Microeconometrics I

Homework II

Professor: André Portela Student: Rafael F. Bressan

2020-10-05

Homework

You have been provided with a sample of 12,834 individuals in the labor force extracted from the 2019 annual supplement of the 2019 US Current Population Survey. Your goal is to estimate the causal effect of union membership/coverage (variable `union`) on weekly earnings (variable `earnings`). The dataset contains many other variables, which could potentially be used as controls (check the dataset dictionary).

1. As a starting point, compare average earnings among individuals with union coverage (`union==1`) vs individuals without such coverage (`union==0`). What is the estimated difference? Is it statistically significant? Do you think such a difference is a credible estimate for the causal impact of union coverage? Why?

First we notice there is an unbalance on the number of unionized and not unionized workers, the former being 1479 workers while the last is much larger, 11336[1]. The difference in average earnings is \$166. It is statistically significant by a t-test of difference in sample means, with different variances. The t-statistic is 9.3.

Since `union` is a dummy variable, this difference in means can be thought as a simple regression of `earnings` on `union` and the coefficient β_1 is the difference in means.

$$\text{earnings}_i = \beta_0 + \beta_1 \text{union}_i + \epsilon_i \tag{1}$$

This is clearly not a good estimator for the causal effect of union coverage on wages due to **selection bias**. There may be many hidden factors driving wages that are also correlated to the willingness of being part of a union, thus, the difference in average wage of unionized workers and not unionized workers is a *biased* estimate for the causal effect under study.

From now on, let's adopt the notation settled in Imbens and Rubin (2015) and let the individual observations be indexed by $i \in \{1, \ldots, N\}$. The potential outcomes of individual i are represented by $Y_i(0)$ if no treatment is taken and $Y_i(1)$ if the individual has been treated. Comparisons of $Y_i(1)$ and $Y_i(0)$ are unit-level causal effects, where we adopt the additive form, that is, the **individual causal effect** is defined as $Y_i(1) - Y_i(0)$. The response we can observe from an individual is $Y_i = Y_i(1)W_i + Y_i(0)(1 - W_i)$ for a treatment $W_i \in \{0, 1\}$. The **average treatment effect** - ATE, is $ATE = E[Y_i(1) - Y_i(0)]$, while the average treatment effect on the treated - ATT is $ATT = E[Y_i(1) - Y_i(0)|W_i = 1]$.

Thus, when we just compare the average earnings of workers, unionized and not, we are making the following estimation:

[1] After removing observations where there is no earnings information available, 19 individuals.

Table 1: Covariates chosen to model (2).

Variable	Description
age	Responsent's age
female	Is respondent a woman?
race	Respondent's race
marital_status	Respondent's marital status
veteran	Is respondent a veteran of US armed forces?
education	Respondent's maximum educational degree
class_of_worker	Worker's class

$$E[Y_i(1)|W_i=1] - E[Y_i(0)|W_i=0] = \underbrace{E[Y_i(1)|W_i=1] - E[Y_i(0)|W_i=1]}_{ATT} + \underbrace{E[Y_i(0)|W_i=1] - E[Y_i(0)|W_i=0]}_{\text{selection bias}}$$

and while the ATT is a valid estimation of causal effect, we don't have a clean figure of it when taking a simple difference of means, which will be plagued by selection bias.

2. **In order to improve and/or assess the credibility of your previous results, you decide to run a linear regression:**

$$\text{earnings}_i = \beta_0 + \beta_1 \text{union}_i + \gamma' Z_i + \epsilon_i \tag{2}$$

where Z_i are a set of controls.

a. Specify a linear model (2) by laying out a set of covariates Z to be included as controls. Justify your set of controls. What is the interpretation of β_1 in your model?

In case one wants to include covariates to the model, it's appropriate to make sure these variables help to explain variations in earnings, be that by theoretical modeling or empirical findings on earnings. With that qualification in mind, I have chosen the following variables to be in the covariates' set. Notice that nothing is being said about the covariates having a *causal effect* on earnings, but merely these variables help to explain variations in earnings, that is, they are correlated. Table 1 bellow presents the variables chosen and a brief description based on the dictionary provided.

Those variables were chosen based on previous literature relating earnings to social and economic factors, (Ashenfelter and Card 2010). Age is usually included in earnings regressions as a quadratic polynomial, reflecting the fact that there is an "optimal" age for earnings (or wages). Other variables like `female`, `race` and `veteran` follows from a literature that depicts prejudice or unfairness when setting wages. Education is a classic regressor for earnings, since it is believed the more educated a worker is, higher her productivity, and by a neoclassical argument, wages must reflect the marginal productivity of labor. The `class_of_worker` is included to capture some specificities from the demand side of labor, for example, due to imperfect competition on the final product markets, some firms may be more profitable than others, and part of this profitability is shared, through a Nash bargain, with its employees.

It shall be noted that variables `marital_status` and `race` originally had many levels and I opted to change them to binary variables. Thus, `marital_status` represent the married status if one and not married otherwise, although `race` was chosen to represent white if one and not white if zero, since white observations are much more common in this dataset.

Table 2: Missing data.

	Missings
V1	0
CPSID	0
CPSIDP	0
public_housing	8701
age	0
female	0
race	0
marital_status	0
veteran	83
employed	0
education	0
worked_last_year	0
total_income_last_year	0
wage_income_last_year	0
own_farm_income_last_year	0
private_health_insurance	0
medicaid	0
class_of_worker	0
class_of_worker_last_year	0
union	0
earnings	19

I have chosen not to include variables like `worked_last_year` and `class_of_worker_last_year` because, while they may be related to current earnings, these are essentially variables that capture model dynamics. Since I am interested in cross-sectional results, lagged variables would have a different interpretation and would capture dynamic factors, like persistence in earnings, but not economic or social traits.

Adding covariates to the model helps making the case for a causal interpretation of β_1, but it is debatable whether we are including all relevant variables or not. A causal interpretation is due **only** if we control for all factors that affect earnings and are correlated to the worker's choice of entering the union. This is *unlikely* to be true in the current setup with a limited number of variables, thus, it is still uncertain one can make causal inference with model (2). That is, the main assumption to have a causal interpretation from a regression model is that, given a set of covariates, treatment and potential outcomes are independent,

$$Y_i(1), Y_i(0) \perp W_i | X_i$$

and while in an observational study we cannot guarantee this assumption, and not even test it, just using a limited set of covariates X_i in our regression model is not the best way to achieve causal interpretation.

b. Estimate the specified model. What is your estimate of β_1? Is it significant? Briefly comment on your results.

Once chosen the covariates, we have a dataset of one dependent variable, `earnings` and eight regressors, the treatment `union` and seven selected controls from table 1. Before attempting to make a regression from this data, we must first make sure we don't have any missing observations. The first step is to exclude from our dataset, any row where `earnings` is missing, there were 19 such rows. Next we investigate further if any other regressors don't have observations. Table 2 shows the resutls.

Table 3: Regressors categorization.

class_of_worker	class_of_worker_last_year
21:10741	0 : 309
25: 413	13: 22
27: 722	14: 109
28: 939	22:10336
	25: 407
	27: 732
	28: 899
	29: 1

There are 83 missing values for veteran status. Since it is much more likely not to be a veteran[2], the choice to impute zero as the value for the missings is appropriate. The option would be to discard such observations, but I'd rather preserve observations that otherwise are complete. The story is different for `public_housing` which had 8,701 missing observations, the majority of our dataset. Therefore, this variable was readily discarded.

Now that we have imputed values for missing veteran status, we shall notice that some of our included variables are categorical in nature. Except for `age`, which is an integer number, all other regressors are categorized according to numerical codes. Even though being categorical variables, `female` and `veteran` can be interpreted as numerical since they present only two possible values. The case for `education` is a bit more complex. The way this variable is included in our dataset, it is a categorical one but, the codes associated with it may have an ordinal interpretation, since a code of, say 70 represents a higher level of education than code 60, and so on. Hence, `education` is kept as a numerical variable in our regressions. Truly categorical, not having an ordinal interpretation are `class_of_worker`, `marital_status`, and `race`. The last two variables were taken care by aggregating the codes in a binary format, but for example, code 20 versus code 28 in `class_of_worker` should not be taken as 20 is lower than 28 as they are just codes for classification of different types of employment. Hence, our approach is to set dummy variables for each level such a categorical variable may take, a procedure known as one-hot encoding. When doing this categorization, one must be careful to have enough observations in any such level. Table 3 shows the results of this categorization.

From this table 3 one can see how level 29 for `class_of_worker_last_year` has only one observation and since this level represents, according to the dictionary provided, "Unpaid family worker", I have changed this observation to level 0, "Not in universe (did not work)".

With a cleaned, prepared and meaningful dataset we are now able to perform the regression in model (2). The coefficient found on variable `union` is the impact of being unionized on our *baseline* individual, that is, the individual in the first line of table 3, meaning a not unionized, male, white, married with spouse present and so on. If we are interested in the effect of joining a union for other representative individuals, we should include the interaction of union to all other covariates and analyze the result.

From the results of table 4 we find that β_1 for the estimated model is significant and has a value of 94, which is lower than the simple difference of means found in model (1). We are still reluctant to give a causal interpretation to the coefficient β_1 in model (2) since it is not at all clear we have controlled for every factor that affects earnings and are correlated to the choice of joining a labor union.

Using the Frisch-Waugh-Lovell theorem, we can show that the OLS estimator for β_1 has the representation:

[2]The ratio of veterans to non-veterans in this dataset is 0.061. Also, we modeled the veteran status with a probit regression and the imputation result were the same, every missing filled by zero.

Table 4: Results from regressions.

	Model 1	Model 2
union	165.689***	94.330***
	(17.828)	(17.112)
5	12815	12815
7	0.005	0.339

* p < 0.1, ** p < 0.05, *** p < 0.01
Note: White corrected standard errors in parentheses.

Table 5: Summary statistics for weights.

Statistic	Not Union	Union
Mean	0.000088	0.000676
Median	0.000067	0.000764
SD	0.000080	0.000128
Min	-0.000008	0.000459
Max	0.000400	0.000860
sum	1.000000	1.000000

$$\hat{\beta}_1 = \sum_{i=1}^{N} \text{union}_i \cdot \omega_i \cdot \text{earnings}_i - \sum_{i=1}^{N} (1 - \text{union}_i) \cdot \omega_i \cdot \text{earnings}_i \tag{3}$$

where weights ω_i are:

$$\omega_i = \frac{\hat{\xi}_i \, (2\text{union}_i - 1)}{\text{SSR}_{\text{union},Z}} \tag{4}$$

with $\hat{\xi}_i$ being the residual of observation i from a linear regression of union $_i$ on Z, including an intercept; and SSR union, Z is the sum of squared residuals of this auxiliary regression.

c. compute the weights for your specification using the formula above. Report summary statistics for the distribution of weights in the control and treatment groups. Do the weights sum to one in the control group? What about the treatment group? Are there any negative values? What about outliers? How do these weights compare with those from other estimators you have seen in class (e.g. Horvitz-Thompson)? Why? Hint: Section III of Imbens G. Matching Methods in Practice. Journal of Human Resources, 2015 ; 50(2): 373-419

Table 5 presents summary statistics for weights by control and treatment groups, not in union and unionized respectively.

The sum of weights are 1 for both control and treatment groups. There are negative values for the weights in the control group, although negative values should not come as a surprise according to Imbens (2015) . As for outliers, we better make a boxplot, shown in figure 1.

