

Matching and Re-weighting

When X differ...

Fernando Rios-Avila

Recap: Potential outcomes and Identification

To identify treatment effects one could **just** compare potential outcomes in two states:

- with treatment
- without treatment

Mathematically, average treatment effects would be:

$$ATE = E(Y_i(1) - Y_i(0))$$

the problem: with real data, we are only able to see one outcome. The counterfactual is not observed:

$$Y_i = Y_i(1) * D + Y_i(0) * (1 - D)$$

and simple differences may not capture ATE, because of selection bias and heterogeneity in effects.

Recap: Gold Standard - RCT

The easiest, but most expensive, way to deal with the problem is using **Randomized Control Trials**.

Effectively, you randomize Treatment, so that potential outcomes are independent of treatment:

$$Y(1), Y(0) \perp D$$

In other words, the distribution of potential outcomes is the same for those treated or untreated units.

$$\begin{aligned}
E(Y, D = 1) &= E(Y(1), D = 1) = E(Y(1), D = 0) \\
E(Y, D = 0) &= E(Y(0), D = 1) = E(Y(0), D = 0) \\
ATT &= E(Y, D = 1) - E(Y, D = 0)
\end{aligned}$$

But what if you can't Randomize

When unconditional fails

More often than not, specially if we didn't construct the data, it would be impossible to find that unconditional independence assumption holds.

For example, treatment (say having health insurance) may vary by age, gender, race, location, etc.

This is similar to the selection bias: Outcomes across treated and untreated groups will be different because:

- Composition: Characteristics of people among the treated could be different than those among the untreated. For example, they could be older, more educated, mostly men, etc.
- Other factors: There could be factors we cannot control for, that also affect outcomes.

There is conditional

When unconditional independence assumption fails, we can call on Conditional independence assumption:

$$Y(1), Y(0) \perp D | X$$

In other words, If we can look into specific groups (given X), it may be possible to impose the Independence assumption.

This relaxes the independence condition, but assumes selection is due to observable characteristics only. (it still needs to be as good as randomized given X)

Implications:

$$\begin{aligned}
E(Y|D = 1, X) &= E(Y(1)|D = 1, X) = E(Y(1)|D = 0, X) \\
E(Y|D = 0, X) &= E(Y(0)|D = 1, X) = E(Y(0)|D = 0, X)
\end{aligned}$$

Intuition

Matching is a methodology that falls within quasi-experimental designs. You cannot or could not decide the assignment rules, so now are using data as given.

The idea is to construct an artificial control and use it as a counter-factual, so that both treated and control groups “look similar” in terms of observables.

Once a group of synthetic controls has been constructed, treatment effects can be calculated for the whole population:

$$ATE(X) = E(Y|D = 1, X) - E(Y|D = 0, X)$$
$$ATE = \int ATE(X) dF_x$$

How can we do this?

we just need to find observational twins!

Matching Twins



Figure 1: Matching on Observables

Subclassification or stratification

Consider the following dataset:

```

frause titanic, clear
expand freq
drop if freq==0
gen class1=class==1
tab survived class1 , nofreq col

```

<IPython.core.display.HTML object>

(Data downloaded from R base)
 (8 zero counts ignored; observations not deleted)
 (2,177 observations created)
 (8 observations deleted)

		class1		
Survived		0	1	Total
No		72.92	37.54	67.70
Yes		27.08	62.46	32.30
Total		100.00	100.00	100.00

If we assume full Independence assumption we would believe that being in first class increased chance of survival in 35.4%. but is that the case?

