Title

teffects aipw — Augmented inverse-probability weighting

Description

`teffects aipw` estimates the average treatment effect (ATE) and the potential-outcome means (POMs) from observational data by augmented inverse-probability weighting (AIPW). AIPW estimators combine aspects of regression-adjustment and inverse-probability-weighted methods. AIPW estimators have the double-robust property. `teffects aipw` accepts a continuous, binary, count, fractional, or nonnegative outcome and allows a multivalued treatment.

See [TE] **teffects intro** or [TE] **teffects intro advanced** for more information about estimating treatment effects from observational data.

Quick start

ATE of binary treatment `treat2` by AIPW using a linear model for outcome `y1` on `x1` and `x2` and a logistic model for `treat2` on `x1` and `w`

```
teffects aipw (y1 x1 x2) (treat2 x1 w)
```

As above, but use a fractional logistic model for fractional outcome `y2`

```
teffects aipw (y2 x1 x2, flogit) (treat2 x1 w)
```

As above, but use a heteroskedastic probit model for binary outcome `y3` and a probit model for `treat2`

```
teffects aipw (y3 x1 x2, hetprobit(x1 x2)) (treat2 x1 w, probit)
```

ATE for each level of three-valued treatment `treat3` on `y1`

```
teffects aipw (y1 x1 x2) (treat3 x1 w)
```

As above, and specify that `treat3` = 3 is the control level

```
teffects aipw (y1 x1 x2) (treat3 x1 w), control(3)
```

Same as above, specified using the label "MyControl" corresponding to `treat3` = 3

```
teffects aipw (y1 x1 x2) (treat3 x1 w), control(MyControl)
```

Menu

Statistics > Treatment effects > Continuous outcomes > Augmented inverse-probability weighting

Statistics > Treatment effects > Binary outcomes > Augmented inverse-probability weighting

Statistics > Treatment effects > Count outcomes > Augmented inverse-probability weighting

Statistics > Treatment effects > Fractional outcomes > Augmented inverse-probability weighting

Statistics > Treatment effects > Nonnegative outcomes > Augmented inverse-probability weighting

Syntax

```
teffects aipw (ovar omvarlist [, omodel noconstant])
  (tvar tmvarlist [, tmodel noconstant]) [if] [in] [weight]
  [, stat options]
```

ovar is a binary, count, continuous, fractional, or nonnegative outcome of interest.

omvarlist specifies the covariates in the outcome model.

tvar must contain integer values representing the treatment levels.

tmvarlist specifies the covariates in the treatment-assignment model.

omodel	Description
Model	
`linear`	linear outcome model; the default
`logit`	logistic outcome model
`probit`	probit outcome model
`hetprobit(`*varlist*`)`	heteroskedastic probit outcome model
`poisson`	exponential outcome model
`flogit`	fractional logistic outcome model
`fprobit`	fractional probit outcome model
`fhetprobit(`*varlist*`)`	fractional heteroskedastic probit outcome model

omodel specifies the model for the outcome variable.

tmodel	Description
Model	
`logit`	logistic treatment model; the default
`probit`	probit treatment model
`hetprobit(`*varlist*`)`	heteroskedastic probit treatment model

tmodel specifies the model for the treatment variable.
For multivalued treatments, only `logit` is available and multinomial logit is used.

stat	Description
Stat	
`ate`	estimate average treatment effect in population; the default
`pomeans`	estimate potential-outcome means

options	Description
Model	
`nls`	estimate conditional means by nonlinear least squares
`wnls`	estimate conditional means by weighted nonlinear least squares
SE/Robust	
`vce(`*vcetype*`)`	*vcetype* may be `robust`, `cluster` *clustvar*, `bootstrap`, or `jackknife`
Reporting	
`level(#)`	set confidence level; default is `level(95)`
`aequations`	display auxiliary-equation results
display_options	control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
Maximization	
maximize_options	control the maximization process; seldom used
Advanced	
`pstolerance(#)`	set tolerance for overlap assumption
`osample(`*newvar*`)`	*newvar* identifies observations that violate the overlap assumption
`control(#`|*label*`)`	specify the level of *tvar* that is the control
`coeflegend`	display legend instead of statistics

omvarlist and *tmvarlist* may contain factor variables; see **[U] 11.4.3 Factor variables**.

`bootstrap`, `by`, `collect`, `jackknife`, and `statsby` are allowed; see **[U] 11.1.10 Prefix commands**.

Weights are not allowed with the `bootstrap` prefix; see [R] **bootstrap**.

`fweights` and `iweights` are allowed; see **[U] 11.1.6 weight**.

`coeflegend` does not appear in the dialog box.

See **[U] 20 Estimation and postestimation commands** for more capabilities of estimation commands.

Options

Model

`noconstant`; see [R] **Estimation options**.

`nls` specifies that the parameters of the outcome model be estimated by nonlinear least squares instead of the default maximum likelihood.

`wnls` specifies that the parameters of the outcome model be estimated by weighted nonlinear least squares instead of the default maximum likelihood. The weights make the estimator of the effect parameters more robust to a misspecified outcome model.

Stat

stat is one of two statistics: `ate` or `pomeans`. `ate` is the default.

`ate` specifies that the average treatment effect be estimated.

`pomeans` specifies that the potential-outcome means for each treatment level be estimated.

SE/Robust

`vce(vcetype)` specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (`robust`), that allow for intragroup correlation (`cluster` *clustvar*), and that use bootstrap or jackknife methods (`bootstrap`, `jackknife`); see [R] ***vce_option***.

Reporting

`level(#)`; see [R] **Estimation options**.

`aequations` specifies that the results for the outcome-model or the treatment-model parameters be displayed. By default, the results for these auxiliary parameters are not displayed.

display_options: `noci`, `nopvalues`, `noomitted`, `vsquish`, `noemptycells`, `baselevels`, `allbaselevels`, `nofvlabel`, `fvwrap(#)`, `fvwrapon(style)`, `cformat(%fmt)`, `pformat(%fmt)`, `sformat(%fmt)`, and `nolstretch`; see [R] **Estimation options**.

Maximization

maximize_options: `iterate(#)`, [`no`]`log`, and `from(`*init_specs*`)`; see [R] **Maximize**. These options are seldom used.

init_specs is one of

matname [`, skip copy`]

[`, #` ...], `copy`

Advanced

`pstolerance(#)` specifies the tolerance used to check the overlap assumption. The default value is `pstolerance(1e-5)`. `teffects` will exit with an error if an observation has an estimated propensity score smaller than that specified by `pstolerance()`.

`osample(`*newvar*`)` specifies that indicator variable *newvar* be created to identify observations that violate the overlap assumption.

`control(`*#* | *label*`)` specifies the level of *tvar* that is the control. The default is the first treatment level. You may specify the numeric level *#* (a nonnegative integer) or the label associated with the numeric level. `control()` may not be specified with statistic `pomeans`.

The following option is available with `teffects aipw` but is not shown in the dialog box:

`coeflegend`; see [R] **Estimation options**.

Remarks and examples

stata.com

Remarks are presented under the following headings:

Overview

AIPW estimators use inverse-probability weights to correct for the missing-data problem arising from the fact that each subject is observed in only one of the potential outcomes; these estimators also use an augmentation term in the outcome model to correct the estimator in case the treatment model is misspecified. If the treatment model is correctly specified, the augmentation term goes to zero in large samples.

AIPW estimators compute averages of the augmented inverse-probability-weighted outcomes for each treatment level. Contrasts of these averages provide estimates of the treatment effects.

AIPW estimators use a model to predict treatment status, and they use another model to predict outcomes. Because of the double-robust property, only one of these two models must be correctly specified for the AIPW estimator to be consistent.