The Horvitz-Thompson – HT – estimator can be written as:

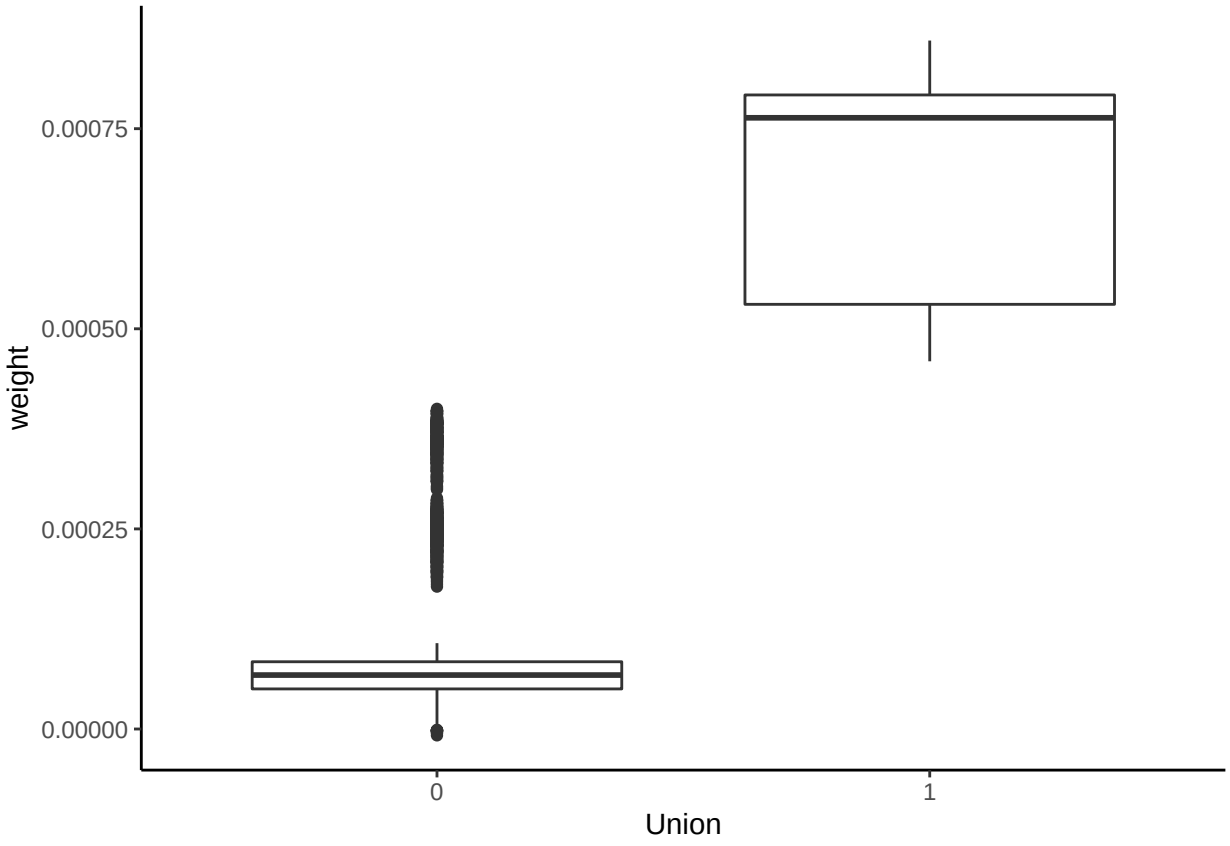

Figure 1: Box-plot of weights.

$$\hat{\beta}_1^{\text{ht}} = \frac{1}{N}\sum_{i=1}^{N}\frac{\text{union}_i \cdot \text{earnings}_i}{e\left(Z_i\right)} - \frac{1}{N}\sum_{i=1}^{N}\frac{(1-\text{union}_i)\cdot \text{earnings}_i}{1-e\left(Z_i\right)} \tag{5}$$

where $e(Z_i)$ is the propensity score given the set of covariates Z_i. Thus, the HT estimator is weighting the regression by a measure that is the inverse of the probability of being assigned to the observed treatment (i.e. active treatment or control treatment). If the population propensity score is known, then the weights for treatment and control are:

$$\omega_i^{\text{ht}} = \begin{cases} 1/N \cdot (1 - e\left(Z_i\right)) & \text{if union}_i = 0 \\ 1/N \cdot e\left(Z_i\right) & \text{if union}_i = 1 \end{cases}$$

This contrasts to the weights assigned by the FWL method, which are related to the inverse of the treatment's variance that is *not* explained by the covariates, that is, the treatment's residual variance in the auxiliary regression.

$$\omega_i = \begin{cases} -\hat{\xi}_i/\text{SSR}_{union,Z} & \text{if union}_i = 0 \\ \hat{\xi}_i/\text{SSR}_{union,Z} & \text{if union}_i = 1 \end{cases}$$

3. State a causal estimand of interest (ATT or ATE) and the assumptions required for the identification of this effect on a selection-on-observables framework. Explain why you require these assumptions.

We can state the average treatment effect – ATE – and the average treatment effect on the treated – ATT – on the population as:

$$\tau_{ate} = E[Y_i(1) - Y_i(0)] \tag{6}$$
$$\tau_{att} = E[Y_i(1) - Y_i(0)|W_i = 1] \tag{7}$$

In the case where the assumptions we'll depict next, hold only after conditioning on a set of covariates, Z_i, the conditional average treatment effects take the form:

$$\tau_{cate}(z) = E[Y_i(1) - Y_i(0)|Z_i = z] \tag{8}$$
$$\tau_{catt}(z) = E[Y_i(1) - Y_i(0)|Z_i = z, W_i = 1] \tag{9}$$

and the estimands from equations (6) and (7) are computed from taking the expected values of the conditional couterparts over the distribution of Z.

The main assumption for this simple characterization is the validity of the so called SUTVA – stable unit treatment value – that incorporates both the idea that units do not interfere with one another and that for each unit there is only a single version of the active treatment.

Assumption (SUTVA) *The potential outcomes for any unit do not vary with the treatments assigned to other units, and, for each unit, there are no different forms or versions of each treatment level, which lead to different potential outcomes.*

First, SUTVA assumes no-interference of one unit's treatment on other unit's outcome, that is, no spillovers are assumed. Second, the individual receiving a treatment (control or active) cannot receive treatments of different efficacy that affects the outcome.

Given SUTVA, there are basically three main assumptions made on the assignment mechanism for identification of causal effects, (Imbens and Rubin 2015).

1. Individualistic assignment: This limits the dependence of a particular unit's assignment probability on the values of covariates and potential outcomes for other units.

An assignment mechanism $\Pr(\mathbf{W} \mid \mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1))$ is individualistic if, for some function $q(\cdot) \in [0, 1]$,

$$p_i(\mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1)) = q\left(X_i, Y_i(0), Y_i(1)\right), \text{ for all } i = 1, \ldots, N$$

and

$$\Pr(\mathbf{W} \mid \mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1)) = c \cdot \prod_{i=1}^{N} q\left(X_i, Y_i(0), Y_i(1)\right)^{W_i} \left(1 - q\left(X_i, Y_i(0), Y_i(1)\right)\right)^{1-W_i}$$

for $(\mathbf{W}, \mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1)) \in \mathbb{A}$, for some set $\mathbb{A}$, and zero elsewhere (c is the constant that ensures that the probabilities sum to unity).

2. Probabilistic assignment: This requires the assignment mechanism to imply a nonzero probability for each treatment value, for every unit.

An assignment mechanism $\Pr(\mathbf{W} \mid \mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1))$ is probabilistic if the probability of assignment to treatment for unit i is strictly between zero and one:

$$0 < p_i(\mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1)) < 1, \text{ for each possible } \mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1)$$

for all $i = 1, \ldots, N$.

3. Unconfounded assignment: This disallows dependence of the assignment mechanism on the potential outcomes.

An assignment mechanism is unconfounded if it does not depend on the potential outcomes:

$$\Pr(\mathbf{W} \mid \mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1)) = \Pr\left(\mathbf{W} \mid \mathbf{X}, \mathbf{Y}'(0), \mathbf{Y}'(1)\right)$$

for all $\mathbf{W}, \mathbf{X}, \mathbf{Y}(0), \mathbf{Y}(1), \mathbf{Y}'(0)$, and $\mathbf{Y}'(1)$

The unconfounded assumption is also known as the conditional independence assumption – CIA, (Angrist and Pischke 2008). Thus, if the assignment mechanism is uncounfounded and individualistic the probability of assignment is the *individual* propensity score. Also, given individualistic assignment, a mechanism that is both probabilistic and uncounfounded is referred as *strongly ignorable treatment assignment.*

4. **Report balance checks (t-stats and normalized differences) for *a priori* relevant (for identification) covariates in the treatment and control group. Are these covariates balanced between groups?**

Table 6: Balance of categorical variable 'class of worker'.

	Not Union		Union	
Level	N	Percent	N	Percent
21	9978.00	88.02	763.00	51.59
25	297.00	2.62	116.00	7.84
27	514.00	4.53	208.00	14.06
28	547.00	4.83	392.00	26.50

Table 7: Balance of numerical variables.

Variable	Mean Control	Mean Treat.	t-stat	Norm. Diff.
age	42.06	45.70	9.74	0.26
age_2	1985.41	2266.80	8.14	0.22
female	0.50	0.49	-0.89	-0.02
race	0.81	0.79	-1.93	-0.05
marital_status	0.54	0.64	7.65	0.21
veteran	0.06	0.07	2.70	0.08
education	90.63	94.76	6.80	0.18

From tables 6 and 7 one can see that numerical variables are well balanced (based on normalized difference) between control and treatment groups, although some improvement can be made, but categorical variables have significant proportion differences.

5. **Estimate a propensity score model for union using logistic regression. State the variable selection method you will use (e.g. "I'll use Imbens and Rubin's stepwise selection algorithm, taking ... as base variables, and letting their method select ..."). Comment on your results. What is the normalized difference of the latent indices of the logistic model, $X_i'\hat{\kappa}$, in the treatment vs control group?**

We will use three methods for variable selection and compare the results. The first method is the one proposed by Imbens (2015). We have selected the same variables as in model (2), and presented in table 1, to be the basic covariates X_B, while all other meaningful variables were left to be chosen by the algorithm[3]. Second order terms were left to be chosen. Notice that variables `total_income_last_year` and `wage_income_last_year` are bad controls, since they are likely to be affected by the union status, and nothing in our metadata says these variables were collected **before** the worker had joined the union, if that is the case. Therefore, these two variables are also discarded from our selection process *for all three algorithms.*

The second algorithm of choice was the logit lasso based on Belloni, Chernozhukov, and Hansen (2014) and implemented in the R package `hdm`. We left the algorithm choose any variable deemed meaningful and their interactions up to the second order without imposing any "must have" variable.

Finally, the third model for estimating the propensity score was a *full* logit model with all meaningful variables and their interactions up to second order. The basic logit model for `union` can be written as:

$$\text{union}_i = \ell(\mathbf{Z}_i'\boldsymbol{\beta}) + \varepsilon_i \tag{10}$$

$$\ell(x) = \frac{e^x}{1+e^x} \tag{11}$$

where $\mathbf{Z}$ is the transformed dataset, possibly including all covariates and their second order terms and interactions. The variables in $\mathbf{Z}$ are selected by the above algorithms.

Table 8 shows the latent indices mean values, variances and the normalized difference for each selection model by treatment. In Annex A we present the list of selected variables and their interactions by selection model. We can see by the normalized difference that Lasso did a good job at making apart the control and treatment units. While the indices' average value are quite different between groups, their standard deviation is very low compared to other algorithms, thus yielding a good model for predicting the union status.

Although this result is desirable from a prediction point of view, the Lasso algorithm may have little overlap of propensity scores between control and treatment groups. This is specially troublesome if we end up with

[3]Variables "V1", "CPSID", "CPSIDP", "public_housing", "employed" were deemed not meaningful for the regression modeling. Besides, "own_farm_income_last_year" had outlier problems and also, was not included in the algorithms.

Table 8: Normalized difference among latent indices for different logistic models specification.

	Not Union		Union		Statistic
Logistic Model	Mean	Var	Mean	Var	Norm. Diff.
Full	-2.6213	2.2889	-1.4255	1.5432	0.8639
Imbens-Rubin	-2.5947	2.2115	-1.4418	1.5175	0.8444
Lasso	-2.4113	0.5208	-1.5448	1.1118	0.9589

many treatment units without a control pair in terms of propensity score. If that is the case, our next step would be trim those treatment units from our sample and, depending on the number of treated individuals in the sample, this may be undesirable. Figure 2 shows histograms for propensity scores from which we can visually analyze the common support depending on the algorithm chosen.

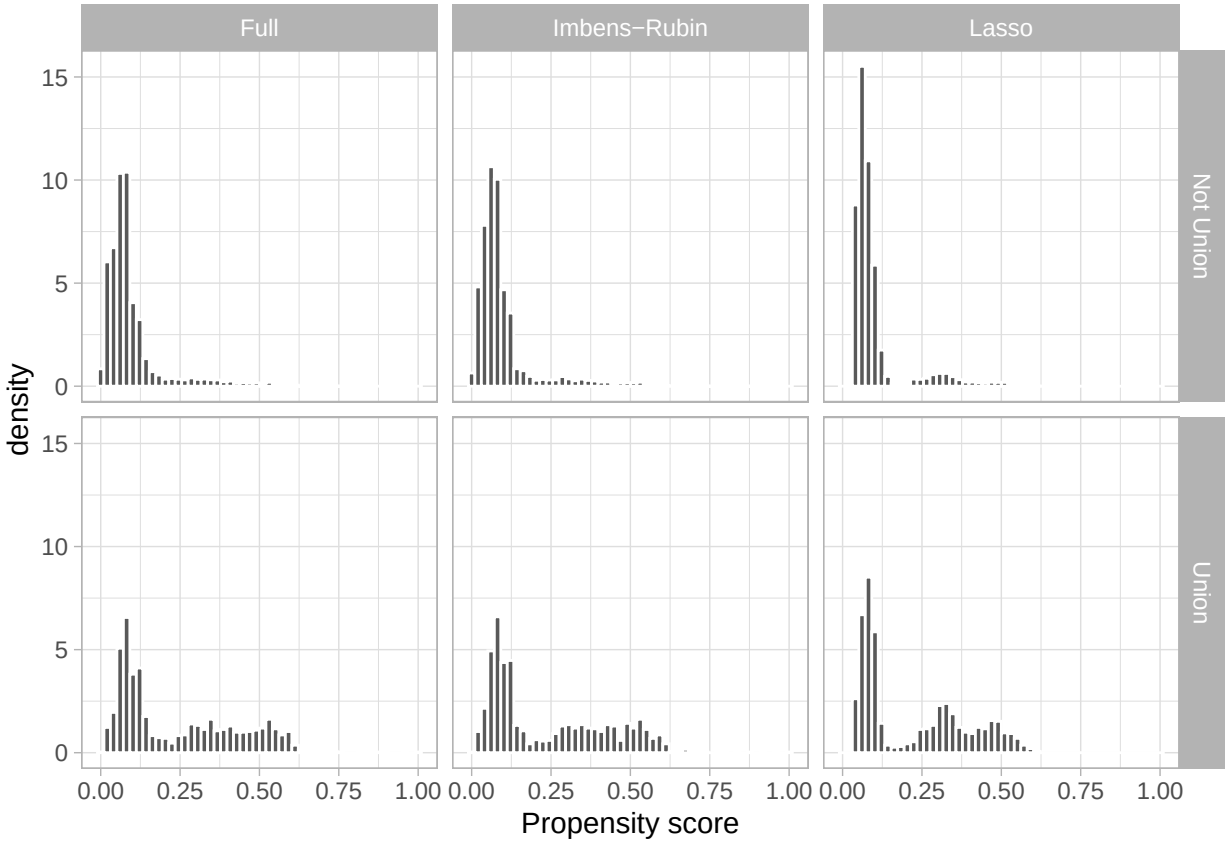

Figure 2: Distribution of propensity score by treatment and logit model

As expected, the Lasso's control group has very narrow range, leaving a great number of treatment units without any control pair. The full model and Imbens-Rubin have similar profiles, thus similar trimming points are expected. Although, as a side note on performance to fit those models, the algorithm IR is much **slower** than fitting a fully specified model.