What if the composition of individuals differs across classes (women and children)

```

tab age class1, nofreq col
tab sex class1, nofreq col

```

		class1		
Age		0	1	Total
Child		5.49	1.85	4.95
Adult		94.51	98.15	95.05
Total		100.00	100.00	100.00

		class1		
Sex		0	1	Total

Male		82.68	55.38		78.65
Female		17.32	44.62		21.35
-----+-----+-----					
Total		100.00	100.00		100.00

There were fewer children, but more women in first class. Perhaps that explains the difference in survival rates

A better approach would be to look into the survival probabilities stratifying the data:

```
gen surv=survived==2
bysort age sex class1:egen sr_mean=mean(survived==2)
table (age sex) (class1), stat(mean surv) nototal
```

		class1	
		0	1
-----+-----			
Age			
Child			
Sex			
Male		.4067797	1
Female		.6136364	1
Adult			
Sex			
Male		.1883378	.3257143
Female		.6263345	.9722222

So even within each group, the survival probability is larger in first class. What about Average?

```
bysort age sex:egen sr_mean_class1=max(sr_mean*(class1==1))
bysort age sex:egen sr_mean_class0=max(sr_mean*(class1==0))
gen teff = sr_mean_class1-sr_mean_class0
sum teff if class1==1 // ATT
sum teff if class1==0 // ATU
sum teff // ATE
```

Variable	Obs	Mean	Std. dev.	Min	Max
teff	325	.2375421	.1125033	.1373765	.5932204
Variable	Obs	Mean	Std. dev.	Min	Max
teff	1,876	.1887847	.1089261	.1373765	.5932204
Variable	Obs	Mean	Std. dev.	Min	Max
teff	2,201	.1959842	.1107948	.1373765	.5932204

What did we do?

The procedure above is a simple stratification approach, aka matching, to analyze the true impact of the treatment (being a 1st class passenger).

1. Stratified the sample in groups by age and gender.
 - Identify the shares of each group by class1
2. Predict probability of survival per strata and class1
3. Obtain the Strata level Effects
4. Aggregate as needed.
 - Here, we could estimate ATE, ATT or ATU!

Where could things go wrong?

Overlapping

The procedure describe above works well whenever there is data overlapping.

- For every combination of X, you see data on the control and treated group $0 < P(D|X) < 1$

When this fails, you wont be able to estimate ATE's, although ATT's or ATU's might still be possible:

- for ATT: $P(D|X) < 1$
- for ATU: $0 < P(D|X)$

For example:

```
frause hhprice, clear
keep price rooms type_h
tab rooms type_h
```

Number of rooms	=0 if house, =1 TownHouse		Total
	0	1	
1	37	72	109
2	1,134	751	1,885
3	4,634	648	5,282
4	2,465	115	2,580
5	465	2	467
6	46	0	46
7	7	0	7
Total	8,788	1,588	10,376

Would not be able to estimate ATE nor ATU. Only ATT for townhouses.

Curse of dimensionality

There is a second problem in terms of stratification. How would we deal with Multiple dimensions? Would it be possible to find “twins” for every observation?

The answer is, probably no. Too many groups to track, too many micro cells to make use of:

```
frause oaxaca, clear
drop if lnwage==.
egen strata=group(educ isco)
bysort strata:egen flag=mean(female)
list educ isco female if (flag==0 | flag==1) & educ == 10, sep(0)
```

(Excerpt from the Swiss Labor Market Survey 1998)
(213 observations deleted)

```
+-----+
| educ   isco   female |
```

158.	10	1	0
159.	10	1	0
197.	10	7	0
198.	10	7	0
199.	10	9	1
200.	10	9	1
	+-----+		

Alternative: Matching as a weighted

The problem of curse of dimensional states that as the number of desired characteristics to match increase, fewer “twins” will be available in the data. At the end...no one will be like you!

The alternative, is to look into People that are sufficiently close so they can be used for matching.

$$ATT_i = Y_i - \sum_{j \in C} w(x_j, x_i) Y_j$$

$$ATT = \frac{1}{N_T} \sum (ATT_i)$$

$$ATT = E(Y|D = 1) - E_i \left(\sum_{j \in C} w(x_j, x_i) Y_j \middle| D = 0 \right)$$

Depending how $w(\cdot)$ is defined, we would be facing different kinds of matching estimators.

