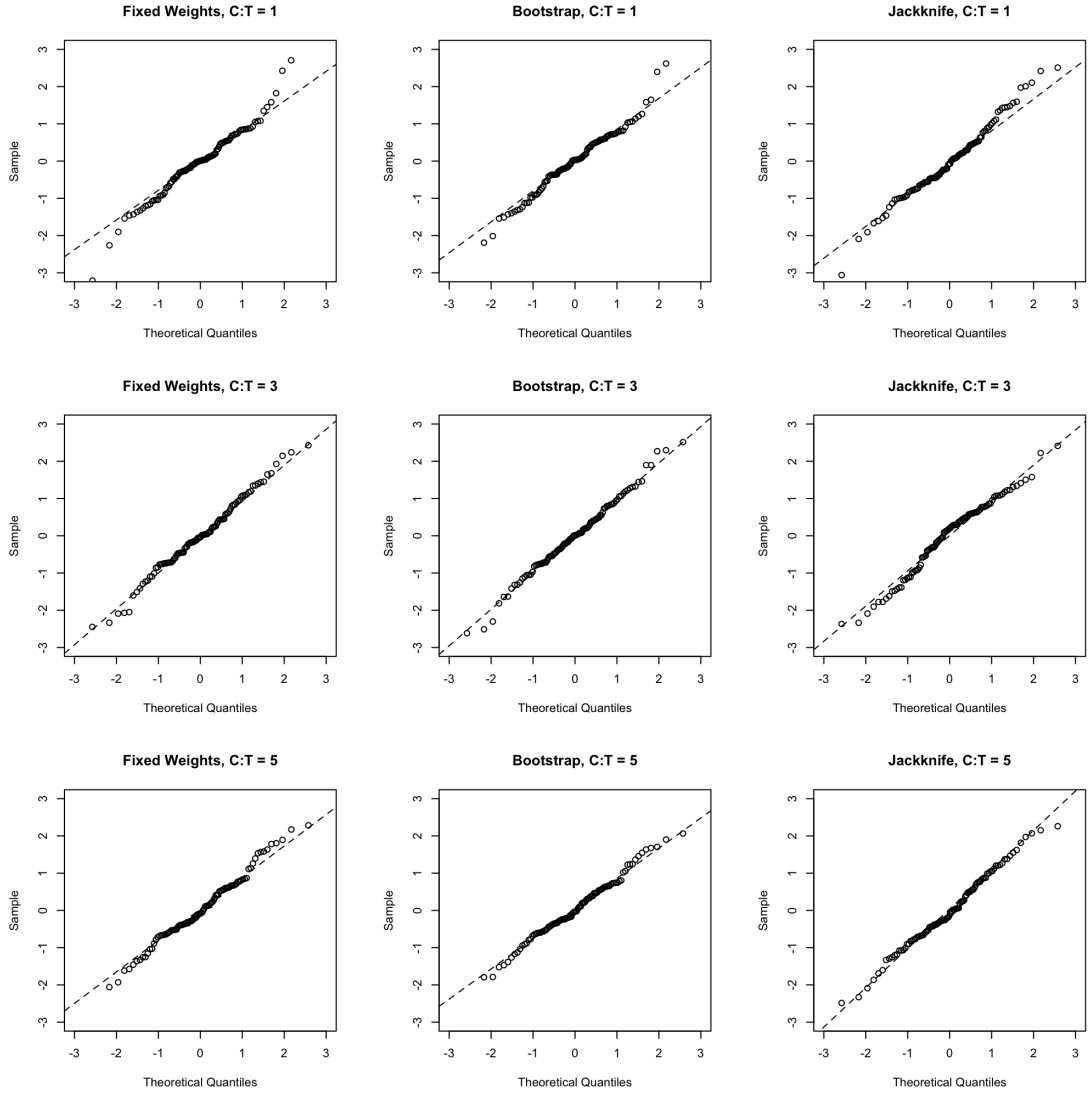
Note: MSE stands for mean squared error, i.e., $\frac{1}{n}\sum_{i=1}^{n}(Y_i-\hat{Y}_i)^2$. Results are averaged over 500 simulations. Results for bias and MSE are $\times 100$ for better presentation. IPWDR1 and IPWDR2 stand for augmented inverse propensity score weighting with an outcome model using level terms and serially expanded covariates respectively. ridge1 and ridge2 stand for ridge regression (with no shrinkage on the treatment variable) with optimal cross-validated penalty and largest penalty that results in one standard deviation of the minimum cross-validation error respectively.