6. **Assess the quality of your estimated propensity score by verifying its balancing properties (e.g. dividing dataset in blocks using Imbens and Rubin's approach and verifying covariate balance within each block).**

There are many NaN values for covariates that are categorical in nature (and were transformed to numeric

Table 9: Balance assessment by blocks for IR model.

	Blocks										
covar	1	2	3	4	5	6	7	8	9	10	11
age	1.21	-1.03	-0.71	-0.32	0.58	1.64	-0.48	-0.33	-3.11	0.98	-0.09
female	-0.06	-0.33	0.79	-0.16	1.16	-1.01	-0.20	1.60	-0.38	0.24	-1.98
race	-0.96	0.13	0.28	0.02	0.00	-0.33	-0.47	1.93	0.38	-0.39	0.34
marital_status	0.24	-0.82	0.37	-0.58	0.89	-0.34	0.09	0.82	-0.01	-1.50	-0.53
veteran	0.94	-0.82	0.65	-1.81	-0.08	0.32	0.12	0.09	-1.90	0.29	2.77
education	-0.14	0.54	0.66	-0.53	1.54	0.05	-0.21	-0.27	1.51	1.01	-0.42
worked_last_year	0.98	-0.74	-0.69	3.33	-0.33	0.11	-0.41	NaN	NaN	NaN	NaN
total_income_last_year	0.97	-2.97	-0.16	-1.60	2.90	-1.02	1.94	-0.02	1.12	-0.06	0.73
wage_income_last_year	0.82	-2.84	0.00	-0.71	2.24	-1.51	2.20	1.03	1.18	0.19	0.55
private_health_insurance	-0.84	0.45	-0.10	1.52	-0.59	0.46	0.32	-0.62	-0.04	-1.77	-0.80
medicaid	0.83	0.96	-0.61	-0.39	-2.74	-0.83	0.15	1.16	0.62	-0.58	0.65
class_of_worker21	3.17	0.12	-0.51	-0.29	-1.42	-1.33	0.58	-1.42	1.00	NaN	NaN
class_of_worker25	-2.00	0.27	0.49	0.67	0.60	-1.07	0.11	1.18	-1.62	0.00	1.24
class_of_worker27	-1.41	-2.24	-2.65	-0.63	0.96	0.73	0.18	-1.03	-0.95	-1.00	0.09
class_of_worker28	-2.00	-1.00	0.83	0.15	0.64	0.47	-0.42	0.09	1.74	0.29	-1.10
class_of_worker_last_year13	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
class_of_worker_last_year14	1.00	-3.61	-1.73	0.38	-0.57	-1.42	-0.06	-1.42	1.00	NaN	1.76
class_of_worker_last_year22	-0.02	-0.80	-0.11	1.35	-0.72	-1.25	0.13	0.01	-1.75	0.00	0.80
class_of_worker_last_year25	-5.69	-0.18	0.74	0.88	0.88	-0.56	-0.05	1.15	-1.62	0.00	0.76
class_of_worker_last_year27	0.38	1.19	-0.25	-1.34	0.51	0.84	-0.50	-0.82	1.00	NaN	0.06
class_of_worker_last_year28	0.91	-0.12	-3.48	-1.82	-0.51	1.08	0.46	-0.07	1.68	0.00	-1.57
age_2	1.24	-1.08	-0.84	-0.26	0.57	1.56	-0.53	-0.37	-3.01	1.05	0.16

Note: Values presented are the t-statistic from the difference of means between treatment and control units.

Table 10: Balance assessment by blocks for Lasso model.

	Blocks											
covar	1	2	3	4	5	6	7	8	9	10	11	12
age	1.27	0.07	0.23	0.28	1.05	-0.15	-1.51	1.23	1.51	-0.92	-0.80	-1.74
female	-0.18	0.45	0.78	-0.40	-1.14	-1.54	1.42	-1.69	-0.36	0.34	0.52	-1.42
race	-0.99	-0.03	-1.15	0.03	-0.73	0.06	-1.79	-0.78	0.04	0.25	0.66	-0.80
marital_status	-0.89	0.77	-0.24	0.65	-1.69	-0.53	0.07	0.49	-0.26	-0.66	-0.51	-0.38
veteran	1.13	-0.65	0.88	0.59	-0.97	-0.13	-0.87	0.86	1.02	1.66	-2.31	-1.06
education	0.76	0.21	-1.68	-2.14	-2.77	-3.09	-2.04	-0.60	-1.20	-1.87	-0.71	-0.05
worked_last_year	-0.81	1.50	4.61	0.26	2.67	1.74	-0.57	-0.39	NaN	NaN	NaN	NaN
total_income_last_year	1.33	0.51	-1.15	-2.15	-1.44	-2.85	-2.08	0.11	-0.40	0.69	0.73	0.95
wage_income_last_year	1.40	0.62	-0.96	-2.19	-1.23	-2.31	-2.01	0.59	-0.72	0.99	0.67	1.19
private_health_insurance	-1.23	1.90	0.87	-0.71	2.86	0.06	-0.98	0.52	-0.41	-0.15	-1.00	0.70
medicaid	2.60	-0.86	-2.19	0.84	-0.20	-2.67	-4.22	-0.93	0.81	1.72	NaN	1.00
class_of_worker21	1.00	2.45	1.41	-0.06	3.04	0.06	-0.67	-1.34	NaN	NaN	NaN	NaN
class_of_worker25	NaN	NaN	NaN	0.45	-2.86	0.14	0.80	1.34	-1.49	0.32	0.00	-0.50
class_of_worker27	-1.00	-2.24	-1.41	-1.41	-1.00	-1.42	-3.52	-0.17	1.13	-0.20	-0.87	-1.31
class_of_worker28	NaN	-1.00	NaN	-1.00	NaN	NaN	0.89	-0.59	0.34	-0.12	0.93	1.45
class_of_worker_last_year13	0.92	-2.24	-1.73	-2.00	-1.42	-1.42	-1.00	-1.00	NaN	NaN	NaN	-1.00
class_of_worker_last_year14	-2.83	-3.61	-2.24	0.38	-3.20	-1.03	-0.54	NaN	-1.00	1.36	NaN	1.43
class_of_worker_last_year22	-1.31	1.94	2.62	0.90	1.51	0.67	-0.97	1.58	-0.47	-0.52	-1.95	0.10
class_of_worker_last_year25	-1.41	-2.65	-2.24	-0.78	-0.33	0.73	0.75	0.69	-0.54	0.43	0.32	-0.21
class_of_worker_last_year27	1.04	0.46	0.25	-0.44	0.33	-4.47	0.07	-0.25	0.27	-0.72	-0.87	-1.71
class_of_worker_last_year28	-1.73	-3.32	-1.41	-0.94	-0.47	0.80	0.65	-5.01	0.95	0.28	1.76	1.01
age_2	1.12	-0.16	0.32	0.06	0.85	0.00	-1.82	0.84	1.41	-0.95	-0.68	-1.79

Note: Values presented are the t-statistic from the difference of means between treatment and control units.

Table 11: Balance assessment by blocks for Full model.

	Blocks							
covar	1	2	3	4	5	6	7	8
age	0.85	-0.20	0.41	-0.11	0.21	1.70	-1.69	-0.35
female	-0.25	0.94	0.40	-0.61	0.99	-0.91	0.29	0.34
race	-1.60	1.54	-0.90	0.62	-0.29	-0.16	0.90	-0.28
marital_status	-0.14	0.02	-0.25	0.45	0.48	-0.25	1.53	-1.89
veteran	0.96	-0.56	-0.32	-0.79	-0.25	0.46	-0.28	1.34
education	0.44	-0.72	2.09	-1.15	0.96	0.02	-0.81	1.98
worked_last_year	1.11	-1.26	-0.25	0.54	-0.48	1.74	-0.19	1.00
total_income_last_year	1.62	-3.53	0.15	-1.87	1.85	0.87	0.19	1.37
wage_income_last_year	1.54	-3.30	0.17	-1.12	1.25	0.69	1.19	1.24
private_health_insurance	-0.21	0.50	-0.39	0.34	0.40	1.05	-0.78	-2.61
medicaid	-0.47	0.89	1.50	-1.25	-1.02	-0.95	-0.83	1.66
class_of_worker21	-0.36	4.71	3.01	-1.35	-0.21	-2.65	-0.36	-0.16
class_of_worker25	-3.61	-3.00	-2.00	0.53	0.98	-0.25	-0.11	0.10
class_of_worker27	-2.24	-3.00	-2.24	0.51	0.10	0.37	-0.32	0.43
class_of_worker28	0.94	-2.00	NaN	1.22	-1.10	0.52	0.40	-0.26
class_of_worker_last_year13	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
class_of_worker_last_year14	0.85	-3.47	-1.73	0.26	-0.51	-0.63	-1.42	1.74
class_of_worker_last_year22	-0.36	-0.66	1.08	0.29	-0.17	-0.93	-0.31	0.04
class_of_worker_last_year25	-0.05	0.14	-2.00	0.53	0.59	-0.11	0.58	-0.16
class_of_worker_last_year27	0.83	0.69	0.01	-1.37	0.29	0.05	-0.48	0.56
class_of_worker_last_year28	0.76	-4.60	-4.02	0.45	-1.10	1.22	0.21	-0.24
age_2	0.83	-0.31	0.28	0.22	0.05	1.64	-1.87	-0.14

Note: Values presented are the t-statistic from the difference of means between treatment and control units.

Table 12: Observations for selected levels from categorical covariates.

Block	CW-21	CW-25	CW-27	CW-28	CW-13	CW-14	CW-22	CW-25	CW-27	CW-28
1	3154	4	2	4	0	20	2878	32	20	24
2	3146	12	5	1	0	13	3055	19	17	18
3	1573	7	7	3	0	3	1541	4	16	12
4	1550	18	15	8	0	6	1483	15	49	27
5	1197	151	147	95	0	10	1193	143	141	81
6	14	63	214	106	0	2	93	61	194	43
7	4	42	241	111	0	3	39	29	233	92
8	2	57	61	277	0	2	22	51	50	272
9	1	24	9	165	0	1	3	24	1	170
10	0	12	1	87	0	0	6	12	0	82
11	0	14	6	80	0	3	5	15	3	74

Table 13: Balance after trimming. Model IR.

Covariate	Mean Control	Mean Treatment	t-stat	Norm.Diff.
age	44.41	46.34	5.18	0.14
female	0.43	0.47	2.90	0.08
race	0.80	0.79	-0.93	-0.03
marital_status	0.60	0.66	4.53	0.13
veteran	0.06	0.08	2.00	0.06
education	92.47	95.30	4.56	0.13
worked_last_year	0.99	0.99	1.53	0.04
total_income_last_year	0.07	0.08	0.22	0.01
wage_income_last_year	0.07	0.09	0.50	0.01
private_health_insurance	0.87	0.93	6.87	0.18
medicaid	0.07	0.05	-3.26	-0.09
age_2	2165.02	2314.90	4.29	0.12

format by one-hot encoding of dummies), this is probably due to a lack of observations of that specific level in the block. This should not be taken as lack of balanace for the covariate itself.

Indeed many strata of `class_of_worker` type of covariate do not have any observation for a given block, thus the statistics can't be computed, as presented in table 12 for the IR model only. But this is mainly due to the attribution for the NA block, which we are removing from balance assessment.

7. **Use Imbens and Rubin's approach (Chapter 16) to trim your dataset in order to improve overlap. Rereport the results in (4). What happened to them? What about the normalized difference of the latent indices of the logistic model?**

The trimming method supressed 2018, 456, 2331 observations from our sample for models IR, Lasso and Full, respectively. We can see from tables 13 to 15 that our numerical covariates had improved balances, measured by the normalized differece.

Comparison from before and after trimming for logistic model's latent indices is found at table 16, and we can observe the normalized difference had different responses depending on the model. Lasso had a slight improvement, while the Full model and specially Imbens-Rubin model had their normalized differences raised.

For our categorical variable of choice, `class_of_worker` the balance had only a minor improvement, and

Table 14: Balance after trimming. Model Lasso.