Types of Matching

Matching on covariates

The first decision to take is whether one should find matches based on covariates, or based on scores (propensity scores).

Using covariates implies that will aim to find the closest “twin” possible, based on multiple dimensions:

$$Eclidean = d(x_i, x_j) = \sqrt{(x_i - x_j)'(x_i - x_j)}$$

$$WEclidean = d(x_i, x_j) = \sqrt{(x_i - x_j)'W(x_i - x_j)}$$

$$Maha = d(x_i, x_j) = \sqrt{(x_i - x_j)'S^{-1}(x_i - x_j)}$$

Distance measures are used to identify the closest matches to a given observation, and thus the weight assigned to that observation.

Has the advantage of looking at individuals who are indeed close to each other, but becomes more difficult as the dimensionality of X 's increase. (you will not find close matches)

Matching on Scores

A second approach is to match individuals based on some summary index that condenses the information in X into a single scalar $h(x)$, reducing the dimensionality problem from K to 1.

Few candidates:

- Propensity Score: $P(D|X)$ based on a logit/probit/binomial model. Most common approach!
- Predicted Mean: $X\beta$ if there is information on outcome to be predicted
- PCA: Using Principal components to reduce dimensionality before Matching

Since there is only 1 dimension to consider, multiple distance measures are possible:

- nearest neighbors, kernel weight matching, radius matching.

But one has to be careful with the approach. King and Nielsen (2019) Argue about the risks of PSM

1 vs K matching; With and without replacement

Two additional questions remain regarding matching. How many “twins” to use, and if twins will be obtained with/without replacement.

- Fewer matches reduce bias (choosing only the closest observation), but increase variance.
- More matches increase bias, but reduce variance. (because of less optimal matches)
- with replacement: control units may be used more than once. This will improve matching quality reducing bias. But by using the same units multiple times, it will increase variance.
- without replacement: Control units are used once, potentially reducing matching quality, but reducing variance. It will be order dependent.

see Caliendo and Kopeing (2008)

What about SE? and Statistical inference?

Well....this is one of the few cases where Bootstrapping WON'T work!

Standard errors are more cumbersome. So we will just rely on software results

Other considerations

Once you have chosen your matching method, find your “statistical twins”, and estimate your differences you are done! (or are you)

Not yet...common practice: Evaluate the balance of your data

Matching aims to reduce or eliminate differences in characteristics between treatment and control units. Thus, one should evaluate the differences (before and after match) of your characteristics

1. Check for overlapping condition.
 - either variable by variable or with pscore
2. Assess Matching Quality: Have differences across groups vanished?
 - Check Standardized differences $\frac{\mu_1 - \mu_2}{\sqrt{0.5 * (V_1 + V_2)}}$
 - t-tests
 - PR2 of regression with matched data

Implementation

In **Stata**, there are at least two approaches that can be used for matching:

- `psmatch2` (from `ssc`)
- `teffects` (Official **Stata** command)

We will use this to answer a simple question:

What is the impact of Training Jobs on Earnings?

Example

This file contains information on experimental and observed data for the analysis of training on earnings program:

```
use https://friosavila.github.io/playingwithstata/drddid/lalonde.dta, clear
keep if year==1978
drop if dwincl==0
label define sample 1 "exper" 2 "CPS" 3 "PSID"
label values sample sample
tab sample treated, m
```

(19,204 observations deleted)

(277 observations deleted)

sample	treated		.	Total
	0	1		
exper	260	185	0	445
CPS	0	0	15,992	15,992
PSID	0	0	2,490	2,490
Total	260	185	18,482	18,927

First Experimental design - RCT

```
reg re treated
tabstat age educ black married nodegree , by(treated)
logit treated age educ black hisp married nodegree
```