A.4.4. Simulation Results: Standard Errors

We also study the finite sample properties of the variance estimator via simulations. We simulate samples of size 900 with outcome design 1 and the treatment assignment mechanism described earlier in the paper. Following Arkhangelsky et al. (2018) and Liu, Wang and Xu (2020), we plot the quantiles of the distribution for the standardized errors of the ATT estimates, i.e., $(\widehat{ATT}-ATT)/(\widehat{VAR}(\widehat{ATT}))^{1/2}$, based on 100 simulated samples against the quantiles of the standard normal distribution—a QQ plot—using three inferential methods: (1) fixed-weight standard errors; (2) bootstrapped standard errors; and (3) jackknife standard errors. If the ATT estimator is consistent and asymptotically normal and the chosen variance estimator precisely estimates its variance, the QQ plot should be very close to a 45-degree line.

Figure A3 presents the result. The first row shows the QQ plots of the three inferential methods when the ratio of the control to treated units is 1, the second row shows the result when the control to treatment ratio is 3 and the third row for when the ratio is 5. Across different control to treatment ratios, the fixed weights variance estimator is well calibrated, as the points are almost exactly on the 45-degree lines. This suggests that using the fixed-weight standard errors as returned by *hbal* is valid for variance estimation.

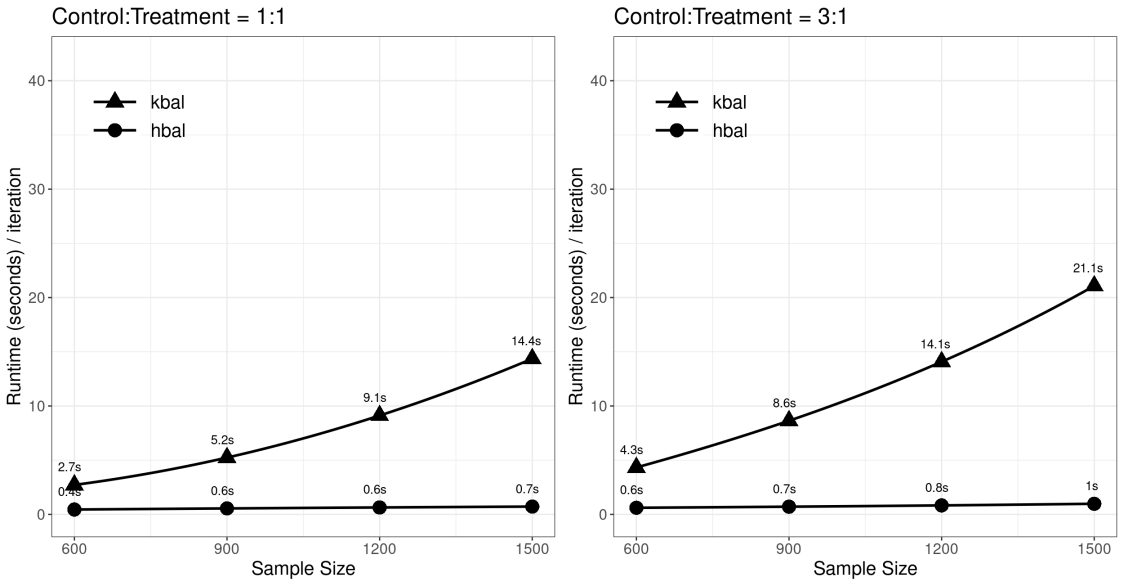
FIGURE A3. STANDARD GAUSSIAN QQ PLOTS OF THE STANDARDIZED ERRORS



A.4.5. Additional Runtime Comparison

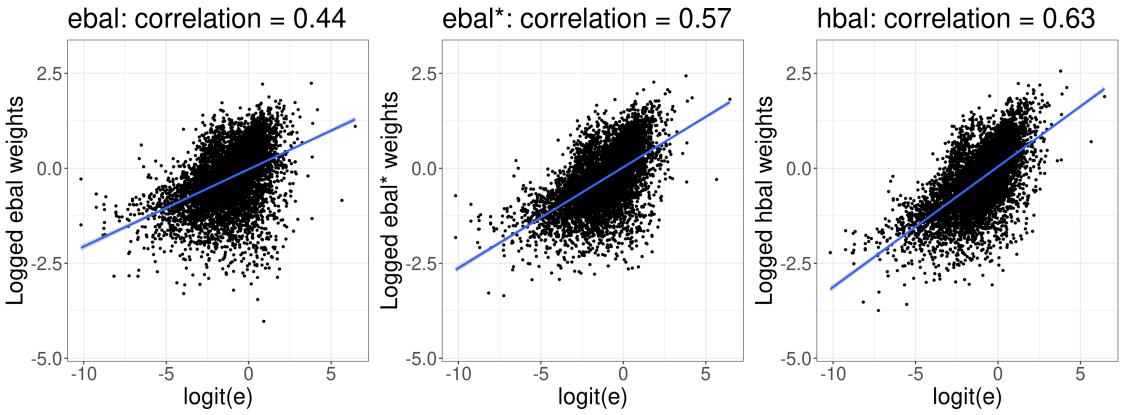