Covariate	Mean Control	Mean Treatment	t-stat	Norm.Diff.
age	42.22	45.75	9.44	0.25
female	0.48	0.48	0.26	0.01
race	0.81	0.79	-1.41	-0.04
marital_status	0.56	0.65	6.40	0.18
veteran	0.06	0.07	2.45	0.07
education	91.12	94.96	6.31	0.17
worked_last_year	0.97	0.99	5.11	0.12
total_income_last_year	0.01	0.06	2.37	0.06
wage_income_last_year	0.01	0.07	2.61	0.07
private_health_insurance	0.83	0.91	10.00	0.25
medicaid	0.07	0.05	-4.45	-0.11
age_2	1995.41	2269.53	7.91	0.22

Table 15: Balance after trimming. Full model.

Covariate	Mean Control	Mean Treatment	t-stat	Norm.Diff.
age	44.45	46.36	5.11	0.14
female	0.44	0.47	2.58	0.07
race	0.80	0.79	-0.71	-0.02
marital_status	0.61	0.67	4.16	0.12
veteran	0.06	0.08	2.29	0.07
education	92.77	95.42	4.25	0.12
worked_last_year	0.99	0.99	1.61	0.04
total_income_last_year	0.08	0.08	-0.24	-0.01
wage_income_last_year	0.09	0.09	0.04	0.00
private_health_insurance	0.88	0.93	5.65	0.15
medicaid	0.08	0.05	-4.08	-0.11
age_2	2166.10	2315.84	4.27	0.12

Table 16: Balance for logit's latent indices.

	Normalized difference	
Model	Before trimming	After trimming
Full	0.86	0.98
Imbens-Rubin	0.84	0.97
Lasso	0.96	0.94

even with the trimming procedure, one cannot say this variable is balacend across union status.

Table 17: Balance of categorical variable 'class of worker'. IR model.

	Not Union		Union	
Level	N	Percent	N	Percent
21	8046.00	85.80	703.00	49.54
25	287.00	3.06	116.00	8.17
27	501.00	5.34	208.00	14.66
28	544.00	5.80	392.00	27.63

Table 18: Balance of categorical variable 'class of worker'. Lasso model.

	Not Union		Union	
Level	N	Percent	N	Percent
21	9535.00	87.53	750.00	51.16
25	297.00	2.73	116.00	7.91
27	514.00	4.72	208.00	14.19
28	547.00	5.02	392.00	26.74

Table 19: Balance of categorical variable 'class of worker'. model.

	Not Union		Union	
Level	N	Percent	N	Percent
21	7750.00	85.39	694.00	49.29
25	284.00	3.13	115.00	8.17
27	498.00	5.49	208.00	14.77
28	544.00	5.99	391.00	27.77

8. **Estimate your causal estimand of interest using subclassification on the estimated propensity score.**

Before we check the overall ATE or ATT, let's check the number of treated and control units left in each block after trimming. If by any chance, a block is left with either only one control or treated unit, standard error for the causal effect cannot be computed for that block, thus we must drop this block as a fine tune trimming procedure. The blocking procedure this time is targeting not the balance of covariates, but the estimation of ATE or ATT through subclassification, thus we will "re-block" the units with the `trim` option set to false in the `ps_block` function. The number of treated and control units in each block and for each of the three models we are assessing are presented in table 20

Table 20: Number of units in each block, by treatment and model.

	Imbens-Rubin		Lasso		Full	
Block	Treatment	Control	Treatment	Control	Treatment	Control
1	NA	NA	102	2640	26	846
2	38	1144	201	3001	86	1515
3	195	3004	135	1454	113	1488
4	120	1481	187	1430	119	1480
5	174	1427	54	346	80	716
6	245	1362	38	362	94	709
7	113	289	39	363	237	1370
8	139	263	102	300	120	281
9	78	121	110	291	133	268
10	102	99	141	260	175	227
11	99	103	32	68	225	176
12	116	85	41	59	NA	NA
13	NA	NA	91	110	NA	NA
14	NA	NA	193	209	NA	NA

We have no such problem in any of our models, but we have to notice the Imbens-Rubin model block #1 had all its observations trimmed out, and therefore must be eliminated from our subclassification estimation.

Thus, we proceed with the computation of estimated causal effects. First, we will estimate both ATE and ATT **without** any controlling covariates. Results shown in table 21 bellow.

Table 21: Causal effects without contolling covariates.

	ATE		ATT	
Model	Estimate	Std.Error	Estimate	Std.Error
Imbens-Rubin	72.67	23.11	93.18	20.38
Lasso	92.82	22.18	68.20	19.72
Full	88.02	24.73	91.93	20.58

Now we will add the three generally most unbalanced covariates (among those originally chosen in item 2) after the trimming procedure. Those are `age`, `age_2` and `education`.

Table 22: Causal effects contolling by unbalanced covariates.

	ATE		ATT	
Model	Estimate	Std.Error	Estimate	Std.Error
Imbens-Rubin	60.65	20.93	84.42	19.37
Lasso	104.54	20.18	81.54	18.37
Full	81.33	22.66	88.24	19.52

The Lasso specification seems to behave differently from the other two models. It computes a somewhat high value for ATE independently of controlling covariates, while the ATT estimate is surprisingly low when not including controls. Besides being the only specification estimating $ATE > ATT$, the Lasso also had the

largest variations in estimates after controlling, showing signs of non-robustness. Imbens-Rubin and Full model have similar behavior, although the Full model appears to be more robust to the inclusion of covariates.

9. **Estimate your causal estimand of interest using matching on the estimated propensity score.**

Table 23: Matching on propensity score. Causal effects without contolling covariates.

	ATE		ATT		Observations		
Model	Estimate	Std.Error	Estimate	Std.Error	N.obs	N.treated	N.matched
Imbens-Rubin	74.4176	28.23574	100.2612	22.46069	10797	1419	10797
Lasso	90.46248	26.21567	46.16244	21.43359	12359	1466	12359
Full	84.08657	28.84148	98.91688	22.72402	10484	1408	10484

10. **Estimate your causal estimand of interest using inverse probability weighting (Horvitz-Thompson). Briefly compare the results from your different estimators.**

For this item I'd rather use a dobly-robust estimation, including as covariates the ones I have chosen in item 2, and presented in table 1, with age entering the regression as a linear and quadratic term. Regression weights, λ_i will be given according to the Horwitz-Thompson formulation. For ATE we have,

$$\lambda_i^{ate} = \frac{1}{e\left(X_i\right)^{W_i} \cdot \left(1 - e\left(X_i\right)\right)^{1-W_i}} = \begin{cases} 1/\left(1 - e\left(X_i\right)\right) & \text{if } W_i = 0 \\ 1/e\left(X_i\right) & \text{if } W_i = 1 \end{cases}$$

and for ATT the weights are slightly changed,

$$\lambda_i^{att} = \frac{1}{P[D_i = 1]} \frac{e(X_i)^{1-W_i}}{\left(1 - e\left(X_i\right)\right)^{1-W_i}} = \begin{cases} P[D_i = 1]^{-1} \cdot e(X_i)/\left(1 - e\left(X_i\right)\right) & \text{if } W_i = 0 \\ P[D_i = 1]^{-1} & \text{if } W_i = 1 \end{cases}$$

Table 24: Doubly-robust causal effect estimation.

	ATE		ATT	
Model	Estimate	Std.Error	Estimate	Std.Error
Imbens-Rubin	73.96	11.61	88.04	11.36
Lasso	93.83	10.57	83.07	10.68
Full	81.04	11.86	89.78	11.53

In general we have found treatment effects lower than the naive estimate from model (2). Although the Lasso model have presented some high estimates for ATE, this model behaved rather erratically, thus is not very reliable. Both models Imbens-Rubin and Full model had more reliable results, not changing significantly when sensitivity to the inclusion of covariates is analyzed for our three estimation methods, subclassification, propensity score matching and doubly-robust regression. The algorithm of Imbens-Rubin to select covariates for a propensity score model took a really long time to run. Therefore, for this application where we have a relatively small number of covariates, my preference relies on the Full model where we estimate the propensity score through a logit model including all meaningful variables and their second order interactions.

When comparing the methods of estimation, my personal preference is the doubly-robust estimator. Besides the "double" robustness property where the treatment effect will be unbiased if either the linear regression or the propensity score specifications are correct, this estimator gives higher precision (i.e. lower standard errors) by including sensible explanatory covariates for the outcome. Even though, one must be careful when

utilizing this type of estimator, since it is sensitive to propensity scores extreme values, thus, it is essential that trimming is made and the overlap property holds in the sample.

Bonus estimator: Causal random forest

As a bonus exercise, I'll implement the algorithm due to Wager and Athey (2018) (also in (Athey et al. 2019)). This is the Generalized Random Forest, which extends the well regarded Breiman (2001) random forest. The authors provided an R package, `grf`, to implement their algorithm and another paper that exemplifies its usage on observational data, (Athey and Wager 2019).

A short explanation of random forest is due. Random forest make predictions as an average of results in b trees according to the following algorithm: 1) For each $b = 1, \ldots, B$, draw a subsample $\mathcal{S}_b \subseteq \{1, \ldots, n\}$; 2) Grow a tree via recursive partitioning on each such subsample of the data; and 3) Make predictions as:

$$\hat{\mu}(x) = \frac{1}{B} \sum_{b=1}^{B} \sum_{i=1}^{n} \frac{Y_i \mathbf{1}\left(\{X_i \in L_b(x), i \in \mathcal{S}_b\}\right)}{|\{i : X_i \in L_b(x), i \in \mathcal{S}_b\}|}$$

where $L_b(x)$ denotes the leaf node from b tree containing the observation x. Therefore the prediction is the average over all trees in the forest of the prediction for specific trees, which is just the average outcome of all samples that happen to be in the same leaf node as the observation x. A random forest prediction is an average of predictions made by individual trees.

This random forest prediction can be used to model either the outcome, $\hat{m}(X)$ or the propensity score, $\hat{e}(X)$, or both. In the case of **out-of-bag** prediction, we estimate $\hat{\mu}^{(-i)}(X_i))$ by only considering those trees b for which $i \notin \mathcal{S}_b$.

If the conditional average treatment effect function is constant, $\tau(x) = \tau$ for all $x \in \mathcal{X}$, then the following estimator is semiparametrically efficient for τ under unconfoundedness (Chernozhukov et al. 2018):

$$\hat{\tau} = \frac{\frac{1}{n}\sum_{i=1}^{n}\left(Y_i - \hat{m}^{(-i)}\left(X_i\right)\right)\left(W_i - \hat{e}^{(-i)}\left(X_i\right)\right)}{\frac{1}{n}\sum_{i=1}^{n}\left(W_i - \hat{e}(-i)\left(X_i\right)\right)^2}$$

And this is exactly a doubly-robust estimator for the average causal effect. Below in table 25 we present the ATE and ATT estimated by a causal random forest together with previously presented results from the doubly-robust estimator of item 10).

Table 25: Doubly-robust and Causal Forest estimates.

	ATE		ATT	
Model	Estimate	Std.Error	Estimate	Std.Error
Imbens-Rubin	73.96	11.61	88.04	11.36
Lasso	93.83	10.57	83.07	10.68
Full	81.04	11.86	89.78	11.53
Causal Forest	**19.62**	**20.00**	**9.87**	**17.24**

And the results now are quite different! According to this procedure, we do not reject the null hypothesis of zero effect for both ATE and ATT.

Just out of curiosity, we present below an exemplary tree from our causal forest in figure 3 and variable importance to explain earnings variance in figure 4.

Two common diagnostics to evaluate if the identifying assumptions behind `grf` hold is a propensity score histogram and covariance balance plot. The overlap assumption requires a positive probability of treatment for each X_i. We have already done this kind of assessment in figure 2.

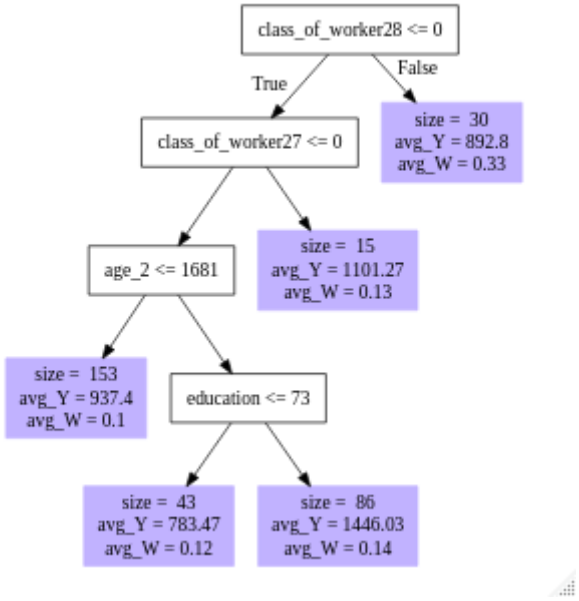

Figure 3: Tree example from the estimated causal forest.

