

Package ‘bife’

May 25, 2019

Type Package

Title Binary Choice Models with Fixed Effects

Version 0.6

Description Estimates fixed effects binary choice models (logit and probit) with potentially many individual fixed effects and computes average partial effects. Incidental parameter bias can be reduced with an asymptotic bias-correction proposed by Fernandez-Val (2009) <doi:10.1016/j.jeconom.2009.02.007>.

License GPL (>= 2)

Depends R (>= 3.1.0)

Imports data.table, Formula, Rcpp, stats

LinkingTo Rcpp, RcppArmadillo

URL <https://github.com/amrei-stammann/bife>

BugReports <https://github.com/amrei-stammann/bife/issues>

RoxygenNote 6.1.1

LazyData true

Suggests alpaca, knitr

VignetteBuilder knitr

NeedsCompilation yes

Author Amrei Stammann [aut, cre],
Daniel Czarnowske [aut] (<<https://orcid.org/0000-0002-0030-929X>>),
Florian Heiss [aut],
Daniel McFadden [ctb]

Maintainer Amrei Stammann <amrei.stammann@hhu.de>

Repository CRAN

Date/Publication 2019-05-24 23:50:17 UTC

R topics documented:

bias_corr	2
bife	3
bife_control	5
coef.bife	6
coef.bifeAPEs	7
fitted.bife	7
get_APEs	8
predict.bife	9
print.bife	10
print.bifeAPEs	11
print.summary.bife	12
print.summary.bifeAPEs	12
psid	13
summary.bife	14
summary.bifeAPEs	14
vcov.bife	15
vcov.bifeAPEs	16
Index	17

bias_corr

Asymptotic bias-correction for binary choice Models with fixed effects

Description

`bias_corr` is a post-estimation routine that can be used to substantially reduce the incidental parameter bias problem (Neyman and Scott (1948)) present in non-linear fixed effects models (see Fernandez-Val and Weidner (2018) for an overview). The command applies the analytical bias-correction derived by Fernandez-Val (2009) to obtain bias-corrected estimates of the structural parameters.

Remark: Fernandez-Val (2009) further refined the bias-correction of Hahn and Newey (2004). The correction is now also applicable to dynamic models.

Usage

```
bias_corr(object, L = 0L)
```

Arguments

object	an object of class "bife".
L	unsigned integer indicating a bandwidth for the estimation of spectral densities proposed by Hahn and Kuersteiner (2011). Default is zero, which should be used if all regressors are assumed to be strictly exogenous. In the presence of weakly exogenous or predetermined regressors, Fernandez-Val and Weidner (2018) suggest to choose a bandwidth not higher than four.

Value

The function `bias_corr` returns a named list of class "bife".

References

Fernandez-Val, I. (2009). "Fixed effects estimation of structural parameters and marginal effects in panel probit models". *Journal of Econometrics* 150(1), 71-85.

Fernandez-Val, I. and Weidner, M. (2018). "Fixed effects estimation of large-t panel data models". *Annual Review of Economics*, 10, 109-138.

Hahn, J. and Kuersteiner, G. (2011). "Bias reduction for dynamic nonlinear panel models with fixed effects". *Econometric Theory*, 27(6), 1152-1191.

Hahn, J. and Newey, W. (2004). "Jackknife and analytical bias reduction for nonlinear panel models". *Econometrica* 72(4), 1295-1319.

Neyman, J. and Scott, E. L. (1948). "Consistent estimates based on partially consistent observations". *Econometrica*, 16(1), 1-32.

Stammann, A., Heiss, F., and and McFadden, D. (2016). "Estimating Fixed Effects Logit Models with Large Panel Data". Working paper.