AIPW estimators use a three-step approach to estimating treatment effects:

1. They estimate the parameters of the treatment model and compute inverse-probability weights.
2. They estimate separate regression models of the outcome for each treatment level and obtain the treatment-specific predicted outcomes for each subject.
3. They compute the weighted means of the treatment-specific predicted outcomes, where the weights are the inverse-probability weights computed in step 1. The contrasts of these weighted averages provide the estimates of the ATEs.

These steps produce consistent estimates of the effect parameters because the treatment is assumed to be independent of the potential outcomes after conditioning on the covariates. The overlap assumption ensures that predicted inverse-probability weights do not get too large. The standard errors reported by `teffects aipw` correct for the three-step process. See [TE] **teffects intro** or [TE] **teffects intro advanced** for more information about this estimator.

We will illustrate the use of `teffects aipw` by using data from a study of the effect of a mother's smoking status during pregnancy (`mbsmoke`) on infant birthweight (`bweight`) as reported by Cattaneo (2010). This dataset also contains information about each mother's age (`mage`), education level (`medu`), marital status (`mmarried`), whether the first prenatal exam occurred in the first trimester (`prenatal1`), and whether this baby was the mother's first birth (`fbaby`).

▷ Example 1: Estimating the ATE

We begin by using `teffects aipw` to estimate the average treatment effect of `mbsmoke` on `bweight`. We use a probit model to predict treatment status as a function of `mmarried`, `mage`, and `fbaby`; to maximize the predictive power of this model, we use factor-variable notation to incorporate quadratic effects of the mother's age, the only continuous covariate in our model. We use linear regression to model birthweight, using `prenatal1`, `mmarried`, `mage`, and `fbaby` as explanatory variables. We type

```
. use https://www.stata-press.com/data/r17/cattaneo2
(Excerpt from Cattaneo (2010) Journal of Econometrics 155: 138-154)
. teffects aipw (bweight prenatal1 mmarried mage fbaby)
> (mbsmoke mmarried c.mage##c.mage fbaby medu, probit)
Iteration 0:   EE criterion =  4.629e-21
Iteration 1:   EE criterion =  1.944e-25
Treatment-effects estimation                   Number of obs     =      4,642
Estimator      : augmented IPW
Outcome model  : linear by ML
Treatment model: probit
```

bweight	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
ATE						
mbsmoke (Smoker vs Nonsmoker)	-230.9892	26.21056	-8.81	0.000	-282.361	-179.6174
POmean						
mbsmoke Nonsmoker	3403.355	9.568472	355.68	0.000	3384.601	3422.109

The average birthweight if all mothers were to smoke would be 231 grams less than the average of 3,403 grams that would occur if none of the mothers had smoked.

◁

By default, `teffects aipw` reports the ATE and the POM for the base (untreated) subjects. The `pomeans` option allows us to view the treated subjects' POM as well; the `aequations` option displays the regression model coefficients used to predict the POMs as well as the coefficients from the model used to predict treatment.

▷ Example 2: Displaying the POMs and equations

Here we use the `pomeans` and `aequations` options to obtain estimates of both POMs and view all the fitted equations underlying our estimates:

```
. teffects aipw (bweight prenatal1 mmarried mage fbaby)
> (mbsmoke mmarried c.mage##c.mage fbaby medu, probit), pomeans aequations
Iteration 0:   EE criterion =  4.629e-21
Iteration 1:   EE criterion =  6.856e-26
Treatment-effects estimation                   Number of obs     =      4,642
Estimator      : augmented IPW
Outcome model  : linear by ML
Treatment model: probit
```

bweight	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
POmeans						
mbsmoke						
Nonsmoker	3403.355	9.568472	355.68	0.000	3384.601	3422.109
Smoker	3172.366	24.42456	129.88	0.000	3124.495	3220.237
OME0						
prenatal1	64.40859	27.52699	2.34	0.019	10.45669	118.3605
mmarried	160.9513	26.6162	6.05	0.000	108.7845	213.1181
mage	2.546828	2.084324	1.22	0.222	-1.538373	6.632028
fbaby	-71.3286	19.64701	-3.63	0.000	-109.836	-32.82117
_cons	3202.746	54.01082	59.30	0.000	3096.886	3308.605
OME1						
prenatal1	25.11133	40.37541	0.62	0.534	-54.02302	104.2457
mmarried	133.6617	40.86443	3.27	0.001	53.5689	213.7545
mage	-7.370881	4.21817	-1.75	0.081	-15.63834	.8965804
fbaby	41.43991	39.70712	1.04	0.297	-36.38461	119.2644
_cons	3227.169	104.4059	30.91	0.000	3022.537	3431.801
TME1						
mmarried	-.6484821	.0554173	-11.70	0.000	-.757098	-.5398663
mage	.1744327	.0363718	4.80	0.000	.1031452	.2457202
c.mage#						
c.mage	-.0032559	.0006678	-4.88	0.000	-.0045647	-.0019471
fbaby	-.2175962	.0495604	-4.39	0.000	-.3147328	-.1204595
medu	-.0863631	.0100148	-8.62	0.000	-.1059917	-.0667345
_cons	-1.558255	.4639691	-3.36	0.001	-2.467618	-.6488926

The coefficient table indicates that the treated POM is 3,172 grams, 231 grams less than the untreated POM. The sections of the table labeled `OME0` and `OME1` represent the linear regression coefficients for the untreated and treated potential-outcome equations, respectively. The coefficients of the `TME1` equation are used in the probit model to predict treatment status.

◁

As is well known, the standard probit model assumes that the error terms in the latent-utility framework are homoskedastic; the model is not robust to departures from this assumption. An alternative is to use the heteroskedastic probit model, which explicitly models the error variance as a function of a set of variables.

▷ Example 3: Heteroskedastic probit treatment model

Here we refit our model as in the previous examples, but we instead use heteroskedastic probit to model the treatment variable. We posit that the heteroskedasticity is a function of the mother's age. We type

```
. teffects aipw (bweight prenatal1 mmarried fbaby)
> (mbsmoke mmarried c.mage##c.mage fbaby medu, hetprobit(c.mage)), aequations
Iteration 0:   EE criterion =  1.746e-19
Iteration 1:   EE criterion =  1.746e-19  (backed up)
Treatment-effects estimation                   Number of obs     =      4,642
Estimator      : augmented IPW
Outcome model  : linear by ML
Treatment model: heteroskedastic probit
```

bweight	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
ATE						
mbsmoke (Smoker vs Nonsmoker)	-230.2699	27.49461	-8.38	0.000	-284.1584	-176.3815
POmean						
mbsmoke Nonsmoker	3403.657	9.540713	356.75	0.000	3384.957	3422.356
OME0						
prenatal1	69.5048	27.04642	2.57	0.010	16.49479	122.5148
mmarried	173.74	24.63865	7.05	0.000	125.4491	222.0308
fbaby	-79.19473	18.62584	-4.25	0.000	-115.7007	-42.68875
_cons	3260.768	28.29282	115.25	0.000	3205.315	3316.221
OME1						
prenatal1	12.86437	39.83916	0.32	0.747	-65.21894	90.94768
mmarried	113.3491	39.47422	2.87	0.004	35.9811	190.7172
fbaby	64.22326	38.42042	1.67	0.095	-11.07939	139.5259
_cons	3051.268	37.30413	81.79	0.000	2978.153	3124.383
TME1						
mmarried	-.3551755	.1044199	-3.40	0.001	-.5598347	-.1505162
mage	.0831898	.0349088	2.38	0.017	.0147699	.1516097
c.mage# c.mage	-.0013458	.0006659	-2.02	0.043	-.002651	-.0000406
fbaby	-.1170697	.044998	-2.60	0.009	-.2052643	-.0288752
medu	-.0435057	.0147852	-2.94	0.003	-.0724842	-.0145272
_cons	-.8757021	.347814	-2.52	0.012	-1.557405	-.1939993
TME1_lnsigma						
mage	-.0236336	.0107134	-2.21	0.027	-.0446315	-.0026357

The equation labeled `TME1_lnsigma` represents the heteroskedasticity function used to model the logarithm of the variance. Because the coefficient on the single variable we specified is significant below the 5% level, we conclude that allowing for heteroskedasticity was indeed necessary.