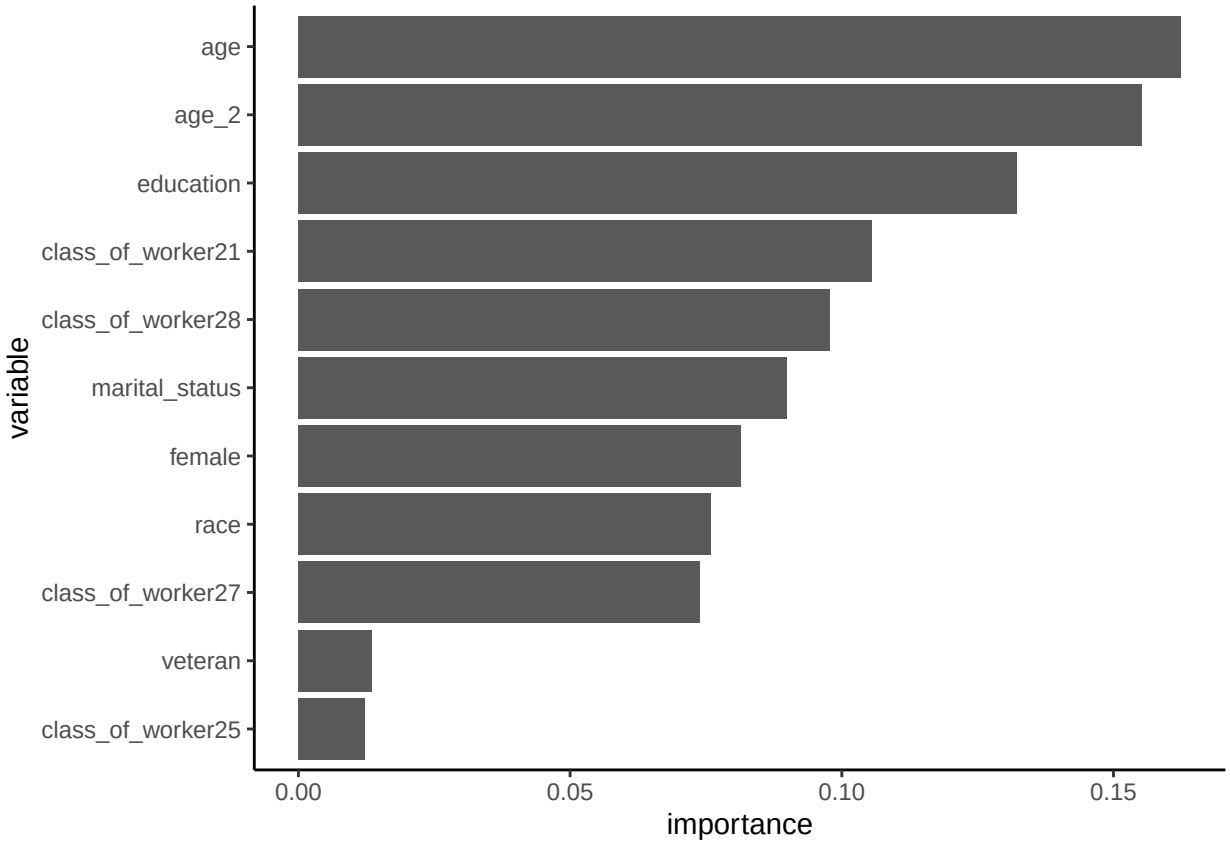

Figure 4: Variable importance in explaning earnings variance.

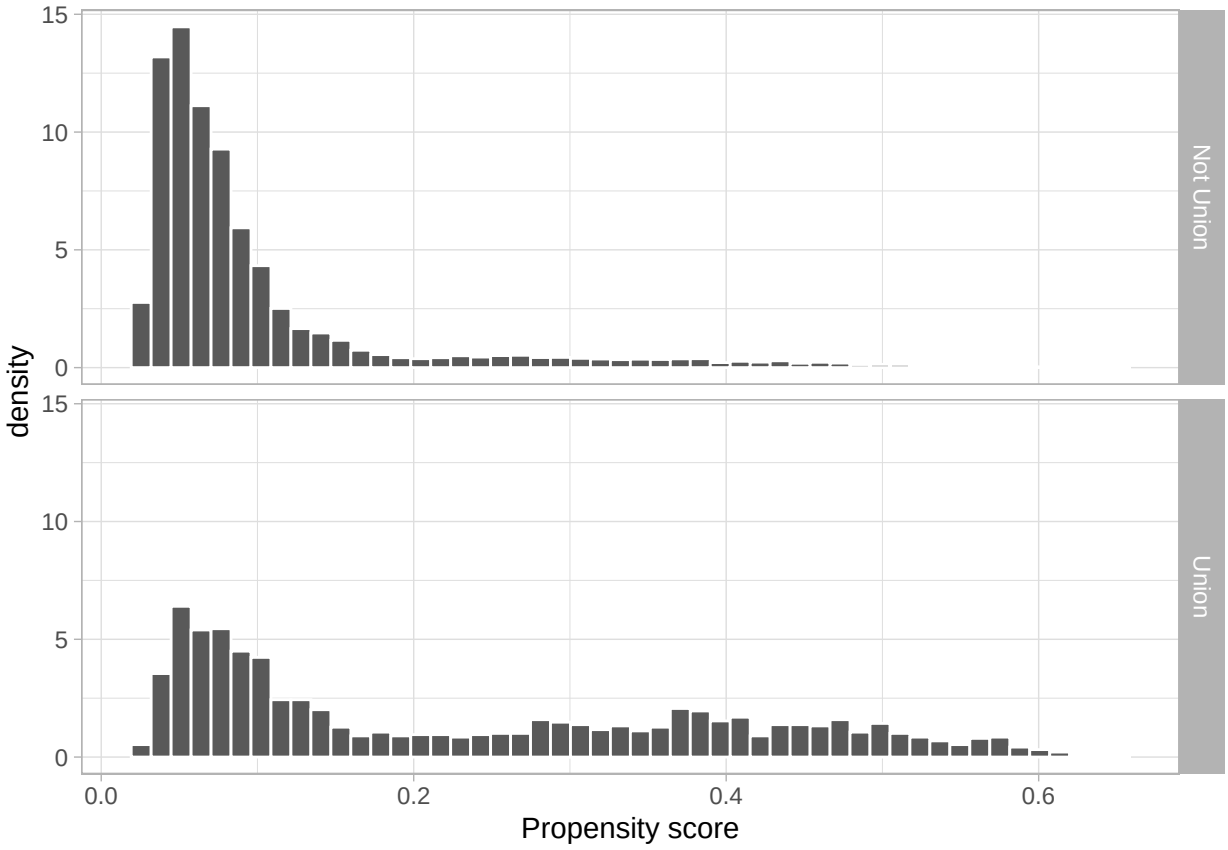

Figure 5: Overlap of propensity scores.

One can also check that the continuous covariates are balanced across the treated and control group by plotting the inverse-propensity weighted histograms of all unities.

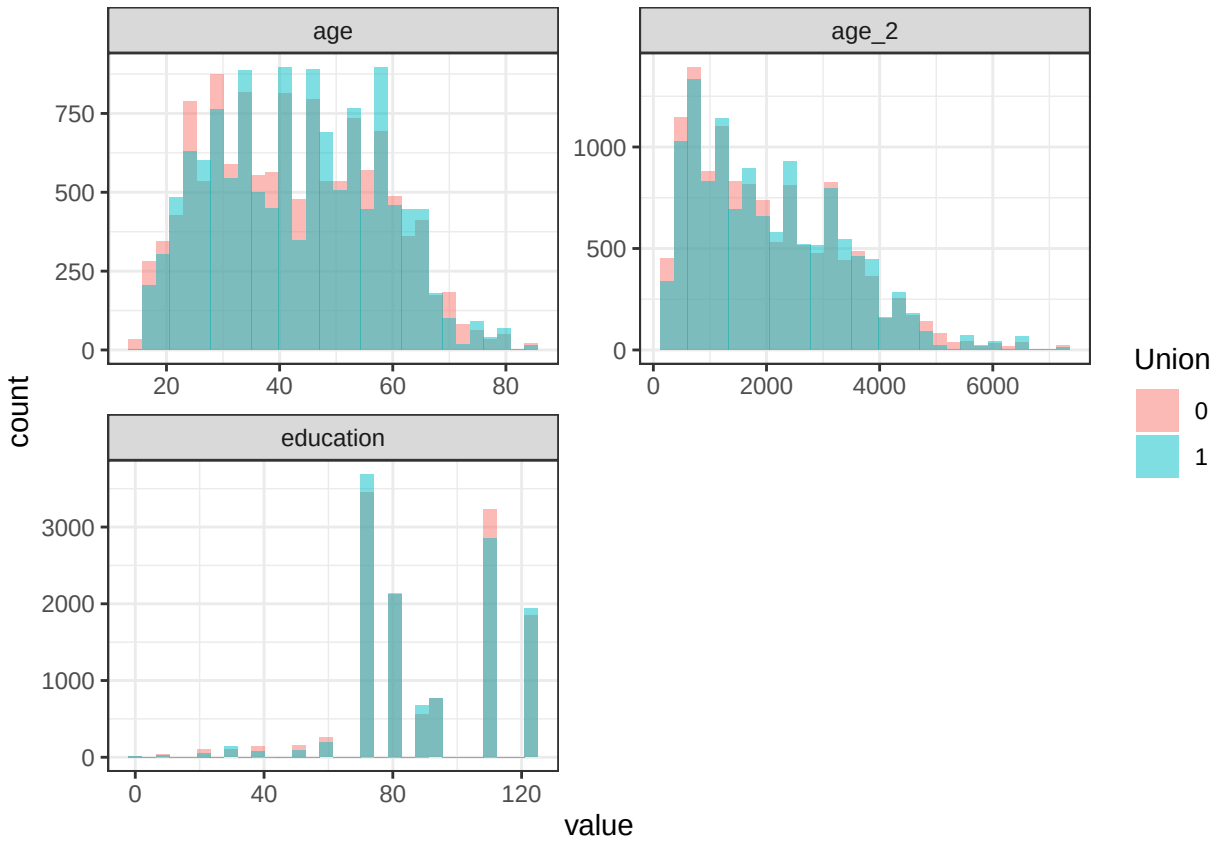

Figure 6: Assessing continuous covariates balance.

While for categorical or binary covariates we analyze their total count in each treatment group, by the levels the variable can assume.

We can see there is good amount of overlap among treatment groups and the balance of covariates seems to be very good.

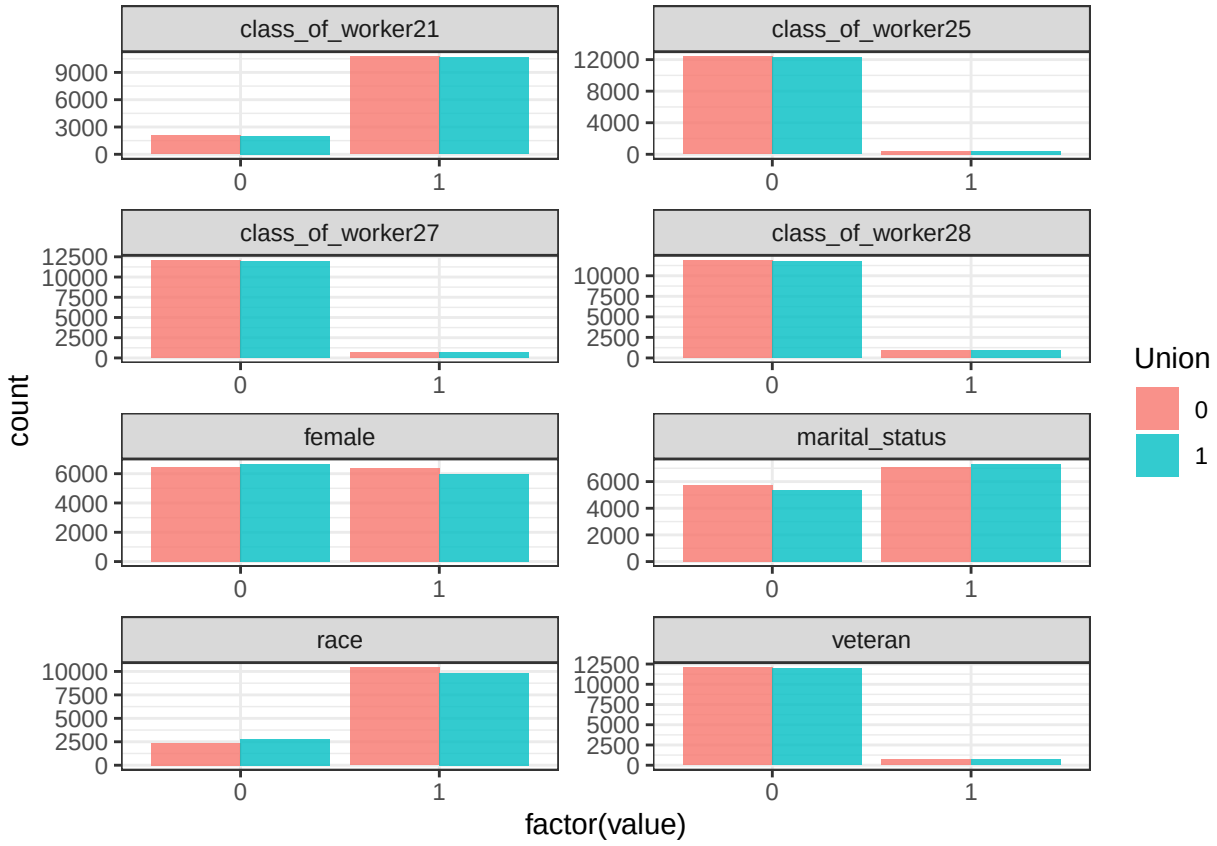

Figure 7: Assessing binary covariates balance.