See Also

[bife](#)

Examples

```
# Load 'psid' dataset
library(bife)
dataset <- psid

# Fit a static logit model
mod <- bife(LFP ~ I(AGE^2) + log(INCH) + KID1 + KID2 + KID3 + factor(TIME) | ID, dataset)
summary(mod)

# Apply analytical bias-correction
mod_bc <- bias_corr(mod)
summary(mod_bc)
```

Description

`bife` can be used to fit fixed effects binary choice models (logit and probit) based on an unconditional maximum likelihood approach. It is tailored for the fast estimation of binary choice models with potentially many individual fixed effects. The routine is based on a special pseudo demeaning algorithm derived by Stammann, Heiss, and McFadden (2016). The estimates obtained are identical to the ones of `glm`, but the computation time of `bife` is much lower.

Remark: The term fixed effect is used in econometrician's sense of having a full set of individual specific intercepts. All other parameters in the model are referred to as structural parameters.

Usage

```
bife(formula, data = list(), model = c("logit", "probit"),
     beta_start = NULL, control = list(), bias_corr = NULL,
     tol_demeaning = NULL, iter_demeaning = NULL, tol_offset = NULL,
     iter_offset = NULL)
```

Arguments

<code>formula</code>	an object of class "formula" (or one that can be coerced to that class): a symbolic description of the model to be fitted. <code>formula</code> must be of type $y x id$ where the <code>id</code> refers to an individual identifier (fixed effect category).
<code>data</code>	an object of class "data.frame" containing the variables in the model.
<code>model</code>	the description of the error distribution and link function to be used in the model. For <code>bife</code> this has to be a character string naming the model function. Default is "logit".
<code>beta_start</code>	an optional vector of starting values used for the structural parameters in the optimization algorithm. Default is zero for all structural parameters.
<code>control</code>	a named list of parameters for controlling the fitting process. See <code>bife_control</code> for details.
<code>bias_corr</code>	deprecated; see <code>bias_corr</code> .
<code>tol_demeaning</code> , <code>iter_demeaning</code> , <code>tol_offset</code> , <code>iter_offset</code>	deprecated; see <code>bife_control</code> .

Details

`bife` drops all observations of cross-sectional units (individuals) with non-varying response. This can be done because these observations do not contribute to the identification of the structural parameters (perfect classification).

If `bife` does not converge this is usually a sign of linear dependence between one or more regressors and the fixed effects. In this case, you should carefully inspect your model specification.

Value

The function `bife` returns a named list of class "bife".

References

Stammann, A., Heiss, F., and and McFadden, D. (2016). "Estimating Fixed Effects Logit Models with Large Panel Data". Working paper.

Examples

```
# Load 'psid' dataset
library(bife)
dataset <- psid

# Fit a static logit model
mod <- bife(LFP ~ I(AGE^2) + log(INCH) + KID1 + KID2 + KID3 + factor(TIME) | ID, dataset)
summary(mod)
```

bife_control	<i>Set bife Control Parameters</i>
--------------	------------------------------------

Description

Set and change parameters used for fitting [bife](#).

Usage

```
bife_control(dev_tol = 1e-08, rho_tol = 1e-04, conv_tol = 1e-06,
            iter_max = 100L, trace = FALSE)
```

Arguments

dev_tol	tolerance level for the first stopping condition of the maximization routine. The stopping condition is based on the relative change of the deviance in iteration r and can be expressed as follows: $(dev_{r-1} - dev_r)/(0.1 + dev_r) < tol$. Default is $1.0e-08$.
rho_tol	tolerance level for the stephalving in the maximization routine. Stephalving only takes place if the deviance in iteration r is larger than the one of the previous iteration. If this is the case, $\ \beta_r - \beta_{r-1}\ _2$ is halved until the deviance is less or numerically equal compared to the deviance of the previous iteration. Stephalving fails if the the following condition holds: $\rho < tol$, where ρ is the stepcorrection factor. If stephalving fails the maximization routine is canceled. Default is $1.0e-04$.
conv_tol	tolerance level that accounts for rounding errors inside the stephalving routine when comparing the deviance with the one of the previous iteration. Default is $1.0e-06$.
iter_max	unsigned integer indicating the maximum number of iterations in the maximization routine. Default is 100L.
trace	logical indicating if output should be produced in each iteration. Default is FALSE.

Value

The function `bife_control` returns a named list of control parameters.