Source	SS	df	MS	Number of obs	=	445
Model	348013183	1	348013183	F(1, 443)	=	8.04
Residual	1.9178e+10	443	43290369.3	Prob > F	=	0.0048
Total	1.9526e+10	444	43976681.9	R-squared	=	0.0178
				Adj R-squared	=	0.0156
				Root MSE	=	6579.5

re	Coefficient	Std. err.	t	P> t	[95% conf. interval]
----	-------------	-----------	---	------	----------------------

treated	1794.342	632.8534	2.84	0.005	550.5745	3038.11
_cons	4554.801	408.0459	11.16	0.000	3752.855	5356.747

Summary statistics: Mean
Group variable: treated

treated	age	educ	black	married	nodegree
0	25.05385	10.08846	.8269231	.1538462	.8346154
1	25.81622	10.34595	.8432432	.1891892	.7081081
Total	25.37079	10.19551	.8337079	.1685393	.7820225

Iteration 0: Log likelihood = -302.1
Iteration 1: Log likelihood = -294.72908
Iteration 2: Log likelihood = -294.71464
Iteration 3: Log likelihood = -294.71464

Logistic regression

Number of obs = 445
LR chi2(6) = 14.77
Prob > chi2 = 0.0221
Pseudo R2 = 0.0244

Log likelihood = -294.71464

treated	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
age	.0059171	.0142668	0.41	0.678	-.0220452	.0338794
educ	-.0639597	.071354	-0.90	0.370	-.203811	.0758916
black	-.2543689	.3639735	-0.70	0.485	-.9677438	.4590061
hisp	-.8291587	.5042305	-1.64	0.100	-1.817432	.159115
married	.2342415	.2661824	0.88	0.379	-.2874665	.7559495
nodegree	-.8385524	.3093833	-2.71	0.007	-1.444933	-.2321722
_cons	1.053028	1.047384	1.01	0.315	-.9998064	3.105862

Then using PScore Matching CPS

```

keep if treated == 1 | sample ==2
replace treated=0 if treated==.
reg re treated
tabstat age educ black hisp married nodegree , by(treated)

```

(2,750 observations deleted)
(15,992 real changes made)

Source		SS	df	MS	Number of obs	=	16,177
-----+							
Model		1.3206e+10	1	1.3206e+10	F(1, 16175)	=	142.43
Residual		1.4997e+12	16,175	92717515.8	Prob > F	=	0.0000
-----+							
					R-squared	=	0.0087
					Adj R-squared	=	0.0087
Total		1.5129e+12	16,176	93528158.4	Root MSE	=	9629

re		Coefficient	Std. err.	t	P> t	[95% conf. interval]
-----+						
treated		-8497.516	712.0207	-11.93	0.000	-9893.156 -7101.877
_cons		14846.66	76.14292	194.98	0.000	14697.41 14995.91

Summary statistics: Mean
Group variable: treated

treated		age	educ	black	hisp	married	nodegree
-----+							
0		33.22524	12.02751	.0735368	.072036	.7117309	.2958354
1		25.81622	10.34595	.8432432	.0594595	.1891892	.7081081
-----+							
Total		33.14051	12.00828	.0823391	.0718922	.7057551	.3005502

We need to do trimming

```

bysort educ black hisp married:egen n11=sum(treated==1)
bysort age black hisp married:egen n22=sum(treated==1)
drop if n11==0 | n22 ==0

```

```
tabstat age educ black hisp married nodegree , by(treated)
reg re treated
```

(13,536 observations deleted)

Summary statistics: Mean
Group variable: treated

treated	age	educ	black	hisp	married	nodegree
0	24.24145	11.69788	.252443	.0260586	.3346906	.2569218
1	25.81622	10.34595	.8432432	.0594595	.1891892	.7081081
Total	24.35176	11.60318	.2938281	.0283983	.3244983	.2885271