◁

Rather than using maximum likelihood to fit the outcome model, you can instruct `teffects aipw` to use a weighted nonlinear least-squares (WNLS) estimator instead. The WNLS estimator may be more robust to outcome model misspecification.

▷ Example 4: Using the WNLS estimator

Here we use WNLS to fit our outcome model:

```
. use https://www.stata-press.com/data/r17/cattaneo2
(Excerpt from Cattaneo (2010) Journal of Econometrics 155: 138-154)

. teffects aipw (bweight prenatal1 mmarried mage fbaby)
> (mbsmoke mmarried c.mage##c.mage fbaby medu, probit), wnls

Iteration 0:   EE criterion =  2.742e-20
Iteration 1:   EE criterion =  3.436e-24

Treatment-effects estimation                   Number of obs     =      4,642
Estimator      : augmented IPW
Outcome model  : linear by WNLS
Treatment model: probit
```

bweight	Coefficient	Robust std. err.	z	P>|z|	[95% conf.	interval]
ATE						
mbsmoke (Smoker vs Nonsmoker)	-227.1956	27.34794	-8.31	0.000	-280.7966	-173.5946
POmean						
mbsmoke Nonsmoker	3403.251	9.596622	354.63	0.000	3384.442	3422.06

The ATE of −227 is slightly greater than the ATE of −231 estimated in example 1. The estimated POMs are nearly indistinguishable.

◁

Video example

Treatment effects: Augmented inverse-probability weighting

Stored results

`teffects aipw` stores the following in `e()`:

Scalars

`e(N)`	number of observations
`e(n`*j*`)`	number of observations for treatment level *j*
`e(N_clust)`	number of clusters
`e(k_eq)`	number of equations in `e(b)`
`e(k_levels)`	number of levels in treatment variable
`e(treated)`	level of treatment variable defined as treated
`e(control)`	level of treatment variable defined as control
`e(converged)`	`1` if converged, `0` otherwise

Macros

`e(cmd)`	`teffects`
`e(cmdline)`	command as typed
`e(depvar)`	name of outcome variable
`e(tvar)`	name of treatment variable
`e(subcmd)`	`aipw`
`e(tmodel)`	`logit`, `probit`, or `hetprobit`
`e(omodel)`	`linear`, `logit`, `probit`, `hetprobit`, `poisson`, `flogit`, `fprobit`, or `fhetprobit`
`e(stat)`	statistic estimated, `ate` or `pomeans`
`e(wtype)`	weight type
`e(wexp)`	weight expression
`e(title)`	title in estimation output
`e(clustvar)`	name of cluster variable
`e(tlevels)`	levels of treatment variable
`e(cme)`	`ml`, `nls`, or `wnls`
`e(vce)`	*vcetype* specified in `vce()`
`e(vcetype)`	title used to label Std. err.
`e(properties)`	`b V`
`e(estat_cmd)`	program used to implement `estat`
`e(predict)`	program used to implement `predict`
`e(marginsnotok)`	predictions disallowed by `margins`
`e(asbalanced)`	factor variables `fvset` as `asbalanced`
`e(asobserved)`	factor variables `fvset` as `asobserved`

Matrices

`e(b)`	coefficient vector
`e(V)`	variance–covariance matrix of the estimators

Functions

`e(sample)`	marks estimation sample

In addition to the above, the following is stored in `r()`:

Matrices

`r(table)`	matrix containing the coefficients with their standard errors, test statistics, *p*-values, and confidence intervals

Note that results stored in `r()` are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Methods and formulas

The methods and formulas presented here provide the technical details underlying the estimators implemented in `teffects ra`, `teffects ipw`, `teffects aipw`, and `teffects ipwra`. See *Methods and formulas* of [TE] **teffects nnmatch** for the methods and formulas used by `teffects nnmatch` and `teffects psmatch`.

Methods and formulas are presented under the following headings:

Parameters and notation

We begin by reviewing the effect parameters estimated by `teffects` and some essential notation.

The potential outcome that an individual would obtain if given treatment level $t \in \{0, 1, \ldots, q\}$ is y_t. Each y_t is a random variable, the realizations of which are y_{ti}. Throughout this document, i subscripts denote realizations of the corresponding, unsubscripted random variables.

The three parameters of interest are

1. the potential-outcome mean (POM) $\alpha_t = E(y_t)$;
2. the average treatment effect (ATE) $\tau_t = E(y_t - y_0)$; and
3. the average treatment effect on the treated (ATET) $\delta_t = E(y_t - y_0 | t = \widetilde{t})$.

The no-treatment level is 0.

The estimators implemented in `teffects` use three assumptions to justify the equations used for estimation and inference about the effect parameters of interest:

1. Conditional mean independence (CMI) allows us to estimate potential-outcome means from the observed outcomes in the sample.
2. Overlap ensures that we have data on each type of individual in each treatment level.
3. Independent observations ensure that the outcome and treatment for one individual has no effect on the outcome or treatment for any other individual.

`teffects ra` implements some regression-adjustment (RA) estimators; `teffects ipw` implements some inverse-probability-weighted (IPW) estimators; `teffects ipwra` implements some inverse-probability-weighted regression-adjustment (IPWRA) estimators; and `teffects aipw` implements some augmented inverse-probability-weighted (AIPW) estimators. All are implemented as estimating-equation (EE) estimators. The estimators are consistent and asymptotically normally distributed under the CMI, overlap, and independence assumptions.

Overview of EE estimators

EE estimators compute estimates by solving sample estimating equations. The sample estimating equations are the sample equivalents of population expectation equations.

Each EE estimator specifies a set of estimating equations for the effect parameters of interest and a set of estimating equations for the auxiliary parameters in the outcome model (OM) or the treatment model (TM). The next few sections provide tremendous detail about the estimating equations that define the RA, IPW, AIPW, and IPWRA estimators.

Ignoring the details for a moment, EE estimators solve systems of equations to compute estimates. A standard robust estimator is consistent for the variance of the estimator (VCE). All the details involve the equations specified by choices of estimator and functional forms for the OM or TM.

When used, the OM is a model for the conditional mean of the outcome variable. We let $\mu(\mathbf{x}, t, \boldsymbol{\beta}_t)$ denote a conditional-mean model for the outcome y conditional on covariates $\mathbf{x}$ and treatment level t. Mathematically, $E(y|\mathbf{x}, t) = \mu(\mathbf{x}, t, \boldsymbol{\beta}_t)$, where $\boldsymbol{\beta}_t$ are the parameters of the conditional-mean model given treatment level t. The table below provides details about the available functional forms.

Outcome model	Functional form for $\mu(\mathbf{x}, t, \boldsymbol{\beta}_t)$
`linear`	$\mathbf{x}\boldsymbol{\beta}_t$
`logit, flogit`	$\exp(\mathbf{x}\boldsymbol{\beta}_t)/\{1 + \exp(\mathbf{x}\boldsymbol{\beta}_t)\}$
`probit, fprobit`	$\Phi(\mathbf{x}\boldsymbol{\beta}_t)$
`poisson`	$\exp(\mathbf{x}\boldsymbol{\beta}_t)$
`hetprobit, fhetprobit`	$\Phi\{\dot{\mathbf{x}}\dot{\boldsymbol{\beta}}_t/\exp(\ddot{\mathbf{x}}\ddot{\boldsymbol{\beta}}_t)\}$

In the cases of `hetprobit` and `fhetprobit`, we use $\dot{\mathbf{x}}$ and $\dot{\boldsymbol{\beta}}_t$ to denote the variables and parameters in the index function, and we use $\ddot{\mathbf{x}}$ and $\ddot{\boldsymbol{\beta}}_t$ to denote the variables and parameters in the variance equation. We define $\mathbf{x}' = (\dot{\mathbf{x}}', \ddot{\mathbf{x}}')$ and $\boldsymbol{\beta}_t' = (\dot{\boldsymbol{\beta}}_t', \ddot{\boldsymbol{\beta}}_t')$.