See Also

`bife`

coef.bife

Extract estimates of structural parameters or fixed effects

Description

`coef.bife` is a generic function which extracts estimates of the structural parameters or fixed effects from objects returned by `bife`.

Usage

```
## S3 method for class 'bife'
coef(object, type = c("sp", "fe"), corrected = NULL,
      fixed = NULL, ...)
```

Arguments

<code>object</code>	an object of class "bife".
<code>type</code>	the type of parameter estimates that should be returned; structural parameters or fixed effects. Default is "sp" referring to the structural parameters.
<code>corrected, fixed</code>	deprecated.
<code>...</code>	other arguments.

Value

The function `coef.bife` returns a named vector of estimates of the requested parameters.

See Also

`bife`

coef.bifeAPEs	<i>Extract estimates of average partial effects</i>
---------------	---

Description

`coef.bifeAPEs` is a generic function which extracts estimates of the average partial effects from objects returned by `get_APEs`.

Usage

```
## S3 method for class 'bifeAPEs'  
coef(object, ...)
```

Arguments

object	an object of class "APEs".
...	other arguments.

Value

The function `coef.bifeAPEs` returns a named vector of estimates of the average partial effects.

See Also

[get_APEs](#)

fitted.bife	<i>Extract bife fitted values</i>
-------------	-----------------------------------

Description

`fitted.bife` is a generic function which extracts fitted values from an object returned by `bife`.

Usage

```
## S3 method for class 'bife'  
fitted(object, ...)
```

Arguments

object	an object of class "bife".
...	other arguments.

Value

The function `fitted.bife` returns a vector of fitted values.

See Also[bife](#)

get_APEs	<i>Compute average partial effects for binary choice models with fixed effects</i>
----------	--

Description

[get_APEs](#) is a post-estimation routine that can be used to estimate average partial effects with respect to all covariates in the model and the corresponding covariance matrix. The estimation of the covariance is based on a linear approximation (delta method). Note that the command automatically determines which of the regressors are continuous or binary.

Remark: The routine currently does not allow to compute average partial effects based on functional forms like interactions and polynomials.

Note: [apeff_bife](#) is deprecated and will be removed soon.

Usage

```
get_APEs(object, n_pop = NULL, weak_exo = FALSE)
```

```
apeff_bife(...)
```

Arguments

object	an object of class "bife".
n_pop	unsigned integer indicating a finite population correction for the estimation of the covariance matrix of the average partial effects proposed by Cruz-Gonzalez, Fernandez-Val, and Weidner (2017). The correction factor is computed as follows: $(n^* - n)/(n^* - 1)$, where n^* and n are the size of the entire population and the full sample size. Default is NULL, which refers to a factor of one and is equal to an infinitely large population.
weak_exo	logical indicating if some of the regressors are assumed to be weakly exogenous (e.g. predetermined). If object is returned by bias_corr , the option will be automatically set to TRUE if the chosen bandwidth parameter is larger than zero. Note that this option only affects the estimation of the covariance matrix. Default is FALSE, which assumes that all regressors are strictly exogenous.
...	arguments passed to the deprecated function apeff_bife .

Value

The function [get_APEs](#) returns a named list of class "bifeAPEs".

References

- Cruz-Gonzalez, M., Fernandez-Val, I., and Weidner, M. (2017). "Bias corrections for probit and logit models with two-way fixed effects". *The Stata Journal*, 17(3), 517-545.
- Fernandez-Val, I. (2009). "Fixed effects estimation of structural parameters and marginal effects in panel probit models". *Journal of Econometrics* 150(1), 71-85.
- Fernandez-Val, I. and Weidner M. (2018). "Fixed effects estimation of large-t panel data models". *Annual Review of Economics*, 10, 109-138.
- Neyman, J. and Scott, E. L. (1948). "Consistent estimates based on partially consistent observations". *Econometrica*, 16(1), 1-32.
- Stammann, A., Heiss, F., and and McFadden, D. (2016). "Estimating Fixed Effects Logit Models with Large Panel Data". Working paper.