Source	SS	df	MS	Number of obs	=	2,641
Model	5.7607e+09	1	5.7607e+09	F(1, 2639)	=	73.89
Residual	2.0575e+11	2,639	77964783.1	Prob > F	=	0.0000
Total	2.1151e+11	2,640	80117339.3	R-squared	=	0.0272
				Adj R-squared	=	0.0269
				Root MSE	=	8829.8

re	Coefficient	Std. err.	t	P> t	[95% conf. interval]
treated	-5786.584	673.1834	-8.60	0.000	-7106.605 -4466.564
_cons	12135.73	178.1702	68.11	0.000	11786.36 12485.1

Lets do some matching

```
teffects nnmatch (re age educ black married nodegree ) (treated)
tebalance summarize
teffects nnmatch (re age educ black married nodegree ) (treated), nn(2)
tebalance summarize
teffects psmatch (re) (treated age educ black married nodegree )
tebalance summarize
```

```
teffects psmatch (re) (treated age educ black married nodegree ) , nn(2)
tebalance summarize
```

```
Treatment-effects estimation      Number of obs      =      2,641
Estimator      : nearest-neighbor matching      Matches: requested =      1
Outcome model  : matching                      min =      1
Distance metric: Mahalanobis                  max =      138
```

```
-----+-----
              |               AI robust
              | Coefficient  std. err.      z    P>|z|      [95% conf. interval]
-----+-----
ATE          |
   treated   |
   (1 vs 0)  |  -3685.665   1188.666   -3.10   0.002   -6015.407   -1355.923
-----+-----
```

(refitting the model using the generate() option)

Covariate balance summary

```
-----+-----
              Raw      Matched
-----+-----
Number of obs =      2,641      5,282
Treated obs   =      185      2,641
Control obs   =      2,456      2,641
-----+-----
```

```
-----+-----
              |Standardized differences      Variance ratio
              |      Raw      Matched      Raw      Matched
-----+-----
   age |   .2342346   -.015417   1.305844   .8410946
   educ |  -.7684118  -.0812288   1.881909   .8598207
   black |   1.473105      0      .7039609      1
 married |  -.3351313  -.0008087   .6923501   .999393
 nodegree |   1.010393      0   1.088086      1
-----+-----
```

```
Treatment-effects estimation      Number of obs      =      2,641
Estimator      : nearest-neighbor matching      Matches: requested =      2
Outcome model  : matching                      min =      2
Distance metric: Mahalanobis                  max =      138
```

		AI robust				
re	Coefficient	std. err.	z	P> z	[95% conf. interval]	
ATE						
treated						
(1 vs 0)	-5166.888	1107.653	-4.66	0.000	-7337.848	-2995.929

(refitting the model using the generate() option)

Covariate balance summary

	Raw	Matched
Number of obs =	2,641	5,282
Treated obs =	185	2,641
Control obs =	2,456	2,641

	Standardized differences		Variance ratio	
	Raw	Matched	Raw	Matched
age	.2342346	-.0209048	1.305844	.7345997
educ	-.7684118	-.0385284	1.881909	.8978301
black	1.473105	.0074673	.7039609	1.006716
married	-.3351313	-.004586	.6923501	.9965432
nodegree	1.010393	.0016705	1.088086	1.001557

Treatment-effects estimation	Number of obs	=	2,641
Estimator : propensity-score matching	Matches: requested	=	1
Outcome model : matching	min	=	1
Treatment model: logit	max	=	138

	AI robust					
re	Coefficient	std. err.	z	P> z	[95% conf. interval]	
ATE						
treated						
(1 vs 0)	-4278.549	1135.847	-3.77	0.000	-6504.768	-2052.331

(refitting the model using the generate() option)

Covariate balance summary

	Raw	Matched
Number of obs =	2,641	5,282
Treated obs =	185	2,641
Control obs =	2,456	2,641