We write the vector of parameters for the outcome model over all treatment levels as $\boldsymbol{\beta}' = (\boldsymbol{\beta}_0', \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_q')$.

Next we provide details about the estimating equations implied by each functional form choice.

When used, the TM is a model for the conditional probability of treatment. We let $p(\mathbf{z}, t, \boldsymbol{\gamma})$ denote the conditional probability model for the probability that a person receives treatment t, conditional on covariates $\mathbf{z}$. The table below provides details about the functional form options allowed in the case of a binary treatment.

Treatment model	Functional form for $p(\mathbf{z}, t, \boldsymbol{\gamma})$
`logit`	$\exp(\mathbf{z}\boldsymbol{\gamma})/\{1 + \exp(\mathbf{z}\boldsymbol{\gamma})\}$
`probit`	$\Phi(\mathbf{z}\boldsymbol{\gamma})$
`hetprobit`	$\Phi\{\dot{\mathbf{z}}\dot{\boldsymbol{\gamma}}/\exp(\ddot{\mathbf{z}}\ddot{\boldsymbol{\gamma}})\}$

In the case of `hetprobit`, we use $\dot{\mathbf{z}}$ and $\dot{\boldsymbol{\gamma}}$ to denote the variables and parameters in the index function, and we use $\ddot{\mathbf{z}}$ and $\ddot{\boldsymbol{\gamma}}$ to represent the variables and parameters in the variance equation. We define $\mathbf{z}' = (\dot{\mathbf{z}}', \ddot{\mathbf{z}}')$, and $\boldsymbol{\gamma}' = (\dot{\boldsymbol{\gamma}}', \ddot{\boldsymbol{\gamma}}')$.

In the multivalued-treatment case, $p(\mathbf{z}, t, \boldsymbol{\gamma})$ is specified as a multinomial logit with $p(\mathbf{z}, t, \boldsymbol{\gamma}) = \exp(\mathbf{z}\boldsymbol{\gamma}_t)/\{1 + \sum_{k=1}^{q} \exp(\mathbf{z}\boldsymbol{\gamma}_k)\}$ and $\boldsymbol{\gamma}' = (\boldsymbol{\gamma}_1', \boldsymbol{\gamma}_2', \ldots, \boldsymbol{\gamma}_q')$. (We present formulas for the case with treatment level 0 as the base with $\boldsymbol{\gamma}_0' = \mathbf{0}'$; see [R] **mlogit** for background.) In `teffects`, the `logit` option in the treatment-model specification means binary logit for the binary-treatment case and multinomial logit for the multivalued-treatment case: this simplifies the use of the command and makes statistical sense.

Below we provide details about the estimating equations implied by each functional form choice. The effect parameters of interest are

1. the POMs denoted by $\boldsymbol{\alpha}' = (\alpha_0, \alpha_1, \ldots, \alpha_q)$;
2. the ATEs denoted by $\boldsymbol{\tau}' = (\tau_1, \tau_2, \ldots, \tau_q)$; and
3. the ATETs denoted by $\boldsymbol{\delta}' = (\delta_1, \delta_2, \ldots, \delta_q)$.

We denote the effect parameters by ϑ and all the parameters in any particular case by $\boldsymbol{\theta}$. More formally, $\boldsymbol{\theta}$ is the concatenation of the effect parameters, the OM parameters, and the TM parameters; $\boldsymbol{\theta}' = (\vartheta', \boldsymbol{\beta}', \boldsymbol{\gamma}')$, where ϑ is $\boldsymbol{\alpha}$, $\boldsymbol{\tau}$, or $\boldsymbol{\delta}$, and $\boldsymbol{\beta}$ or $\boldsymbol{\gamma}$ may not be present, depending on the case at hand.

In the subsections below, we discuss estimators for the elements in $\boldsymbol{\theta}$ in detail and note how these elements change over the cases defined by effect parameters and estimators. The parameter vector $\boldsymbol{\theta}$ denotes all the parameters, no matter which particular case is under consideration.

The EE estimators described in this section are defined by a set of equations,

$$E\{\mathbf{s}(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta})\} = \mathbf{0}$$

where $\mathbf{s}(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta})$ is a vector of estimating functions. Note the notation: estimating equations equate the expected value of a vector of estimating functions to zero.

Because each of the estimating functions has mean zero, we can construct estimators that find the estimates $\widehat{\boldsymbol{\theta}}$ by solving a system of equations,

$$1/N \sum_{i}^{N} \mathbf{s}_i(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}}) = \mathbf{0}$$

where $\mathbf{s}_i(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})$ are the sample realizations of the estimating functions. In other words, the parameter estimates set the average of the realizations of each estimating function to zero. Almost all the details below involve specifying the sample realizations $\mathbf{s}_i(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})$.

Estimators that set the expected value of estimating functions to zero are known as estimating-equations (EE) estimators, M estimators, or Z estimators in the statistics literature and as generalized method of moments (GMM) estimators in the econometrics literature. See van der Vaart (1998, 41), Stefanski and Boos (2002), and Tsiatis (2006, sec. 3.2) for statistics; and see Wooldridge (2010, chap. 14), Cameron and Trivedi (2005, chap. 6), and Newey and McFadden (1994) for econometrics.

We refer to them as EE estimators because this name is closest to being independent of any discipline. The estimators are implemented using `gmm` because they are exactly identified generalized method of moments (GMM) estimators. When weights are specified by the user, they are applied to the estimating equations just as `gmm` applies user-specified weights.

Each estimator has a set of estimating equations for the effect parameters and either an OM or a TM, or both. The OM parameters or the TM parameters are auxiliary parameters used to estimate the effect parameters of interest.

Each set of parameters has its own set of sample estimating equations:

$1/N \sum_i \mathbf{s}_{e,i}(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}}) = \mathbf{0}$ are the sample estimating equations for the effect parameters. These equations determine the effect parameter estimates $\widehat{\vartheta}$ as functions of the data and the other estimated parameters.

$1/N \sum_i \mathbf{s}_{\mathrm{om},i}(\mathbf{x}_i, w_i, \widehat{\boldsymbol{\beta}}) = \mathbf{0}$ are the sample estimating equations for OM parameters that use the weights w_i, which are functions of the TM parameters.

$1/N \sum_i \mathbf{s}_{\mathrm{tm},i}(\mathbf{z}_i, \widehat{\boldsymbol{\gamma}}) = \mathbf{0}$ are the sample estimating equations for TM parameters.

The whole set of sample estimating functions is $\mathbf{s}_i(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})$ with

$$\mathbf{s}_i(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})' = (\mathbf{s}_{e,i}(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})', \mathbf{s}_{\mathrm{om},i}(\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}})', \mathbf{s}_{\mathrm{tm},i}(\mathbf{z}_i, \widehat{\boldsymbol{\gamma}})')$$

although not all the estimators have each of three components.

VCE for EE estimators

The Huber/White/robust sandwich estimator is consistent for the variance–covariance of the estimator (VCE). See van der Vaart (1998, 41), Stefanski and Boos (2002), and Tsiatis (2006, sec. 3.2) for statistics; and see Wooldridge (2010, chap. 14), Cameron and Trivedi (2005, chap. 6), and Newey and McFadden (1994) for econometrics.