Figure A4 reports runtime comparisons when the ratio of the number of control units to the number of treated units is 1:1 and 3:1. Results are similar to the runtime comparison result reported in the main text. Across sample sizes and control to treatment ratios, *hbal* finds its solution weights at a fraction of *kbal*'s runtime.

FIGURE A4. ADDITIONAL RUNTIME COMPARISON



A.4.6. Correlations with Inverse Propensity Scores

FIGURE A5. CORRELATION BETWEEN IPW, *ebal*, *ebal**, AND *hbal* WEIGHTS



Note: Points are from 10 simulations with sample size = 900 and control to treatment ratio = 5:1. $logit(e) = log(\frac{\pi(x)}{1-\pi(x)})$, where $\pi(x)$ are the true propensity scores. *ebal* weights are obtained from mean balancing on the level terms while *ebal** and *hbal* weights are obtained from balancing on the serially expanded covariates.

A.5. Additional Information on Black and Owens (2016)

A.5.1. Summary Statistics

TABLE A8. SUMMARY STATISTICS FOR CONTENDING JUDGES

	Mean	Median	St Dev	Min	Max
Vacancy Period					
Outcome: Ideological Vote with President	0.62	1.00	0.48	0.00	1.00
JCS Score	0.16	0.22	0.33	-0.70	0.58
Ideological Distance with President	0.17	0.12	0.18	0.00	1.04
Ideoloogical Composition of Panel	0.34	0.29	0.25	0.00	1.15
Median JCS (Circuit)	0.14	0.14	0.25	-0.60	0.69
Median JCS (Supreme Court)	0.01	-0.05	0.21	-0.36	0.41
Decision Reversal	0.28	0.00	0.45	0.00	1.00
Publication Status	0.61	1.00	0.49	0.00	1.00
Non-Vacancy Period					
Outcome: Ideological Vote with President	0.52	1.00	0.49	0.00	1.00
JCS Score	0.07	0.15	0.49	-0.70	0.58
Ideological Distance with President	0.45	0.44	0.49	0.00	1.18
Ideoloogical Composition of Panel	0.29	0.22	0.49	0.00	1.16
Median JCS (Circuit)	0.00	0.01	0.49	-0.69	0.69
Median JCS (Supreme Court)	0.03	0.06	0.49	-0.38	0.47
Decision Reversal	0.31	0.00	0.49	0.00	1.00
Publication Status	0.84	1.00	0.49	0.00	1.00