	Standardized differences		Variance ratio	
	Raw	Matched	Raw	Matched
age	.2342346	.0014058	1.305844	.9313458
educ	-.7684118	-.1308249	1.881909	.9665937
black	1.473105	-.0926638	.7039609	.90999
married	-.3351313	-.0973289	.6923501	.9197524
nodegree	1.010393	.0821105	1.088086	1.07103

Treatment-effects estimation Number of obs = 2,641
Estimator : propensity-score matching Matches: requested = 2
Outcome model : matching min = 2
Treatment model: logit max = 138

	re	Coefficient	AI robust std. err.	z	P> z	[95% conf. interval]	
ATE							
treated							
(1 vs 0)		-4380.078	1158.019	-3.78	0.000	-6649.754	-2110.403

(refitting the model using the generate() option)

Covariate balance summary

	Raw	Matched
Number of obs =	2,641	5,282
Treated obs =	185	2,641
Control obs =	2,456	2,641

		Standardized differences		Variance ratio	
		Raw	Matched	Raw	Matched

	+				
age		.2342346	-.06133	1.305844	.8834346
educ		-.7684118	-.1321518	1.881909	1.021302
black		1.473105	-.0698339	.7039609	.933348
married		-.3351313	-.0414439	.6923501	.9674741
nodegree		1.010393	.0939209	1.088086	1.080951

A missing variable? Earnings in previous year. May capture information of Need to do treatment (selection)

```
tabstat age educ black hisp married nodegree re74, by(treated)
gen dre = re-re74
teffects nnmatch (dre age educ black married nodegree ) (treated)

teffects nnmatch (dre age educ black married nodegree ) (treated), nn(2)

teffects psmatch (dre) (treated age educ black married nodegree )

teffects psmatch (dre) (treated age educ black married nodegree ) , nn(2)
```

Summary statistics: Mean
Group variable: treated

treated		age	educ	black	hisp	married	nodegree

0		24.24145	11.69788	.252443	.0260586	.3346906	.2569218
1		25.81622	10.34595	.8432432	.0594595	.1891892	.7081081

Total		24.35176	11.60318	.2938281	.0283983	.3244983	.2885271

treated		re74

0	9347.406
1	2095.574
-----+-----	
Total	8839.421

Treatment-effects estimation	Number of obs	=	2,641
Estimator : nearest-neighbor matching	Matches: requested	=	1
Outcome model : matching	min	=	1
Distance metric: Mahalanobis	max	=	138

		AI robust				
	dre	Coefficient	std. err.	z	P> z	[95% conf. interval]
-----+-----						
ATE						
treated						
(1 vs 0)		2616.653	1803.172	1.45	0.147	-917.4997 6150.806

Treatment-effects estimation	Number of obs	=	2,641
Estimator : nearest-neighbor matching	Matches: requested	=	2
Outcome model : matching	min	=	2
Distance metric: Mahalanobis	max	=	138

		AI robust				
	dre	Coefficient	std. err.	z	P> z	[95% conf. interval]
-----+-----						
ATE						
treated						
(1 vs 0)		730.2925	1674.91	0.44	0.663	-2552.47 4013.055

Treatment-effects estimation	Number of obs	=	2,641
Estimator : propensity-score matching	Matches: requested	=	1
Outcome model : matching	min	=	1
Treatment model: logit	max	=	138

		AI robust				
	dre	Coefficient	std. err.	z	P> z	[95% conf. interval]
-----+-----						
ATE						
treated						
(1 vs 0)		2162.311	1740.12	1.24	0.214	-1248.262 5572.884

Treatment-effects estimation		Number of obs	=	2,641
Estimator	: propensity-score matching	Matches: requested	=	2
Outcome model	: matching	min	=	2
Treatment model	: logit	max	=	138

		AI robust				
	dre	Coefficient	std. err.	z	P> z	[95% conf. interval]
ATE						
treated						
(1 vs 0)		1833.03	1739.496	1.05	0.292	-1576.318 5242.379

In this case, Matching alone could not get the right answer. Who were the most likely to “go to the training?”