The formula is

$$\widehat{\mathbf{V}} = (1/N)\overline{\mathbf{G}}\,\overline{\mathbf{S}}\,\overline{\mathbf{G}}'$$

where

$$\overline{\mathbf{G}} = \left\{(1/N)\sum_i \frac{\partial s_i(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})}{\partial \widehat{\boldsymbol{\theta}}}\right\}^{-1}$$

and

$$\overline{\mathbf{S}} = (1/N)\sum_i s_i(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}}) s_i(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})'$$

The matrix $\overline{\mathbf{G}}$ is not symmetric because our EE estimators come from stacking moment conditions instead of optimizing a single objective function. The implication is that the robust formula should always be used because, even under correct specification, the nonsymmetric $\overline{\mathbf{G}}$ and the symmetric $\overline{\mathbf{S}}$ converge to different matrices.

TM and OM estimating functions

Although the sample estimating functions for the effect parameters, the $\mathbf{s}_{e,i}(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})$, are estimator specific, the sample estimating functions for the TM parameters, the $\mathbf{s}_{\mathrm{tm},i}(\mathbf{z}_i, \widehat{\boldsymbol{\gamma}})$, and the sample estimating functions for the OM parameters, the $\mathbf{s}_{\mathrm{om},i}(\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}})'$, are used in multiple estimators. We provide details about the TM and the OM sample estimating functions here.

TM estimating functions

The sample estimating functions used to estimate the parameters of the TM $p(\mathbf{z}, t, \boldsymbol{\gamma})$ are the sample score equations from the quasimaximum likelihood (QML) estimator.

In the binary-treatment case, $p(\mathbf{z}, t, \boldsymbol{\gamma})$ may be logit, probit, or heteroskedastic probit. In the multivalued-treatment case, $p(\mathbf{z}, t, \boldsymbol{\gamma})$ is a multinomial logit. We now give formulas for the $\mathbf{s}_{\mathrm{tm},i}(\mathbf{z}_i, \widehat{\boldsymbol{\gamma}})$ for each case.

logit and probit

In the `logit` and `probit` cases,

$$\mathbf{s}_{\mathrm{tm},i}(\mathbf{z}_i, \widehat{\boldsymbol{\gamma}}) = \left[\frac{g(\mathbf{z}_i\widehat{\boldsymbol{\gamma}}')\left\{t_i - G(\mathbf{z}_i\widehat{\boldsymbol{\gamma}}')\right\}}{G(\mathbf{z}_i\widehat{\boldsymbol{\gamma}}')\left\{1 - G(\mathbf{z}_i\widehat{\boldsymbol{\gamma}}')\right\}}\right]\mathbf{z}_i$$

where $G(z)$ is the logistic cumulative distribution function for the logit, $G(z)$ is the normal cumulative distribution function for the probit, and $g(\cdot) = \{\partial G(z)\}/(\partial z)$ is the corresponding density function.

hetprobit

In the `hetprobit` case, there are two sets of sample score equations, $\mathbf{s}_{\mathrm{tm},1,i}(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}})$ and $\mathbf{s}_{\mathrm{tm},2,i}(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}})$:

$$\mathbf{s}_{\mathrm{tm},1,i}(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}) = \left(\frac{\phi\left\{q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right)\right\}\left[t_i-\Phi\left\{q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right)\right\}\right]}{\Phi\left\{q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right)\right\}\left[1-\Phi\left\{q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right)\right\}\right]\exp(\ddot{\mathbf{z}}_i\widehat{\ddot{\boldsymbol{\gamma}}}')}\right)\dot{\mathbf{z}}_i'$$

and

$$\mathbf{s}_{\mathrm{tm},2,i}(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}) = \left(\frac{\phi\left\{q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right)\right\}\dot{\mathbf{z}}_i\widehat{\dot{\boldsymbol{\gamma}}}'\left[\Phi\left\{q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right)\right\}-t_i\right]}{\Phi\left\{q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right)\right\}\left[1-\Phi\left\{q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right)\right\}\right]\exp(\ddot{\mathbf{z}}_i\widehat{\ddot{\boldsymbol{\gamma}}}')}\right)\ddot{\mathbf{z}}_i'$$

where $\phi(\cdot)$ is the standard normal density function, and $q\left(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}\right) = \left(\dot{\mathbf{z}}_i\widehat{\dot{\boldsymbol{\gamma}}}'/\exp(\ddot{\mathbf{z}}_i\widehat{\ddot{\boldsymbol{\gamma}}}')\right)$.

mlogit

In the `mlogit` case, $p(\mathbf{z},t,\boldsymbol{\gamma}) = \exp(\mathbf{z}\boldsymbol{\gamma}_t)/\left\{1+\sum_{k=1}^{q}\exp(\mathbf{z}\boldsymbol{\gamma}_k)\right\}$. We present formulas for the case with treatment level 0 as the base with $\boldsymbol{\gamma}_0' = \mathbf{0}'$; see [R] **mlogit** for background.

There are q vectors of sample estimating functions for the `mlogit` case, $\mathbf{s}_{\mathrm{tm},k,i}(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}})$ for each $k\in\{1,\ldots,q\}$, 1 for each vector $\widehat{\boldsymbol{\gamma}}_k$, $k\in\{1,\ldots,q\}$. These sample estimating functions are

$$\mathbf{s}_{\mathrm{tm},k,i}(\mathbf{z}_i,\widehat{\boldsymbol{\gamma}}) = \begin{cases}\{1-p(\mathbf{z}_i,k,\widehat{\boldsymbol{\gamma}})\}\mathbf{z}_i' & t_i=k\\ -p(\mathbf{z}_i,k,\widehat{\boldsymbol{\gamma}})\mathbf{z}_i' & \text{otherwise}\end{cases}$$

OM estimating functions

The parameters of the OM $\mu(\mathbf{x},t,\boldsymbol{\beta}_t)$ are estimated either by weighted QML or by weighted nonlinear least squares. The estimating functions used to estimate the parameters of the OM are either the score equations from the weighted QML estimator or the moment conditions for the weighted nonlinear least-squares estimator.

The estimating functions for the OM parameters in $\mu(\mathbf{x},t,\boldsymbol{\beta}_t)$ vary over the models for the conditional mean because $\mu(\mathbf{x},t,\boldsymbol{\beta}_t)$ may be linear, logit, probit, heteroskedastic probit, or poisson.

Let N_t be the number of observations in treatment level t, and let $t_i(t)=1$ if $t_i=t$, with $t_i(t)=0$ if $t_i\neq t$.

There are two sets of sample estimating functions for the OM parameters with weights $w_i(t)$:

1. $\mathbf{s}_{\mathrm{ml,om},i}\{\mathbf{x}_i,w_i(t),\widehat{\boldsymbol{\beta}}_t)\}$ are the sample estimating functions for the weighted QML estimator.
2. $\mathbf{s}_{\mathrm{nls,om},i}\{\mathbf{x}_i,w_i(t),\widehat{\boldsymbol{\beta}}_t)\}$ are the sample estimating functions for the weighted nonlinear least-squares estimator.