Annex A - Selected covariates

Imbens-Rubin

```
##  [1] "age"
##  [2] "age_2"
##  [3] "female"
##  [4] "race"
##  [5] "marital_status"
##  [6] "veteran"
##  [7] "education"
##  [8] "class_of_worker"
##  [9] "private_health_insurance"
## [10] "class_of_worker_last_year"
## [11] "medicaid"
## [12] "education:class_of_worker_last_year"
## [13] "class_of_worker:class_of_worker_last_year"
## [14] "female:education"
## [15] "age_2:class_of_worker_last_year"
## [16] "marital_status:class_of_worker_last_year"
## [17] "race:class_of_worker_last_year"
## [18] "female:class_of_worker_last_year"
## [19] "age:female"
## [20] "female:class_of_worker"
## [21] "marital_status:medicaid"
## [22] "private_health_insurance:class_of_worker_last_year"
## [23] "class_of_worker:private_health_insurance"
## [24] "age:class_of_worker_last_year"
## [25] "class_of_worker_last_year:medicaid"
## [26] "education:private_health_insurance"
## [27] "veteran:class_of_worker_last_year"
## [28] "veteran:class_of_worker"
## [29] "age:age_2"
## [30] "marital_status:veteran"
```

Lasso

```
##  [1] "education"
##  [2] "age:education"
##  [3] "age:worked_last_year"
##  [4] "female:class_of_worker_last_year14"
##  [5] "race:class_of_worker27"
##  [6] "race:class_of_worker_last_year14"
##  [7] "race:class_of_worker_last_year27"
##  [8] "marital_status:class_of_worker28"
##  [9] "marital_status:worked_last_year"
## [10] "education:class_of_worker25"
## [11] "education:class_of_worker27"
## [12] "education:class_of_worker_last_year14"
## [13] "class_of_worker25:worked_last_year"
## [14] "class_of_worker27:worked_last_year"
## [15] "class_of_worker28:worked_last_year"
## [16] "class_of_worker25:private_health_insurance"
## [17] "class_of_worker27:class_of_worker_last_year28"
## [18] "private_health_insurance:class_of_worker_last_year27"
```

Full

```
##  [1] "age"
##  [2] "age_2"
##  [3] "female"
##  [4] "race"
##  [5] "marital_status"
##  [6] "veteran"
##  [7] "education"
##  [8] "class_of_worker"
##  [9] "worked_last_year"
## [10] "private_health_insurance"
## [11] "medicaid"
## [12] "class_of_worker_last_year"
## [13] "age:age_2"
## [14] "age:female"
## [15] "age:race"
## [16] "age:marital_status"
## [17] "age:veteran"
## [18] "age:education"
## [19] "age:class_of_worker"
## [20] "age:worked_last_year"
## [21] "age:private_health_insurance"
## [22] "age:medicaid"
## [23] "age:class_of_worker_last_year"
## [24] "age_2:female"
## [25] "age_2:race"
## [26] "age_2:marital_status"
## [27] "age_2:veteran"
## [28] "age_2:education"
## [29] "age_2:class_of_worker"
## [30] "age_2:worked_last_year"
## [31] "age_2:private_health_insurance"
## [32] "age_2:medicaid"
## [33] "age_2:class_of_worker_last_year"
## [34] "female:race"
## [35] "female:marital_status"
## [36] "female:veteran"
## [37] "female:education"
## [38] "female:class_of_worker"
## [39] "female:worked_last_year"
## [40] "female:private_health_insurance"
## [41] "female:medicaid"
## [42] "female:class_of_worker_last_year"
## [43] "race:marital_status"
## [44] "race:veteran"
## [45] "race:education"
## [46] "race:class_of_worker"
## [47] "race:worked_last_year"
## [48] "race:private_health_insurance"
## [49] "race:medicaid"
## [50] "race:class_of_worker_last_year"
## [51] "marital_status:veteran"
## [52] "marital_status:education"
```

```
## [53] "marital_status:class_of_worker"
## [54] "marital_status:worked_last_year"
## [55] "marital_status:private_health_insurance"
## [56] "marital_status:medicaid"
## [57] "marital_status:class_of_worker_last_year"
## [58] "veteran:education"
## [59] "veteran:class_of_worker"
## [60] "veteran:worked_last_year"
## [61] "veteran:private_health_insurance"
## [62] "veteran:medicaid"
## [63] "veteran:class_of_worker_last_year"
## [64] "education:class_of_worker"
## [65] "education:worked_last_year"
## [66] "education:private_health_insurance"
## [67] "education:medicaid"
## [68] "education:class_of_worker_last_year"
## [69] "class_of_worker:worked_last_year"
## [70] "class_of_worker:private_health_insurance"
## [71] "class_of_worker:medicaid"
## [72] "class_of_worker:class_of_worker_last_year"
## [73] "worked_last_year:private_health_insurance"
## [74] "worked_last_year:medicaid"
## [75] "worked_last_year:class_of_worker_last_year"
## [76] "private_health_insurance:medicaid"
## [77] "private_health_insurance:class_of_worker_last_year"
## [78] "medicaid:class_of_worker_last_year"
```