TABLE A9. SUMMARY STATISTICS FOR NON-CONTENDING JUDGES

	Mean	Median	St Dev	Min	Max
Vacancy Period					
Outcome: Ideological Vote with President	0.53	1.00	0.50	0.00	1.00
JCS Score	-0.07	-0.11	0.35	-0.69	0.61
Ideological Distance with President	0.53	0.50	0.34	0.00	1.33
Ideoloogical Composition of Panel	0.32	0.28	0.23	0.00	1.16
Median JCS (Circuit)	-0.05	-0.03	0.27	-0.69	0.58
Median JCS (Supreme Court)	-0.05	0.06	0.20	-0.36	0.23
Decision Reversal	0.34	0.00	0.47	0.00	1.00
Non-Vacancy Period					
Outcome: Ideological Vote with President	0.50	1.00	0.47	0.00	1.00
JCS Score	-0.05	-0.10	0.47	-0.69	0.61
Ideological Distance with President	0.47	0.44	0.47	0.00	1.33
Ideological Composition of Panel	0.31	0.27	0.47	0.00	1.16
Median JCS (Circuit)	-0.05	-0.07	0.47	-0.69	0.58
Median JCS (Supreme Court)	0.00	0.06	0.47	-0.38	0.23
Decision Reversal	0.34	0.00	0.47	0.00	1.00

A.5.2. Covariate Balance

FIGURE A6. COVARIATE BALANCE FOR CONTENDING JUDGES PRE- AND POST-WEIGHTING

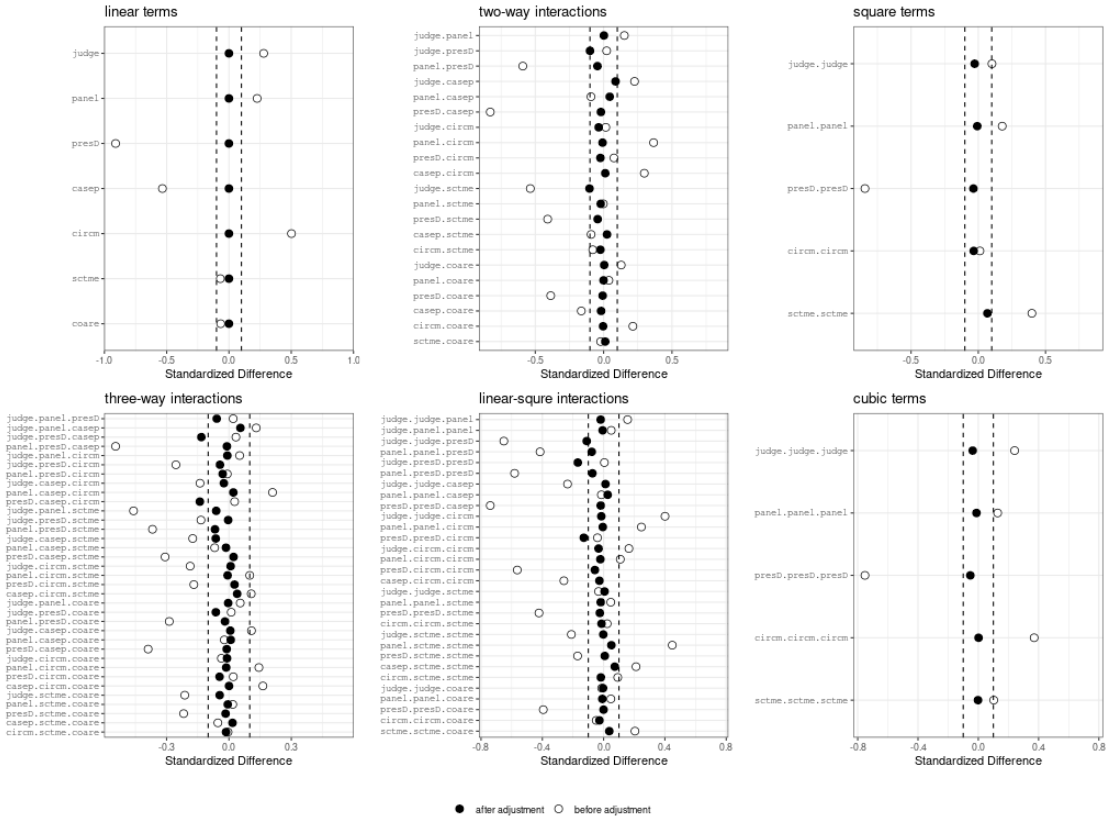
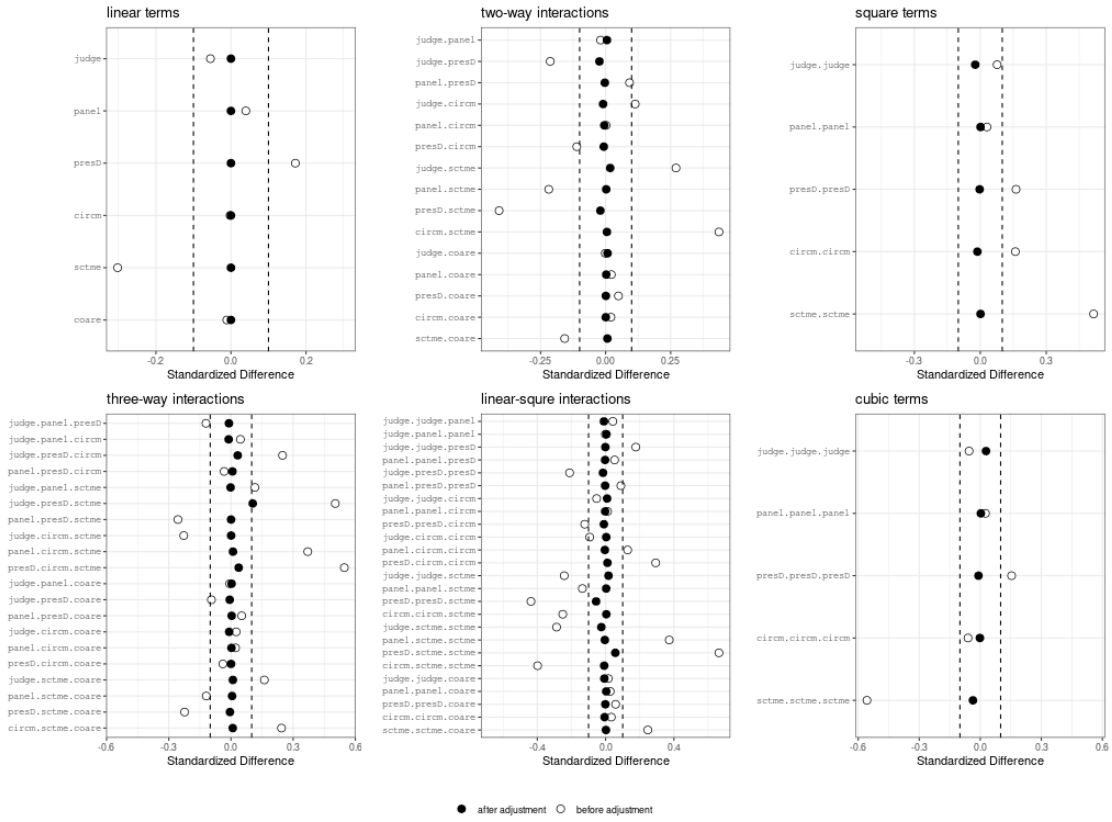