OM QML

Here are the formulas for the $\mathbf{s}_{\mathrm{ml,om},i}\{\mathbf{x}_i,w_i(t),\widehat{\boldsymbol{\beta}}_t\}$ for each functional form choice.

linear

In the linear case,

$$\mathbf{s}_{\mathrm{ml,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)(y_i - \mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\mathbf{x}_i'$$

logit, flogit, probit, and fprobit

In the `logit`, `flogit`, `probit`, and `fprobit` cases,

$$\mathbf{s}_{\mathrm{ml,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)\left[\frac{g(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\left\{y_i - G(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\right\}}{G(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\left\{1 - G(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\right\}}\right]\mathbf{x}_i$$

where $G(z)$ is the logistic cumulative distribution function for the logit and flogit, $G(z)$ is the normal cumulative distribution function for the probit and fprobit, and $g(\cdot) = \{\partial G(z)\}/(\partial z)$ is the corresponding density function.

hetprobit and fhetprobit

In the `hetprobit` and `fhetprobit` cases, there are two sets of sample score equations, $\mathbf{s}_{\mathrm{ml,om},1,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}$ and $\mathbf{s}_{\mathrm{ml,om},2,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}$:

$$\mathbf{s}_{\mathrm{ml,om},1,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)\left(\frac{\phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}\left[y_i - \Phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}\right]}{\Phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}\left[1 - \Phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}\right]\exp(\ddot{\mathbf{x}}_i\widehat{\ddot{\boldsymbol{\beta}}}_t')}\right)\dot{\mathbf{x}}_i'$$

and

$$\mathbf{s}_{\mathrm{ml,om},2,i}(\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t) = w_i(t)t_i(t)\left(\frac{\phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}\dot{\mathbf{x}}_i\widehat{\dot{\boldsymbol{\beta}}}_t'\left[\Phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\} - y_i\right]}{\Phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}\left[1 - \Phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}\right]\exp(\ddot{\mathbf{x}}_i\widehat{\ddot{\boldsymbol{\beta}}}_t')}\right)\ddot{\mathbf{x}}_i'$$

where $\phi(\cdot)$ is the standard normal density function, $\mathbf{s}_{\mathrm{ml,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}' = [\mathbf{s}_{\mathrm{ml,om},1,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}', \mathbf{s}_{\mathrm{ml,om},2,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}']$, and $q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right) = \left(\dot{\mathbf{x}}_i\widehat{\dot{\boldsymbol{\beta}}}_t'/\exp(\ddot{\mathbf{x}}_i\widehat{\ddot{\boldsymbol{\beta}}}_t')\right)$.

poisson

In the `poisson` case,

$$\mathbf{s}_{\mathrm{ml,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)\{y_i - \exp(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\}\mathbf{x}_i'$$

OM WNL

Here are the formulas for the $\mathbf{s}_{\mathrm{nls,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t)\}$ for each functional form choice.

linear

In the linear case,

$$\mathbf{s}_{\text{nls,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)(y_i - \mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\mathbf{x}_i'$$

logit, flogit, probit, and fprobit

In the `logit`, `flogit`, `probit`, and `fprobit` cases,

$$\mathbf{s}_{\text{nls,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)\left[g(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\left\{y_i - G(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\right\}\right]\mathbf{x}_i$$

where $G(z)$ is the logistic cumulative distribution function for the logit and flogit, $G(z)$ is the normal cumulative distribution function for the probit and fprobit, and $g(\cdot) = \{\partial G(z)\}/(\partial z)$ is the corresponding density function.

hetprobit and fhetprobit

In the `hetprobit` and `fhetprobit` cases, there are two sets of sample score equations, $\mathbf{s}_{\text{nls,om},1,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}$ and $\mathbf{s}_{\text{nls,om},2,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}$:

$$\mathbf{s}_{\text{nls,om},1,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)\left(\frac{\phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}}{\exp(\ddot{\mathbf{x}}_i\widehat{\ddot{\boldsymbol{\beta}}}_t')}\left[y_i - \Phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}\right]\right)\dot{\mathbf{x}}_i'$$

and

$$\mathbf{s}_{\text{nls,om},2,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)\left(\frac{\phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\}}{\exp(\ddot{\mathbf{x}}_i\widehat{\ddot{\boldsymbol{\beta}}}_t')}\dot{\mathbf{x}}_i\widehat{\dot{\boldsymbol{\beta}}}_t'\left[\Phi\left\{q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right)\right\} - y_i\right]\right)\ddot{\mathbf{x}}_i'$$

where $\phi(\cdot)$ is the standard normal density function, $\mathbf{s}_{\text{nls,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}' = [\mathbf{s}_{\text{nls,om},1,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}', \mathbf{s}_{\text{nls,om},2,i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\}']$, and $q\left(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t\right) = \left(\dot{\mathbf{x}}_i\widehat{\dot{\boldsymbol{\beta}}}_t'/\exp(\ddot{\mathbf{x}}_i\widehat{\ddot{\boldsymbol{\beta}}}_t')\right)$.

poisson

In the `poisson` case,

$$\mathbf{s}_{\text{nls,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}_t\} = w_i(t)t_i(t)\{y_i - \exp(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\}\exp(\mathbf{x}_i\widehat{\boldsymbol{\beta}}_t')\mathbf{x}_i'$$

Effect estimating functions

We now describe the sample estimating functions for the effect parameters, which vary over estimator and effect parameter.

RA estimators

RA estimators estimate the effect parameters using means of the observation-level predictions of the conditional means of the outcomes. There is no model for the conditional probability of treatment.

The RA estimators use unweighted QML estimators to estimate the parameters of the conditional mean model. In other words, the RA estimators use the sample estimating functions $\mathbf{s}_{\mathrm{ml,om},i}(\mathbf{x}_i, 1, \widehat{\boldsymbol{\beta}})$, given above.

For the RA estimators, the vector of sample estimating functions is the concatenation of the sample estimating functions for the effect parameters with the sample estimating functions for the OM parameters. Algebraically,

$$\mathbf{s}_{\mathrm{ra},i}(\mathbf{x}_i, \widehat{\boldsymbol{\theta}})' = \mathbf{s}_{\mathrm{ra},e,i}(\mathbf{x}_i, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\beta}})', \mathbf{s}_{\mathrm{ml,om},i}(\mathbf{x}_i, 1, \widehat{\boldsymbol{\beta}})'$$

The estimating functions $\mathbf{s}_{\mathrm{ra},e,i}(\mathbf{x}_i, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\beta}})'$ vary over the effect parameter.

RA for POM

The RA estimators for the POM parameters estimate $\boldsymbol{\theta}' = (\boldsymbol{\alpha}', \boldsymbol{\beta}')$ using two types of estimating equations: 1) those for the POM parameters $\boldsymbol{\alpha}$, and 2) those for the conditional-mean model parameters $\boldsymbol{\beta}_t$ in $\mu(\mathbf{x}, t, \boldsymbol{\beta}_t)$.

The sample estimating functions for the $\widehat{\boldsymbol{\beta}}_t$ are given in *OM estimating functions* above.

The elements of $\mathbf{s}_{\mathrm{ra},e,i}(\mathbf{x}_i, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}})$ for the POM parameters $\boldsymbol{\alpha}$ are given by

$$\mu(\mathbf{x}_i, t, \widehat{\boldsymbol{\beta}}_t) - \widehat{\alpha}_t \tag{RAPOM}$$

RA for ATE

The RA estimators for the ATE parameters estimate $\boldsymbol{\theta}' = (\boldsymbol{\tau}', \boldsymbol{\beta}')$ using two types of estimating equations: 1) those for the ATE parameters $\boldsymbol{\tau}$, and 2) those for the OM parameters $\boldsymbol{\beta}_t$ in $\mu(\mathbf{x}, t, \boldsymbol{\beta}_t)$.

The sample estimating functions that determine the $\widehat{\boldsymbol{\beta}}_t$ are given in *OM estimating functions* with $w_i(t) = 1$.