Annex B - R Code

```r
#' Homework II - Microeconometrics I
#' Author: Rafael Felipe Bressan
#'
#' Loading libraries
library(dplyr)
library(tidyr)
library(broom)
library(data.table)
library(sandwich)
library(lmtest)
library(modelsummary)
library(Matching)
library(hdm)
library(grf)

#' Reading in the dataset
data <- fread("I/input/cps_union_data.csv")

#' 1. Comparing average earning for unionized and not unionized workers
ear_na <- sum(is.na(data$earnings))
earnings_union <- data[union == 1 & !is.na(earnings), earnings]
earnings_not <- data[union == 0 & !is.na(earnings), earnings]
#' Difference in averages
avg_dif <- mean(earnings_union) - mean(earnings_not)
#' t-test for difference of means
test <- t.test(earnings_union, earnings_not,
               alternative = "greater", var.equal = FALSE)

#' First prepare a dataset with all variables
#' remove NAs on earnings, set veterans NAs to zero, remove unnecessary columns,
#' change marital status to married (1), not married (0)
#' change race to white (1), not white (0)
#' change to categorical variables
cat_cols_full <- c("class_of_worker", "class_of_worker_last_year")
data_full <- data[!is.na(earnings), ][
  is.na(veteran), veteran := 0][
    , c("V1", "CPSID", "CPSIDP", "public_housing", "employed") := NULL][
      , `:=`(marital_status = ifelse(marital_status %in% c(1, 2), 1, 0),
             race = ifelse(race == 1, 1, 0))][
      , (cat_cols_full) := lapply(.SD, factor), .SDcols = cat_cols_full]
#' Insert a column for age^2
data_full[, age_2 := age^2]
#' Gotta standardize variables total_income_last_year, wage_income_last_year and
#' own_farm_income_last_year
standardize <- function(x) {
  (x - mean(x)) / sd(x)
}
sd_cols <- c("total_income_last_year", "wage_income_last_year",
             "own_farm_income_last_year")
data_full[, (sd_cols) := lapply(.SD, standardize), .SDcols = sd_cols]

#' Number of observations inside each level of a factor variable
```

```
nobs_level <- data.frame(unclass(summary(data_full[, ..cat_cols_full], maxsum = 10)),
                         check.names = FALSE, stringsAsFactors = FALSE)
nobs_level[is.na(nobs_level)] <- ""
#' class_of_worker_last_year 29 has only one observation. Set it to level 0 and
#' drop level 29 from this factor
data_full[class_of_worker_last_year == 29, class_of_worker_last_year := "0"]
data_full[, class_of_worker_last_year := droplevels(class_of_worker_last_year)]

#' 2. Regression with covariates
#' a. Getting the covariates description from dictionary
dic <- fread("I/input/dictionary.csv", header = TRUE)
#' descriptions
desc <- dic[variable != "", .(variable, description)]
#' Select covariates
covar <- c("age", "age_2", "female", "race", "marital_status",
           "veteran", "education", "class_of_worker")
covariates <- desc[variable %in% covar]

#' b. Estimate the model
#' Cleaning and preparing data for regression
columns <- c("earnings", "union", covar)
data_cov <- data_full[, ..columns]
#' Missings in other variables
missings <- colSums(apply(data, 2, is.na))
#' 83 missings in veteran. Let's input values to them

# Probit model to veteran ------------------------------------------------
vet_ratio <- table(data$veteran)
#' #' Impute values to veteran
#' #' Do not separate data by veteran is NA or not right now
#' vet <- data[, -c("V1", "CPSID", "CPSIDP")]
#' #' Any other variable with NA?
#' colSums(apply(vet, 2, is.na))
#' #' 8636 NAs in public_housing, drop this column.
#' #' 19 NAs in earnings, drop only the rows
#' vet <- vet[, public_housing := NULL][!is.na(earnings)]
#' #' Make variables categorical. Numerical variables are:
#' num_cols <- c("veteran", "age", "total_income_last_year", "wage_income_last_year",
#'               "own_farm_income_last_year", "earnings")
#' cat_cols <- names(vet)[!(names(vet) %in% num_cols)]
#' vet[, (cat_cols) := lapply(.SD, as.factor), .SDcols = cat_cols]
#' #' All factors have two or more levels?
#' lapply(vet[, ..cat_cols], function(x) length(levels(x)))
#' #' employed has only one level. Conveys no information!
#' vet[, employed := NULL]
#' cat_cols <- cat_cols[!(cat_cols %in% "employed")]
#' #' All levels, for each variable, have more than one observation?
#' summary(vet[, ..cat_cols], maxsum = 16) # education has 16 levels
#' #' class_of_worker_last_year's level 29 has only one observation
#' #' Let's aggregate it with level 0. Level 29 is unpaid family worker and
#' #' best corresponds to level 0, did not work
#' vet[class_of_worker_last_year == 29, class_of_worker_last_year := "0"]
#' #' Now separate data by veteran NA or not
```

```r
#' vet_na <- vet[is.na(veteran)]
#' vet_not <- vet[!is.na(veteran)]
#' #' Run a probit regression of veteran on all other
#' vet_prob <- glm(veteran~., family = binomial(link = "probit"), data = as.data.frame(vet_not))
#' summary(vet_prob)
#' #' Predict values of veteran status to vet_na
#' vet_pred <- ifelse(
#'   predict(vet_prob, newdata = vet_na, type = "response") > vet_ratio[2]/vet_ratio[1],
#'   1, 0
#' )
#' All predictions are equal to zero, better set veteran status to zero directly
# data_cov[is.na(veteran), veteran := 0]

# 2b Estimating the model ------------------------------------------------
#' Categorical variables: race, marital_status, class_of_worker
# num_cols <- c("earnings", "age", "age_2", "union", "female", "veteran", "education")
cat_cols <- cat_cols_full[cat_cols_full %in% names(data_cov)]
# data_cov[, (cat_cols) := lapply(.SD, as.factor), .SDcols = cat_cols]

#' Estimate the model
model0 <- lm(earnings~union, data = data_cov)
model1 <- lm(earnings~., data = data_cov)
model1_coef_tbl <- coeftest(model1, vcov. = vcovHC, type = "HC0")
model0_coef_tbl <- coeftest(model0, vcov. = vcovHC, type = "HC0")
robust1.se <- model1_coef_tbl[, "Std. Error"]
robust1.pval <- model1_coef_tbl[, "Pr(>|t|)"]
robust0.se <- model0_coef_tbl[, "Std. Error"]
robust0.pval <- model0_coef_tbl[, "Pr(>|t|)"]

# 2c Weights -------------------------------------------------------------
#' Regression of union on all covariates
union_cov <- lm(union~.-earnings, data_cov)
resid_union_cov <- residuals(union_cov)
ssr_union_cov <- deviance(union_cov)
#' Compute the weight vector
union_vec <- data_cov[, union]
weight <- (resid_union_cov*(2*union_vec - 1))/ssr_union_cov
data_cov[, weight := weight] # column bind weight to data_cov
#' Summary statistics
w_summary <- datasummary(Mean+Median+SD+Min+Max+sum ~ weight*factor(union),
                         data = data_cov,
                         fmt = "%.6f",
                         output = 'data.frame')
#' Do the weights sum one in control?
data_cov[, sum(.SD), by = union, .SDcols = "weight"]
#' negative values?
data_cov[, min(.SD), by = union, .SDcols = "weight"]

# 4 Balance checks -------------------------------------------------------
#' Assessing balance of covariates using the three methods by Imbens-Rubin
#' balance table: t-test and normalized difference -------------------------
table.test <- function(data, covariates, treat_var, col_ret = "all")
{
```

```
  #' data.table is allowed in data, but has different syntax
  if (is.data.table(data))  treat_vec = data[, get(treat_var)]
  else treat_vec = data[, treat_var]

  table = c()

  for (lab in covariates)
  {
    if (is.data.table(data))  cov_vec = data[, get(lab)]
    else cov_vec = data[, lab]

    control = cov_vec[treat_vec == 0 & !is.na(treat_vec)]
    treatment = cov_vec[treat_vec == 1 & !is.na(treat_vec)]

    normalized_diff = (mean(treatment) - mean(control))/sqrt((var(treatment) + var(control))/2)

    table.line = cbind( "meanc" = mean(control),
                        "meant" = mean(treatment),
                        "tstat" = tryCatch({t.test(control, x = treatment)$statistic},
                                           error = function(e){NaN}),
                        "norm.diff" = normalized_diff)

    table = rbind(table, table.line)
  }

  table <- as.data.frame(table)
  rownames(table) <- NULL

  if (col_ret[1] == "all")
    return(cbind(covar = covariates, table))
  else
    return(cbind(covar = covariates, table[, col_ret, drop = FALSE]))
}
#' First the categorical variables
cat_form <- paste0(
  paste0(cat_cols, collapse = "+"), "~factor(union)*((N=1)+Percent('col'))"
)
cat_balance <- datasummary(as.formula(cat_form),
                           data = data_cov[, c("union", ..cat_cols)],
                           output = 'data.frame')
#' Then numerical variables
num_balance <- table.test(data_cov, covar[!covar %in% "class_of_worker"], "union")

# 5 Select covariates ----------------------------------------------------

#' Stepwise model selection - Imbens and Rubin -----------------------------
#' Imbens and Rubin's stepwise selection algorithm
#' treatment: character variable for treatment indicator variable
#' Xb: character vector with names of basic covariates: you may pass it as  c()
#' if you do not want any basic covariate
#' Xt: character vector with names for covariates to be tested for inclusion
#' data: dataframe with variables
#' Clinear: threshold, in terms of likelihood ratio statistics, for inclusion of
```

```r
#' linear terms
#' Cquadratic: threshold, in terms of likelihood ratio statistics, for inclusion
#' of quadratic/interaction terms
#' Intercept: does model include intercept?
#' Author: Luis Alvarez
#' Modifications: Rafael F. Bressan
ir_stepwise <- function(treatment, Xb, Xt, data, Clinear = 1, Cquadratic = 2.71,
                        intercept = TRUE)
{
  #Add or not intercept
  if (intercept)
    inter.add = "1"
  else inter.add = "-1"


    #Formula for model
    if (length(Xb) == 0)
      formula = paste(treatment, inter.add, sep = " ~ ")
    else formula = paste(treatment, paste(c(inter.add,Xb), collapse = " + "),
                         sep = " ~ ")

      continue = TRUE

      Xt_left = Xt
      # First order inclusion
      while (continue) {
        null.model = glm(as.formula(formula), data, family = "binomial")

        null.lkl = logLik(null.model)

        test.stats = c()
        for (covariate in Xt_left)
        {
          formula.test = paste(formula, covariate, sep = " + ")
          test.model = glm(as.formula(formula.test), data, family = "binomial")

          lkl.ratio = 2*(as.numeric(logLik(test.model)) - as.numeric(null.lkl))
          test.stats = c(test.stats, lkl.ratio)
        }

        if (max(test.stats,na.rm = TRUE) < Clinear)
          continue = FALSE else {

            add.coef = Xt_left[which.max(test.stats)]

            formula = paste(formula, add.coef, sep = " + ")

            Xt_left = Xt_left[-which.max(test.stats)]
          }

      }

      #Defining Xstar set. Set of first order included variables
```

```
      Xstar = c(Xb, Xt[!(Xt %in% Xt_left)])

      #Creating all combinations of Xstar interactions
      combinations = expand.grid(Xstar, Xstar)
      Xcomb = paste(combinations[,1],combinations[,2],sep = ":")

      continue = TRUE

      Xcomb_left = Xcomb

      while (continue) {
        null.model = glm(as.formula(formula), data, family = "binomial")

        null.lkl = logLik(null.model)

        test.stats = c()
        for (covariate in Xcomb_left)
        {
          formula.test = paste(formula, covariate, sep = " + ")
          test.model = glm(as.formula(formula.test), data, family = "binomial")

          lkl.ratio = 2*(as.numeric(logLik(test.model)) - as.numeric(null.lkl))
          test.stats = c(test.stats, lkl.ratio)
        }

        if (max(test.stats,na.rm = TRUE) < Cquadratic)
          continue = FALSE else {

            add.coef = Xcomb_left[which.max(test.stats)]

            formula = paste(formula, add.coef, sep = " + ")

            Xcomb_left = Xcomb_left[-which.max(test.stats)]
          }

      }

      return(list(formula = formula,
                  inc_x = Xstar))
}

#' own_farm_income_last_year is mostly zero and has an outlier. Better not use it
#' total_income_last_year and wage_income_last_year are bad controls! Do not
#' include them.
xt <- names(data_full)[!names(data_full) %in%
                         c(covar, "earnings", "union", "own_farm_income_last_year",
                           "total_income_last_year", "wage_income_last_year")]
#' TEST ONLY: Select a random sample of full data for quick results
#' set.seed(1234)
#' s <- sample(nrow(data_full), 2000)
ir_form <- ir_stepwise("union", covar, xt, data = data_full)
ps_ir <- glm(as.formula(ir_form$formula), family = "binomial",
             data = data_full[, -c("earnings")])
```

```r
terms_ir <- attr(terms(ps_ir), "term.labels")

#' Model selection via Lasso
x_lasso <- c(covar, xt)
ps_lasso <- rlassologit(union~(.)^2,
                        data = data_full[, c("union", ..x_lasso)])
summary(ps_lasso, all = FALSE)
terms_lasso <- names(coef(ps_lasso))[ps_lasso$index]

#' Model with full set of covariates!
ps_all <- glm(union~(.)^2, family = "binomial",
              data = data_full[, c("union", ..covar, ..xt)])
terms_all <- attr(terms(ps_all), "term.labels")

#' data table with union and different latent indices based on estimation method
dt_ps <- data.table(model = c("Imbens-Rubin", "Lasso", "Full"),
                    union = list(data_full$union),
                    ps = list(predict(ps_ir, type = 'response'),
                              as.vector(predict(ps_lasso, type = 'response')),
                              predict(ps_all, type = 'response')),
                    li = list(predict(ps_ir),
                              as.vector(predict(ps_lasso, type = 'link')),
                              predict(ps_all)))

li_bal <- dt_ps[
  , .(union = unlist(union),
      li = unlist(li),
      ps = unlist(ps))
  , by = model][
    , .(mean = sapply(.SD, mean), var = sapply(.SD, var))
    , by = .(union, model), .SDcols = "li"] %>%
  dcast(model~union, value.var = c("mean", "var"))
li_bal[, norm_diff := .(abs(mean_1 - mean_0)/sqrt((var_1 + var_0)/2))]
setcolorder(li_bal, c("model", "mean_0", "var_0", "mean_1", "var_1", "norm_diff"))

# 6 Quality of PS ------------------------------------------------------------
#' Function that subdivides a given propensity score vector in subblocks
#' treat = vector with treatment assignments
#' lin.psm = vector with linearized PSs
#' K = how many covariates will we want to test/use in bias correction of
#' estimates later on?
#' t.max = threshold for tstat in making a further subdivide
#' trim = should we discard extreme observations so there is overlap?
#' Author: Luis Alvarez
ps_blocks <- function(treat, lin.psm, K, t.max = 1.96,  trim = TRUE)
{
  if(trim){
    b0 = min(plogis(lin.psm[treat==1]))
    b1 = max(plogis(lin.psm[treat==0]))
  } else
  {
    b0 = 0
    b1 = 1
```

```
  }
  b_vec = c(b0,b1)
  while (TRUE)
  {
    J = length(b_vec)-1
    b_vec_new = do.call(c,lapply(1:J, function(j){
      sample = (b_vec[j] <= plogis(lin.psm)) & (plogis(lin.psm) < b_vec[j+1])

      ps.treat = lin.psm[sample&treat==1]
      ps.control = lin.psm[sample&treat==0]

      #print(length(ps.control))
      #print(length(ps.treat))

      t.test.pass = tryCatch({abs(t.test(ps.control, ps.treat)$statistic) > t.max}, error = function(e)

      med.val = median(c(ps.treat, ps.control))

      Nt.below = sum(ps.treat < med.val)
      Nt.above = sum(ps.treat >= med.val)
      Nc.below = sum(ps.control < med.val)
      Nc.above = sum(ps.control >= med.val)

      sample.crit = min(Nt.below, Nt.above, Nc.below, Nc.above) >= max(3, K+2)

      if(t.test.pass&sample.crit)
        return(c(b_vec[j], plogis(med.val), b_vec[j+1])) else return(c(b_vec[j], b_vec[j+1]))

    }))
    b_vec_new = unique(b_vec_new)

    #print(length(b_vec_new))
    if(length(b_vec_new)==length(b_vec))
      break else b_vec = b_vec_new
  }

  #Constructing blocking variable now
  block_var = rep(NA, length(treat))

  for (j in 1:(length(b_vec) - 1))
    block_var[(b_vec[j] <= plogis(lin.psm)) & (plogis(lin.psm) < b_vec[j+1])] = j
  #' Propensity scores lower than b0 will be in block -2 and PS higher than or
  #' equal to b1 will be block -1
  # block_var[plogis(lin.psm) < b0] <- -2
  # block_var[plogis(lin.psm) >= b1] <- -1

  return(block_var)