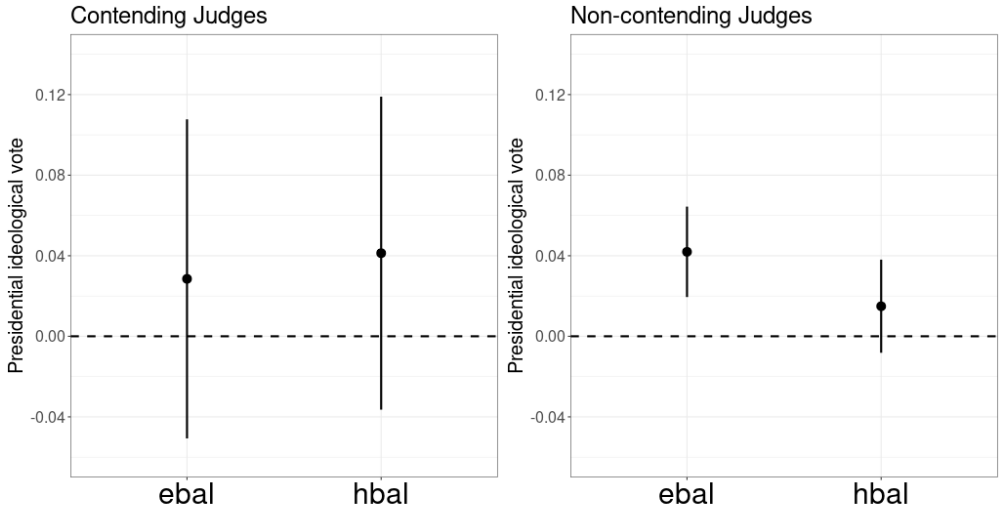
FIGURE A7. COVARIATE BALANCE FOR NON-CONTENDING JUDGES PRE- AND POST-WEIGHTING



A.5.3. Results From *ebal* and *hbal*

In contrast to the results in the main text, here we report ATT estimates and confidence intervals from *ebal* and *hbal* without using an outcome model. Similar to the results in the main text, *ebal* and *hbal* yield similar estimates for the contending judges while *ebal* gives higher estimate than *hbal* for the non-contending judges. Because we do not use an outcome model, the confidence intervals for the contending judges are wider, resulting in estimates from both *ebal* and *hbal* being statistically insignificant. For non-contending judges, we get the same conclusion as the main text - *ebal*'s estimate suggests non-contending judges tend to be more likely to rule in line with the president during a vacancy period, while *hbal*'s estimate shows no significant difference between the vacancy and non-vacancy periods.

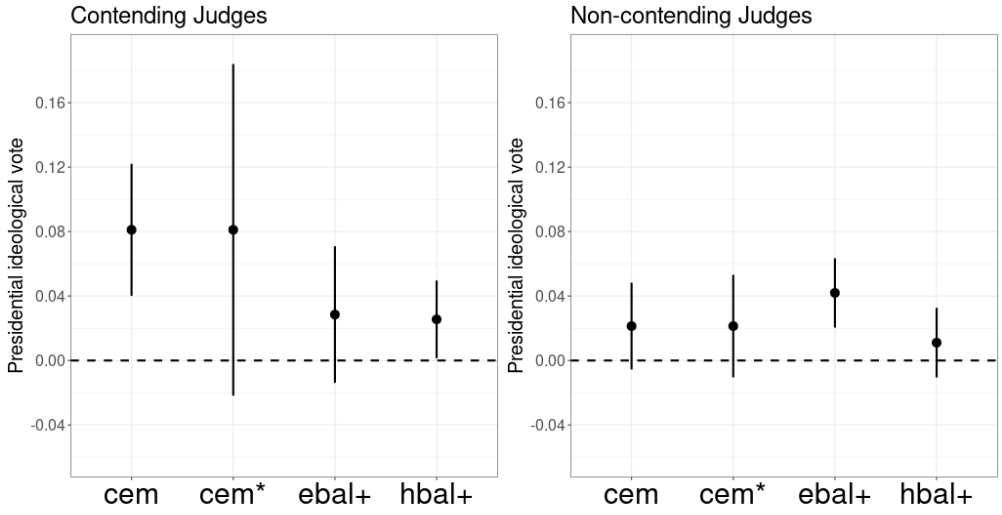
FIGURE A8. RESULTS WITHOUT OUTCOME MODELS



A.5.4. Comparison with the Original Results

In Black and Owens (2016), coarsened exact matching (CEM) (Iacus, King and Porro, 2012) was used to match circuit judges between vacancy and non-vacancy periods. Here we compare the estimates from CEM with those from *ebal* and *hbal+*. To enable direct comparison, we use a linear model with CEM weights, instead of the logistic model used in the original paper. In addition, we also report estimates using a linear regression with CEM weights and clustered, robust standard errors (cem*). For *ebal+* and *hbal+*, we also use clustered, robust standard errors. Point estimates with their confidence intervals are shown in Figure A9.

FIGURE A9. COMPARISON WITH ORIGINAL RESULTS



Note: *cem* uses standard OLS regression using the same specification as in Black and Owens (2016). *cem*, ebal+, habl+* use linear regression with clustered, robust standard errors.