The elements of $\mathbf{s}_{\mathrm{ra},e,i}(\mathbf{x}_i, \widehat{\boldsymbol{\tau}}, \widehat{\boldsymbol{\beta}})$ for the ATE parameters $\boldsymbol{\tau}$ are given by

$$\mu(\mathbf{x}_i, t, \widehat{\boldsymbol{\beta}}_t) - \mu(\mathbf{x}_i, 0, \widehat{\boldsymbol{\beta}}_t) - \widehat{\tau}_t \tag{RAATE}$$

RA for ATET

The RA estimators for the ATET parameters estimate $\boldsymbol{\theta}' = (\boldsymbol{\delta}', \boldsymbol{\beta}')$ using two types of estimating equations: 1) those for the ATET parameters $\boldsymbol{\delta}$, and 2) those for the OM parameters $\boldsymbol{\beta}_t$ in $\mu(\mathbf{x}, t, \boldsymbol{\beta}_t)$.

The sample estimating functions that determine the $\widehat{\boldsymbol{\beta}}_t$ are given in *OM estimating functions* above with $w_i(t) = 1$.

The elements of $\mathbf{s}_{\mathrm{ra},e,i}(\mathbf{x}_i, \widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\beta}})$ for the ATET parameters $\boldsymbol{\delta}$ are given by

$$Nt_i(\widetilde{t})/N_{\widetilde{t}}\left\{\mu(\mathbf{x}_i, t, \widehat{\boldsymbol{\beta}}_t) - \mu(\mathbf{x}_i, 0, \widehat{\boldsymbol{\beta}}_t) - \widehat{\delta}_t\right\} \tag{RAATET}$$

IPW estimators

IPW estimators estimate the effect parameters using means of the observed outcomes weighted by the inverse probability of treatment. There is no outcome model. The IPW estimators use QML estimators to estimate the parameters of the conditional probability model.

The vector of estimating functions is the concatenation of the estimating functions for the effect parameters with the estimating functions for the conditional-probability parameters. The sample estimating functions used by the IPW estimators are

$$\mathbf{s}_{\mathrm{ipw},i}(\mathbf{x}_i, \widehat{\boldsymbol{\theta}})' = \mathbf{s}_{\mathrm{ipw},e,i}(\mathbf{x}_i, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\gamma}})', \mathbf{s}_{\mathrm{tm},i}(\mathbf{z}_i, 1, \widehat{\boldsymbol{\gamma}})'$$

The estimating functions $\mathbf{s}_{\mathrm{ipw},e,i}(\mathbf{z}_i, \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\gamma}})'$ vary over the effect parameter.

All the IPW estimators use normalized inverse-probability weights. These weights are not related to the weights $w_i(t)$ that were used in the OM equations. The functional form for the normalized inverse-probability weights varies over the effect parameters POM, ATE, and ATET.

The POM and ATE estimators use normalized inverse-probability weights. The unnormalized weights for individual i and treatment level t are $d_i(t) = t_i(t)/p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})$, and the normalized weights are $\overline{d}_i(t) = N_t d_i(t)/\sum_i^N d_i(t)$.

The ATET estimator uses normalized treatment-adjusted inverse-probability weights. The treatment-adjusted inverse-probability weights adjust the inverse-probability weights for the probability of getting the conditional treatment $\widetilde{t}$. The unnormalized weights are $f_i = p(\mathbf{z}_i, \widetilde{t}, \widehat{\boldsymbol{\gamma}})/p(\mathbf{z}_i, t_i, \widehat{\boldsymbol{\gamma}})$, and the normalized weights are $\overline{f}_i = N f_i/\sum_i^N f_i$.

The IPW effect estimators are weighted cell averages. The sample estimating functions used in POM estimators are the sample estimating functions from weighted OLS regression on treatment-cell indicators. The POM estimators use a full set of $q+1$ of treatment indicator variables $\mathbf{a}_i$. (The ith observation on the kth variable in $\mathbf{a}_i$ is 1 if i received treatment $k-1$ and 0 otherwise, for $k \in \{1, 2, \ldots, q+1\}$.)

The sample estimating functions used in the ATE and the ATET estimators are the sample estimating functions from weighted OLS regression on treatment-cell indicators, excluding the indicator for the control level and including a constant term. The variables $\widetilde{\mathbf{a}}_i$ used in the ATE and ATET sample estimating functions include q of treatment indicator variables and a variable that is always 1. (The first q variables in $\widetilde{\mathbf{a}}_i$ are treatment indicators: the ith observation on the kth variable in $\widetilde{\mathbf{a}}_i$ is 1 if i received treatment k and 0 otherwise, for $k \in \{1, 2, \ldots, q\}$. The $(q+1)$th variable is always 1.) This definition of $\widetilde{\mathbf{a}}_i$ sets the last treatment level to be the control; renaming the treatments handles the more general case allowed for by `teffects`.

Having defined $\mathbf{a}_i$ and $\widetilde{\mathbf{a}}_i$, we can give the sample estimating functions that the IPW estimators use for the effects parameters.

IPW for POM

We begin with the IPW estimators for the potential-outcome means. In this case, $\boldsymbol{\theta}' = (\boldsymbol{\alpha}', \boldsymbol{\gamma}')$.

The sample estimating functions for the $\widehat{\boldsymbol{\gamma}}$ are given in *TM estimating functions* above.

The sample estimating functions for $\widehat{\boldsymbol{\alpha}}$ are given by

$$\mathbf{s}_{\mathrm{ipw},e,i,t}(\mathbf{z}_i, \widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\gamma}})' = \overline{d}_i(t)(y_i - \mathbf{a}_i\widehat{\boldsymbol{\alpha}})\mathbf{a}_i' \qquad \text{(IPWPOM)}$$

IPW for ATE

The full parameter vector for the IPW estimators for the ATE is $\boldsymbol{\theta}' = (\boldsymbol{\tau}', \boldsymbol{\gamma}')$.

The sample estimating functions for the $\widehat{\boldsymbol{\gamma}}$ are given in *TM estimating functions* above.

The sample estimating functions for $\widehat{\boldsymbol{\tau}}$ are given by

$$\mathbf{s}_{\mathrm{ipw},e,i,t}(\mathbf{z}_i, \widehat{\boldsymbol{\tau}}, \widehat{\boldsymbol{\gamma}})' = \overline{d}_i(t)(y_i - \widetilde{\mathbf{a}}_i\widehat{\boldsymbol{\tau}})\widetilde{\mathbf{a}}_i' \tag{IPWATE}$$

IPW for ATET

The full parameter vector for the IPW estimators for the ATET is $\boldsymbol{\theta}' = (\boldsymbol{\delta}', \boldsymbol{\gamma}')$.

The sample estimating functions for the $\widehat{\boldsymbol{\gamma}}$ are given in *TM estimating functions* above.

The sample estimating functions for $\widehat{\boldsymbol{\delta}}$ are given by

$$\mathbf{s}_{\mathrm{ipw},e,i,t}(\mathbf{z}_i, \widehat{\boldsymbol{\delta}}, \widehat{\boldsymbol{\gamma}})' = \overline{f}_i(t)(y_i - \widetilde{\mathbf{a}}_i\widehat{\boldsymbol{\delta}})\widetilde{\mathbf{a}}_i' \tag{IPWATET}$$

AIPW estimators

This section documents the sample estimating functions used by the augmented inverse-probability-weighted (AIPW) estimators implemented in `teffects`. These AIPW estimators are efficient-influence-function estimators as discussed in [TE] **teffects intro** and [TE] **teffects intro advanced**. The `teffects` implementation was primarily inspired by Cattaneo, Drukker, and Holland (2013), which was based on Cattaneo (2010). Tan (2010) was influential by identifying the implemented weighted nonlinear least-squares estimator as having relatively good properties when both the conditional mean function and the conditional probability function are misspecified.

The AIPW estimating functions for the treatment parameters include terms from a conditional probability model and from a conditional mean model for the outcome.