}

#' Appending latent indices to data_full, then computing blocks
k_ir <- length(terms_ir)
k_lasso <- length(terms_lasso)
k_all <- length(terms_all)
```

```
cols_full <- names(data_full)[
  ! names(data_full) %in% c("earnings", "own_farm_income_last_year", "li_ir",
                            "li_lasso", "li_all", "block_ir", "block_lasso",
                            "block_all")]
num_cols_full <- cols_full[! cols_full %in% cat_cols_full]
#' Minimum number of elements in a block is the number of all covariates to use
K <- length(c(covar, xt))
data_full[, `:=`(li_ir = predict(ps_ir, type = 'link'),
                 li_lasso = as.vector(predict(ps_lasso, type = 'link')),
                 li_all = predict(ps_all, type = 'link'))][
                   , `:=`(block_ir = ps_blocks(union, li_ir, K),
                          block_lasso = ps_blocks(union, li_lasso, K),
                          block_all = ps_blocks(union, li_all, K))]

#' Third approach, single covariate, single stratum at a time
#' First we need to expand factor variables into dummies
tbl_covar <- num_cols_full[num_cols_full != "union"]

summary_block_ir <-
  model.matrix(~.-1, data = data_full[, c(..cols_full, "block_ir")]) %>%
  na.omit() %>%
  as_tibble() %>%
#' Using table.test by blocks and only on numerical variables
  group_by(block_ir) %>%
  summarise(t_stat = list(table.test(cur_data(), names(cur_data()),
                                     "union", "tstat"))) %>%
  unnest(t_stat) %>%
  filter(covar != "union") %>%
  pivot_wider(covar, names_from = block_ir, values_from = tstat)

summary_block_lasso <-
  model.matrix(~.-1, data = data_full[, c(..cols_full, "block_lasso")]) %>%
  na.omit() %>%
  as_tibble() %>%
  group_by(block_lasso) %>%
  summarise(t_stat = list(table.test(cur_data(), names(cur_data()),
                                     "union", "tstat"))) %>%
  unnest(t_stat) %>%
  filter(covar != "union") %>%
  pivot_wider(covar, names_from = block_lasso, values_from = tstat)

summary_block_all <-
  model.matrix(~.-1, data = data_full[, c(..cols_full, "block_all")]) %>%
  na.omit() %>%
  as_tibble() %>%
  group_by(block_all) %>%
  summarise(t_stat = list(table.test(cur_data(), names(cur_data()),
                                     "union", "tstat"))) %>%
  unnest(t_stat) %>%
  filter(covar != "union") %>%
  pivot_wider(covar, names_from = block_all, values_from = tstat)

#' Many of class_of_worker has problems in the t-stat. Probably due to lack of
```

```r
#' observations
obs_block_ir <- model.matrix(~.-1,
                             data = data_full[, c(..cols_full, "block_ir")]) %>%
  na.omit() %>%
  as_tibble() %>%
  dplyr::select(matches("class_of_worker"), "block_ir") %>%
  group_by(block_ir) %>%
  summarise(across(everything(), sum))
#' Indeed many strata of class_of_worker type of covariate do not have any
#' observation for a given block, thus the statistics can't be computed.
#' But this is mainly due to the attribution for the NA block, which we are
#' removing from balance assessment.
#'
#' Re-blocking!! To assess number of treated and control by blocks with trim
#' option set to false. This is the blocking config we will pass to the
#' forthcomming subclassification procedure.
data_full[, `:=`(block_ir = ps_blocks(union, li_ir, k_ir, trim = FALSE),
                 block_lasso = ps_blocks(union, li_lasso, k_lasso, trim = FALSE),
                 block_all = ps_blocks(union, li_all, k_all, trim = FALSE))]

# 7 Trim the sample ----------------------------------------------------
#' Author: Luis Alvarez
#' Adapted by: Rafael Bressan
trimming.imbens2 <- function(lin.ps)
{
  inv.vec = 1/(plogis(lin.ps)*(1 - plogis(lin.ps)))

  if (max(inv.vec) <= 2*mean(inv.vec))
  {
    print("No trimming")
    return(rep(TRUE, length(lin.ps)))
  }else {
    # value function
    value_fun <- function(gamma) {
      2*sum(inv.vec[inv.vec <= gamma])  - gamma*sum(inv.vec <= gamma)
    }
    # root finding. g is a list with root
    g <- uniroot(value_fun, c(min(inv.vec), max(inv.vec)))

    alpha.trim <- 1/2 - sqrt(1/4 - 1/g$root)
    print(paste("Trimming threshold alpha is ",alpha.trim))
    return(plogis(lin.ps) <= 1 - alpha.trim &  plogis(lin.ps) >= alpha.trim)
  }
}

trim_idx_ir <- trimming.imbens2(data_full$li_ir)
trim_idx_lasso <- trimming.imbens2(data_full$li_lasso)
trim_idx_all <- trimming.imbens2(data_full$li_all)

#' Add trimming indexes to dt_ps
dt_ps[, trim_idx := list(trim_idx_ir, trim_idx_lasso, trim_idx_all)]

#' Balance for numerical variables after trimming
```

```r
trim_bal_ir <- table.test(as.data.frame(data_full[trim_idx_ir]),
                          num_cols_full, "union") %>%
  filter(covar != "union")
trim_bal_lasso <- table.test(as.data.frame(data_full[trim_idx_lasso]),
                          num_cols_full, "union") %>%
  filter(covar != "union")
trim_bal_all <- table.test(as.data.frame(data_full[trim_idx_all]),
                          num_cols_full, "union") %>%
  filter(covar != "union")
#' Balance for categorical variables after trimming
cat_bal_ir <- datasummary(as.formula(cat_form),
                           data = data_full[trim_idx_ir, c("union", ..cat_cols)],
                           output = 'data.frame')
cat_bal_lasso <- datasummary(as.formula(cat_form),
                          data = data_full[trim_idx_lasso, c("union", ..cat_cols)],
                          output = 'data.frame')
cat_bal_all <- datasummary(as.formula(cat_form),
                          data = data_full[trim_idx_all, c("union", ..cat_cols)],
                          output = 'data.frame')

#' Table for balance of latent indices before and after trimming
no_trim_li <- dt_ps %>%
  as_tibble() %>%
  dplyr::select(-c(ps, trim_idx)) %>%
  unnest(c(union, li)) %>%
  group_by(model, union) %>%
  summarise(avg = mean(li), vari = var(li)) %>%
  mutate(diff = avg - lag(avg),
         sqrt = sqrt((vari + lag(vari))/2),
         norm_diff = diff / sqrt) %>%
  na.omit()
trim_li <- dt_ps %>%
  as_tibble() %>%
  dplyr::select(-ps) %>%
  unnest(c(union, li)) %>%
  filter(c(trim_idx_ir, trim_idx_lasso, trim_idx_all)) %>%
  group_by(model, union) %>%
  summarise(avg = mean(li), vari = var(li)) %>%
  mutate(diff = avg - lag(avg),
         sqrt = sqrt((vari + lag(vari))/2),
         norm_diff = diff / sqrt) %>%
  na.omit()
trim_bal_li <- no_trim_li %>%
  left_join(trim_li, by = "model", suffix = c("_no_trim", "_trim")) %>%
  dplyr::select(matches("norm_diff")) %>%
  arrange(model)

# 8 Estimating effects by subclassification ----------------------------------
#'
#' @param dt data.table
#' @param outcome character string, name of outcome variable
#' @param treat character string, name of treatment variable
#' @param block character string, name of block variable
```

```r
#' @param controls Optional. Vector of character strings, name of control
#' variables
#' @return list containing a data.table called blocks with all weights, standard
#' deviations and effects by blocks, and a data.table called effects with ATE
#' and ATT (and their robust standard errors).
#' @author Rafael Bressan
subclassification <- function(dt, outcome, treat, block, controls = NULL) {
  stopifnot(is.data.table(dt))

  K <- length(controls)
  #' regression formula
  if (K == 0)
    form <- paste(outcome, treat, sep = " ~ ")
  else
    form <- paste(outcome, paste(c(treat, controls), collapse = "+"),
                  sep = " ~ ")
  #' Number of observations
  nobs <- nrow(dt)
  nb <- dt[, .(nb = .N), by = c(block)]
  nt <- dt[get(treat) == 1, .(nt = .N), by = c(block)]
  nc <- dt[get(treat) == 0, .(nc = .N), by = c(block)]
  dt_n <- nb[nt, on = c(block)][nc, on = c(block)][
    , `:=`(w_ate = nb/nobs,
           w_att = nt/sum(nt))]
  #' DT to hold results by block
  reg_block <- dt[, .(reg = list(lm(as.formula(form), data = .SD)))
                  , by = c(block)][
                    , `:=`(tau_b = coef(reg[[1]])[treat],
                           rob_se = sqrt(sandwich::vcovHC(reg[[1]])[treat, treat]))
                    , by = c(block)]

  dt_out <- dt_n[reg_block[, c(..block, "tau_b", "rob_se")], on = c(block)]
  setnames(dt_out, block, "block")
  ate <- sum(dt_out$w_ate * dt_out$tau_b)
  ate_se <- sqrt(sum((dt_out$w_ate * dt_out$rob_se)^2))
  att <- sum(dt_out$w_att * dt_out$tau_b)
  att_se <- sqrt(sum((dt_out$w_att * dt_out$rob_se)^2))
  dt_eff <- data.table(stat = c("mean", "se"),
                       ate = c(ate, ate_se),
                       att = c(att, att_se))

  return(list(blocks = dt_out, effects = dt_eff))
}

#' Subclassification without any controlling covariate
sub_nc_ir <- subclassification(data_full[trim_idx_ir],
                               "earnings", "union", "block_ir")
sub_nc_lasso <- subclassification(data_full[trim_idx_lasso],
                                  "earnings", "union", "block_lasso")
sub_nc_all <- subclassification(data_full[trim_idx_all],
                                "earnings", "union", "block_all")
#' Subclassification with unbalanced controls: age, education and
#' private_health_insurance
```

```r
sub_ir <- subclassification(
  data_full[trim_idx_ir],
  "earnings", "union", "block_ir",
  controls = c("age", "education", "age_2"))
sub_lasso <- subclassification(
  data_full[trim_idx_lasso],
  "earnings", "union", "block_lasso",
  controls = c("age", "education", "age_2"))
sub_all <- subclassification(
  data_full[trim_idx_all],
  "earnings", "union", "block_all",
  controls = c("age", "education", "age_2"))
#' Table with number of treated and control by block and model
ntc_ir <- sub_nc_ir$blocks[, c("block", "nt", "nc")]
ntc_lasso <- sub_nc_lasso$blocks[, c("block", "nt", "nc")]
ntc_all <- sub_nc_all$blocks[, c("block", "nt", "nc")]
#' Merge tables
ntc_block <-
  merge(ntc_ir, ntc_lasso, by = "block", all = TRUE) %>%
  merge(ntc_all, by = "block", all = TRUE)
#' ATT and ATE for three models without controlling covariates
sub_effect_nc <- rbind(sub_nc_ir$effects,
                       sub_nc_lasso$effects,
                       sub_nc_all$effects) %>%
  mutate(model = rep(c("Imbens-Rubin", "Lasso", "Full"), each = 2)) %>%
  pivot_wider(id_cols = model, names_from = stat, values_from = c(ate:att))
#' ATT and ATE for three models with controlling covariates
sub_effect <- rbind(sub_ir$effects,
                    sub_lasso$effects,
                    sub_all$effects) %>%
  mutate(model = rep(c("Imbens-Rubin", "Lasso", "Full"), each = 2)) %>%
  pivot_wider(id_cols = model, names_from = stat, values_from = c(ate:att))

# 9 Estimate effects with Matching ----------------------------------------
#' Compute estimates by matching PS
#' Add column "effect"
dt_ps[, effect := list(c("ATE", "ATT"))]

match_nc_effect <- dt_ps %>%
  as_tibble() %>%
  unnest(effect) %>%
  rowwise() %>%
  mutate(match_nc = list(Match(Y = data_full[trim_idx, earnings],
                               Tr = data_full[trim_idx, union],
                               X = li[trim_idx],
                               estimand = effect,
                               M = 1, replace = TRUE)),
         coef_nc = list(tibble(estimate = match_nc["est"],
                               se = match_nc["se"],
                               orig.nobs = match_nc["orig.nobs"],
                               orig.treated.nobs = match_nc["orig.treated.nobs"],
                               match.obs = match_nc["orig.wnobs"]))) %>%
  dplyr::select(model, effect, coef_nc)
```

```r
match_nc <- match_nc_effect %>%
  unnest(coef_nc) %>%
  pivot_wider(id_cols = model,
              names_from = effect,
              values_from = c(estimate:match.obs)) %>%
  dplyr::select(model, matches("_ATE$"), matches("_ATT$"))

# 10 IPW doubly-robust estimation ----------------------------------------
#' Overall probability of treatment. For ATT estimation
p_treat <- mean(data_full$union)

ipw_weights <- dt_ps %>%
  as_tibble() %>%
  dplyr::select(-li) %>%
  unnest(effect) %>%
  rowwise() %>%
  mutate(
    weight = list(case_when(
      effect == "ATE" ~ ifelse(union == 1, 1/ps, 1/(1 - ps)),
      effect == "ATT" ~ (1/p_treat)*ifelse(union == 1, 1, ps/(1 - ps)),
      TRUE ~ 1.0 # no weighting
    )))

#' regressions
#' Doubly-robust regression with the same covariates as item 2
dbl_form <- paste("earnings~union", paste(covar, collapse = "+"), sep = "+")
ipw_reg <- ipw_weights %>%
  mutate(lm_fit = list(lm(as.formula(dbl_form),
                          data = data_full[trim_idx],
                          weights = weight[trim_idx])),
         coef_rob = list(coeftest(lm_fit)),
         est = list(tidy(coef_rob))) %>%
  dplyr::select(model, effect, est) %>%
  unnest(est) %>%
  filter(term == "union") %>%
  pivot_wider(id_cols = model,
              names_from = effect,
              values_from = c(estimate:p.value)) %>%
  dplyr::select(model, estimate_ATE, std.error_ATE, estimate_ATT, std.error_ATT)

#' BONUS: Generalized Random Forest
#'
#' 3 steps: i) Estimate a model for earnings; ii) Estimate a model for the
#' propensity score and; iii) Estimate the causal forest
#'
#' i) Model for earnings utilizes union and covariates of item 2
#'
Y <- data_full$earnings
X <- model.matrix(~.-1+class_of_worker, data_full[, c("union", ..covar)])
Y_forest <- regression_forest(X, Y)
Y_hat <- predict(Y_forest)$predictions
#' ii) Propensity score: Could use the propensity score models from before. But
#' let's try a random forest too.
```

```
ps_cols <- cols_full[!cols_full == "union"]
W <- data_full$union
X_ps <- model.matrix(~.-1+class_of_worker+class_of_worker_last_year,
                  data_full[, ..ps_cols])
W_forest <- regression_forest(X_ps, W)
W_hat <- predict(W_forest)$predictions
#' iii) Causal forest
cf <- causal_forest(X, Y, W, Y_hat, W_hat, tune.parameters = "all")
#' Estimated ATE for every observation
tau_hat <- predict(cf)$predictions
#' ATE and ATT
cf_ate <- average_treatment_effect(cf)
cf_att <- average_treatment_effect(cf, "treated")
#' Table with doubly-robust and causal forest effects
cf_dr_effects <- ipw_reg %>%
  add_row(model = "Causal Forest",
          estimate_ATE = cf_ate["estimate"],
          std.error_ATE = cf_ate["std.err"],
          estimate_ATT = cf_att["estimate"],
          std.error_ATT = cf_att["std.err"])
#' Get a sample tree
tree <- get_tree(cf, 3)
tree_plot <- plot(tree)
#' Variable importance for causal estimation
var_imp <- tibble(variable = colnames(X),
                  importance = variable_importance(cf)) %>%
  arrange(desc(importance)) %>%
  filter(variable != "union") %>%
  mutate(variable = factor(variable, levels = variable))
#' Check whether causal forest predictions are well calibrated.
test_calibration(cf)
#' Histograms of covariates by union status
X_ipw <- data.frame(X) %>%
  mutate(ipw = ifelse(union == 1, 1/W_hat, 1/(1 - W_hat))) %>%
  pivot_longer(cols = -c(union, ipw),
               names_to = "covariate",
               values_to = "value")

save.image("I/input/homework_II.RData")
```

References

Angrist, Joshua D, and Jörn-Steffen Pischke. 2008. *Mostly Harmless Econometrics: An Empiricist's Companion*. Princeton university press.

Ashenfelter, Orley, and David Card. 2010. *Handbook of Labor Economics*. Elsevier.

Athey, Susan, Julie Tibshirani, Stefan Wager, and others. 2019. "Generalized Random Forests." *The Annals of Statistics* 47 (2): 1148–78.

Athey, Susan, and Stefan Wager. 2019. "Estimating Treatment Effects with Causal Forests: An Application." *arXiv Preprint arXiv:1902.07409*.

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): 608–50.

Breiman, Leo. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32.

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

Imbens, Guido W. 2015. "Matching Methods in Practice: Three Examples." *Journal of Human Resources* 50 (2): 373–419.

Imbens, Guido W, and Donald B Rubin. 2015. *Causal Inference in Statistics, Social, and Biomedical Sciences.* Cambridge University Press.

Wager, Stefan, and Susan Athey. 2018. "Estimation and Inference of Heterogeneous Treatment Effects Using Random Forests." *Journal of the American Statistical Association* 113 (523): 1228–42.