So instead we change the question: How much the change in earnings compare across groups.

Wait: What about Reweighting?

An alternative method to Matching is to do Re-weighting.

We have seen this!

Your control group has a distribution $g(x)$ and your treatment $f(x)$. We can use some weighting factors $h(x)$ that reshapes $g(x) \rightarrow \hat{f}(x)$.

How? Using Propensity scores

Why does it work? Just as matching, your goal is to compare distributions of outcomes, forcing differences in observed characteristics to be the same.

IPW, does this by reweighting the distribution! (rather than matching)

Inverse Probability Weighting:IPW

S1: Estimate Pscore

$$p(D = 1|X) = F(X\beta)$$

S2: Estimate IPW

For ATT: $W(D = 1, x) = 1$ & $W(D = 0, X) = \frac{\hat{p}(x)}{1-\hat{p}(x)}$

For ATU: $W(D = 0, x) = 1$ & $W(D = 1, X) = \frac{1-\hat{p}(x)}{\hat{p}(x)}$

For ATE: $W(D = 0, x) = \frac{1}{1-\hat{p}(x)}$ & $W(D = 1, X) = \frac{1}{\hat{p}(x)}$

s3: Estimate Treatment effect:

$$TE = \sum_{i \in D=1} w_i^s(1)Y_i - \sum_{i \in D=0} w_i^s(0)Y_i$$

Even Better: Go DR!

An interesting advantage of IPW approach is that you can gain efficiency using Doubly Robust Methods. Namely, instead of comparing outcomes directly, you could compare predicted outcomes!

$$ATT = \frac{1}{N_t} \sum (Y_1 - X' \hat{\beta}_w^0)$$

$$ATU = \frac{1}{N_c} \sum (X' \hat{\beta}_w^1 - Y_0)$$

$$ATE = \frac{1}{N} \sum (X' \hat{\beta}_w^1 - X' \hat{\beta}_w^0)$$

where $\hat{\beta}_w^k$ can be modeled using weighted least squares

Comparing to Matching

```
teffects ipw (re) (treated age educ black married nodegree) , iter(3) nolog
teffects ipwra (re age educ black married nodegree) (treated age educ black married nodegree) , iter(3) nolog
teffects ipw (dre) (treated age educ black married nodegree), iter(3) nolog
teffects ipwra (dre age educ black married nodegree) (treated age educ black married nodegree) , iter(3) nolog
```

```
Treatment-effects estimation      Number of obs      =      2,641
Estimator      : inverse-probability weights
Outcome model  : weighted mean
Treatment model: logit

-----
               |               Robust
               |               std. err.      z    P>|z|      [95% conf. interval]
re | Coefficient
```

ATE							
treated							
(1 vs 0)		-4833.352	1088.667	-4.44	0.000	-6967.101	-2699.603
POmean							
treated							
0		11979.19	179.1903	66.85	0.000	11627.99	12330.4
Treatment-effects estimation							
				Number of obs	=	2,641	
Estimator		: IPW regression adjustment					
Outcome model		: linear					
Treatment model		: logit					

		Robust					
re		Coefficient	std. err.	z	P> z	[95% conf. interval]	
ATE							
treated							
(1 vs 0)		-4835.38	1012.598	-4.78	0.000	-6820.036	-2850.724
POmean							
treated							
0		11976.52	179.0958	66.87	0.000	11625.49	12327.54
Warning: Convergence not achieved.							
Treatment-effects estimation							
				Number of obs	=	2,641	
Estimator		: inverse-probability weights					
Outcome model		: weighted mean					
Treatment model		: logit					

		Robust					
dre		Coefficient	std. err.	z	P> z	[95% conf. interval]	
ATE							
treated							
(1 vs 0)		1475.71	1792.427	0.82	0.410	-2037.382	4988.802
POmean							
treated							