As was reported in Black and Owens (2016), estimates from the matched data using CEM (cem) show that contending judges are significantly more likely to vote in line with the president's ideology during a vacancy period than a non-vacancy period. In contrast, the hypothesis of no difference is not rejected for the non-contending judges. However, when we take heteroskedasticity and clustering into account by using clustered, robust standard errors, the CEM estimate (cem*) for the contending judges is no longer significant at the 0.05 level. The uncertainties associated with CEM estimates are comparably larger than those of *ebal+* and *hbal+*, as for both contending and non-contending judge datasets, the CEM-matched datasets drop more than 80% of the observations.

A.6. The LaLonde Data

We also apply our method the canonical example of the LaLonde (1986) dataset. The experimental estimate of the effect of a job training program, the National Supported Work (NSW) program, is widely used as a benchmark for matching and weighting methods. Here we use Dehejia and Wahba (1999) subset of the experimental sample from LaLonde (1986) as the treated sample.

Following LaLonde (1986), the treated sample from the experimental study is compared to a control sample drawn from a separate, observational sample. The two commonly used control samples are from the Current Population Survey-Social Security Administration file (CPS-1) and the Panel Study of Income Dynamics (PSID-1). Since *ebal* has been shown to recover the experimental estimate well using data from CPS-1 as the control sample (Hainmueller, 2012), here we use the alternative PSID-1 data as the control sample and compare the performance of *ebal* and *hbal*.

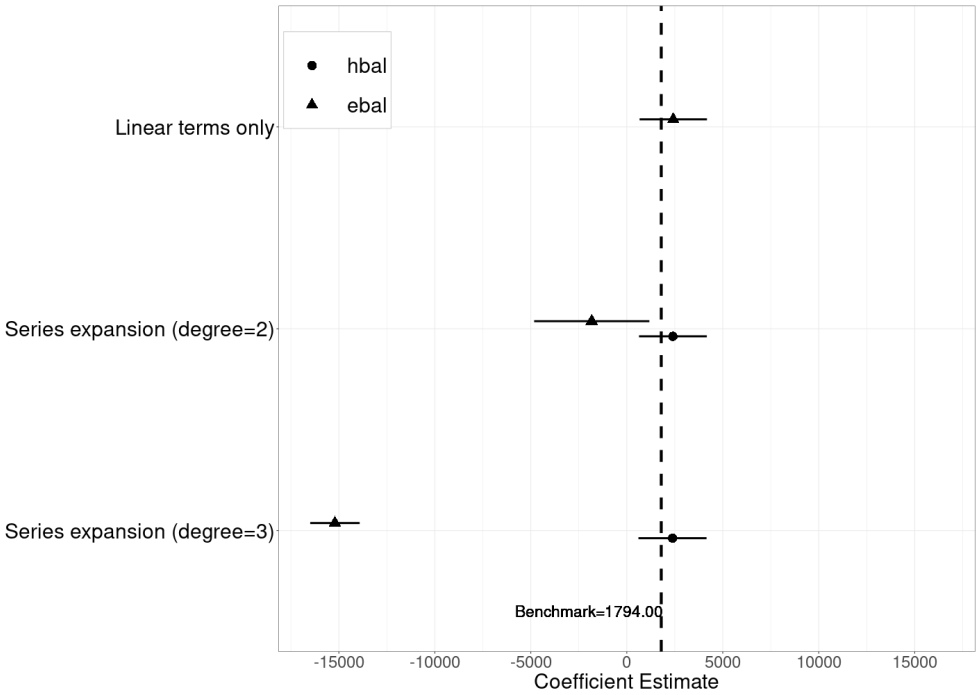


FIGURE A10. EFFECT OF TRAINING PROGRAM ON INCOME ($)

The treated sample from Dehejia and Wahba (1999) contains 185 NSW participants and the control sample from PSID-1 contains 2,490 nonparticipants. The outcome of interest is the post-treatment earnings in 1978. The experimental estimate of the effect of the NSW program on the treated units is $1,794, which is computed by difference-in-means in the original experimental data with the 185 treated units. Both the treated and control samples contain 10 pretreatment covariates that are used to control for selection into the training

program. These include age, education, real earnings in 1974 and 1975, and six indicator variables: Black, Hispanic, married, high school degree, and unemployment status in 1974 and 1975. To compare *ebal* and *hbal*, we use two specifications. For specification 1, we serially expand the 10 Raw covariates up the second degree. This includes the covariates' pairwise interactions, as well as square terms for the continuous variables–age, years of education, and real earnings in 1974 and 1975. For specification 2, we serially expand the 10 Raw covariates up the third degree. We drop all nonsensical (such as $Black * Hispanic$) terms. Overall, specification 1 includes 56 covariate combinations and specification 2 includes 184 covariate combinations. As linear regression runs inro rank deficiency issues, we instead use the solution weights to estimate a weighted difference in means.