The sample-estimation-equations vector has three parts for the AIPW estimators:

$$\mathbf{s}_{\mathrm{aipw},i}(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})' = [\mathbf{s}_{\mathrm{aipw},e,i}(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})', \mathbf{s}_{\mathrm{aipw,tm},i}(\mathbf{z}_i, \widehat{\boldsymbol{\gamma}})', \mathbf{s}_{\mathrm{aipw,om},i}\{\mathbf{x}_i, w_i(t), \widehat{\boldsymbol{\beta}}\}']$$

The sample estimating functions for the parameters of the TM are the $\mathbf{s}_{\mathrm{tm},i}(\mathbf{z}_i, \widehat{\boldsymbol{\gamma}})$ given in *TM estimating functions* above.

`teffects aipw` implements three AIPW estimators: the standard AIPW estimator, the NLS AIPW estimator, and the WNLS AIPW estimator.

The three AIPW estimators differ in how they estimate the parameters of the OM.

By default, `teffects aipw` uses the standard AIPW estimator that estimates the parameters of the OM by the unweighted ML estimator, whose sample estimating functions, $\mathbf{s}_{\mathrm{ml,om},i}(\mathbf{x}_i, 1, \widehat{\boldsymbol{\beta}})$, are given in *OM estimating functions* above.

When the `nls` option is specified, `teffects aipw` uses the NLS AIPW estimator that estimates the parameters of the OM by the unweighted NLS estimator, whose sample estimating functions, $\mathbf{s}_{\mathrm{nls,om},i}(\mathbf{x}_i, 1, \widehat{\boldsymbol{\beta}})$, are given in *OM estimating functions* above.

When the `wnls` option is specified, `teffects aipw` uses the WNLS AIPW estimator that estimates the parameters of the OM by the WNLS estimator, whose sample estimating functions, $\mathbf{s}_{\text{nls,om},i}\{\mathbf{x}_i, \widetilde{w}_i(t), \widehat{\boldsymbol{\beta}}\}$, are given in *OM estimating functions* above. The weights for person i in treatment level t are

$$\widetilde{w}_i(t) = \frac{t_i(t)}{p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})}\left\{\frac{t_i(t)}{p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})} - 1\right\} \tag{WNLSW}$$

Now we discuss the sample estimating functions for the effect parameters, the $\mathbf{s}_{e,i}(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})$.

AIPW for POM

We begin with the AIPW estimators for the potential-outcome means. In this case, $\boldsymbol{\theta}' = (\boldsymbol{\alpha}', \boldsymbol{\gamma}', \boldsymbol{\beta}')$, and the elements of $\mathbf{s}_{\text{aipw},e,i}(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})$ are given by

$$\frac{t_i(t)}{p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})}y_i - \mu(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t)\left\{\frac{t_i(t)}{p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})} - 1\right\} - \alpha_t$$

AIPW for ATE

The AIPW estimators for the ATE estimate $\boldsymbol{\theta}' = (\boldsymbol{\tau}', \boldsymbol{\gamma}', \boldsymbol{\beta}')$, and the elements of $\mathbf{s}_{\text{aipw},e,i}(\mathbf{x}_i, \mathbf{z}_i, \widehat{\boldsymbol{\theta}})$ are given by

$$\frac{t_i(t)}{p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})}y_i - \mu(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t)\left\{\frac{t_i(t)}{p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})} - 1\right\} - \tau_0 \text{ if } t = 0$$

$$\frac{t_i(t)}{p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})}y_i - \mu(\mathbf{x}_i, \widehat{\boldsymbol{\beta}}_t)\left\{\frac{t_i(t)}{p(\mathbf{z}_i, t, \widehat{\boldsymbol{\gamma}})} - 1\right\} - \tau_t - \tau_0 \text{ if } t > 0$$

IPWRA estimators

The IPWRA estimators combine inverse-probability weighting (IPW) with regression-adjustment (RA) methods. The sample estimating functions for IPWRA include sample estimating functions from both RA and IPW. The vector of sample estimating functions is

$$\mathbf{s}_{\text{ipwra},i}(\mathbf{x}_i, \widehat{\boldsymbol{\theta}})' = \mathbf{s}_{\text{ra},e,i}(\mathbf{x}_i, \widehat{\vartheta}, \widehat{\boldsymbol{\beta}})', \mathbf{s}_{\text{ml,om},i}\{\mathbf{x}_i, w_i(j), \widehat{\boldsymbol{\beta}}\}', \mathbf{s}_{\text{tm},i}(\mathbf{z}_i, \widehat{\boldsymbol{\gamma}})'$$

where $\widehat{\boldsymbol{\theta}}' = (\widehat{\vartheta}', \widehat{\boldsymbol{\beta}}', \widehat{\boldsymbol{\gamma}}')$, $\widehat{\vartheta} = \widehat{\boldsymbol{\alpha}}$ for POM, $\widehat{\vartheta} = \widehat{\boldsymbol{\tau}}_t$ for ATE, and $\widehat{\vartheta} = \widehat{\boldsymbol{\delta}}_t$ for ATET. The sample estimating functions, $\mathbf{s}_{\text{ra},e,i}(\mathbf{x}_i, \widehat{\vartheta}, \widehat{\boldsymbol{\beta}})$, for POM, ATE, and ATET are given in equations (RAPOM), (RAATE), and (RAATET). For the OM sample estimating functions, $\mathbf{s}_{\text{ml,om},i}(\cdot)$, we replace the RA unity weights, $w_i(t) = 1$, with $d_i(j)$ for POM or ATE and $\overline{f}_i$ for ATET, the normalized inverse-probability weights described in *IPW estimators* above. The specific form of the OM sample estimating functions are given in *OM estimating functions* above. The TM sample estimating functions are given in *TM estimating functions* above.

References

Cameron, A. C., and P. K. Trivedi. 2005. *Microeconometrics: Methods and Applications*. New York: Cambridge University Press.

Cattaneo, M. D. 2010. Efficient semiparametric estimation of multi-valued treatment effects under ignorability. *Journal of Econometrics* 155: 138–154. https://doi.org/10.1016/j.jeconom.2009.09.023.

Cattaneo, M. D., D. M. Drukker, and A. D. Holland. 2013. Estimation of multivalued treatment effects under conditional independence. *Stata Journal* 13: 407–450.

Huber, C. 2015. Introduction to treatment effects in Stata: Part 1. *The Stata Blog: Not Elsewhere Classified*. http://blog.stata.com/2015/07/07/introduction-to-treatment-effects-in-stata-part-1/.

Newey, W. K., and D. L. McFadden. 1994. Large sample estimation and hypothesis testing. In Vol. 4 of *Handbook of Econometrics*, ed. R. F. Engle and D. L. McFadden, 2111–2245. Amsterdam: Elsevier. https://doi.org/10.1016/S1573-4412(05)80005-4.

Stefanski, L. A., and D. D. Boos. 2002. The calculus of M-estimation. *American Statistician* 56: 29–38. https://doi.org/10.1198/000313002753631330.

Tan, Z. 2010. Bounded, efficient and doubly robust estimation with inverse weighting. *Biometrika* 97: 661–682. https://doi.org/10.1093/biomet/asq035.

Tsiatis, A. A. 2006. *Semiparametric Theory and Missing Data*. New York: Springer.

van der Vaart, A. W. 1998. *Asymptotic Statistics*. Cambridge: Cambridge University Press.

Wooldridge, J. M. 2010. *Econometric Analysis of Cross Section and Panel Data*. 2nd ed. Cambridge, MA: MIT Press.

Also see

[TE] **teffects postestimation** — Postestimation tools for teffects

[TE] **teffects** — Treatment-effects estimation for observational data

[TE] **teffects ipwra** — Inverse-probability-weighted regression adjustment

[U] **20 Estimation and postestimation commands**