Figure A10 reports the results from the two specifications, with estimates from using only the original covariates as reference. Across the two specifications, *hbal* recovers the experimental estimate fairly well. Estimate of the effect of the job training program is \$2,402 with a standard error of \$901 for specification 1 and \$2,381 with a standard error of \$905 for specification 2. Estimates from *ebal* vary much more widely. For specification 1, the estimated effect is -\$1,826 and for specification 2, the estimated effect is -\$15,205. The respective standard errors are \$1,531 and \$657.

A closer look at *ebal*'s optimization reveals that, for both specifications, it fails to find a set of solution weights that reduce the loss function below the specified tolerance level (0.001). For specification 1, the value of the loss function hovers around 0.27, with little improvement after 10 iterations. Similarly for specification 2, *ebal* fails to reduce the loss function below 123.89 after 200 iterations. The large losses in both specifications means that there is still substantial imbalance between the treated and the control sample after applying *ebal*. This results in the poor estimated effect from *ebal*. As is often the case when the covariate space is large and the sample size limited, *ebal* may not find a solution that satisfies the specified tolerance. Researchers are then left to either drop covariates until the tolerance can be met or accept the imbalance after preprocessing as is.

In contrast, *hbal*, by applying a hierarchical penalty to the covariates, keeps the imbalance of variables to a minimum while optimally choosing a penalty level. This not only reduces the variance of the estimator but also induces numerical stability that enables entropy balancing to more often find a solution when the covariate space is large and the sample size limited. As shown in Figure 2, *hbal* produces estimates close to the experimental benchmark and is fairly stable across specifications, even when the conditions are unfavorable to entropy-based methods.

References

Arkhangelsky, Dmitry, Susan Athey, David A. Hirshberg, Guido W. Imbens and Stefan Wager. 2018. "Synthetic Difference in Differences.".

Belloni, Alexandre, Victor Chernozhukov and Christian Hansen. 2014. "Inference on treatment effects after selection among high-dimensional controls." *The Review of Economic Studies* 81(2):608–650.

Black, Ryan C and Ryan J Owens. 2016. "Courting the president: how circuit court judges alter their behavior for promotion to the Supreme Court." *American Journal of Political Science* 60(1):30–43.

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

Deville, Jean-Claude and Carl-Erik Särndal. 1992. "Calibration estimators in survey sampling." *Journal of the American statistical Association* 87(418):376–382.

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

Iacus, Stefano M, Gary King and Giuseppe Porro. 2012. "Causal inference without balance checking: Coarsened exact matching." *Political analysis* 20(1):1–24.

Imbens, Guido, Phillip Johnson and Richard H Spady. 1995. "Information theoretic approaches to inference in moment condition models.".

Kullback, S. 1959. "Information theory and statistics Wiley." *New York* .

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

Little, Roderick JA and Mei-Miau Wu. 1991. "Models for contingency tables with known margins when target and sampled populations differ." *Journal of the American Statistical Association* 86(413):87–95.

Liu, Licheng, Ye Wang and Yiqing Xu. 2020. "A Practical Guide to Counterfactual Estimators for Causal Inference with Time-Series Cross-Sectional Data.".

Owen, Art B. 1988. "Empirical likelihood ratio confidence intervals for a single functional." *Biometrika* 75(2):237–249.

Powell, Michael JD. 1994. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in optimization and numerical analysis*. Springer pp. 51–67.

Qin, Jin and Jerry Lawless. 1994. "Empirical likelihood and general estimating equations." *the Annals of Statistics* 22(1):300–325.

Zhao, Qingyuan and Daniel Percival. 2016. "Entropy balancing is doubly robust." *Journal of Causal Inference* 5(1).