See Also

[bias_corr](#), [bife](#)

Examples

```
# Load 'psid' dataset
library(bife)
dataset <- psid

# Fit a static logit model
mod <- bife(LFP ~ I(AGE^2) + log(INCH) + KID1 + KID2 + KID3 + factor(TIME) | ID, dataset)
summary(mod)

# Compute average partial effects
mod_ape <- get_APEs(mod)
summary(mod_ape)

# Apply analytical bias-correction
mod_bc <- bias_corr(mod)
summary(mod_bc)

# Compute bias-corrected average partial effects
mod_ape_bc <- get_APEs(mod_bc)
summary(mod_ape_bc)
```

predict.bife

Predict method for bife fits

Description

[predict.bife](#) is a generic function which obtains predictions from an object returned by [bife](#).

Usage

```
## S3 method for class 'bife'
predict(object, type = c("link", "response"),
        X_new = NULL, alpha_new = NULL, corrected = NULL, ...)
```

Arguments

object	an object of class "bife".
type	the type of prediction required. "link" is on the scale of the linear predictor whereas "response" is on the scale of the response variable. Default is "link".
X_new	a regressor matrix for predictions. If not supplied predictions are based on the regressor matrix returned by the object bife . See Details.
alpha_new	a scalar or vector of fixed effects. If not supplied predictions are based on the vector of fixed effects returned by bife . See Details.
corrected	deprecated.
...	other arguments

Details

The model frame returned by the object [bife](#) only includes individuals that were not dropped before the fitting process (due to perfect classification). The linear predictors of perfectly classified observations are equal to $-\infty$ or ∞ whereas the predicted probabilities are equal to their response. In-sample predictions are only based on non-perfectly classified observations.

If `alpha_new` is supplied as a scalar the linear predictor is computed using the same value of the fixed effect for each observation. If `alpha_new` is supplied as a vector it has to be of same length as the rows of the corresponding regressor matrix.

Value

The function `predict.bife` returns a vector of predictions.

See Also

[bife](#)

print.bife

Print bife

Description

[print.bife](#) is a generic function which displays some minimal information from objects returned by [bife](#).

Usage

```
## S3 method for class 'bife'
print(x, digits = max(3L, getOption("digits") - 3L), ...)
```

Arguments

x an object of class "bife".

digits unsigned integer indicating the number of decimal places. Default is max(3L, getOption("digits") - 3L).

... other arguments.

See Also

[bife](#)

print.bifeAPEs	<i>Print bifeAPEs</i>
----------------	-----------------------

Description

[print.bifeAPEs](#) is a generic function which displays some minimal information from objects returned by [get_APEs](#).

Usage

```
## S3 method for class 'bifeAPEs'
print(x, digits = max(3L, getOption("digits") - 3L),
      ...)
```

Arguments

x an object of class "bifeAPEs".

digits unsigned integer indicating the number of decimal places. Default is max(3L, getOption("digits") - 3L).

... other arguments.

See Also

[get_APEs](#)

```
print.summary.bife      Print summary.bife
```

Description

[print.summary.bife](#) is a generic function which displays summary statistics from objects returned by [summary.bife](#).

Usage

```
## S3 method for class 'summary.bife'
print(x, digits = max(3L, getOption("digits")) -
      3L), ...)
```

Arguments

<code>x</code>	an object of class "summary.bife".
<code>digits</code>	unsigned integer indicating the number of decimal places. Default is <code>max(3L, getOption("digits")) - 3L</code> .
<code>...</code>	other arguments.

See Also

[bife](#)

```
print.summary.bifeAPEs
      Print summary.bifeAPEs
```

Description

[print.summary.bifeAPEs](#) is a generic function which displays summary statistics from objects returned by [summary.bifeAPEs](#).

Usage

```
## S3 method for class 'summary.bifeAPEs'
print(x, digits = max(3L, getOption("digits")) -
      3L), ...)
```

Arguments

<code>x</code>	an object of class "summary.bifeAPEs".
<code>digits</code>	unsigned integer indicating the number of decimal places. Default is <code>max(3L, getOption("digits")) - 3L</code> .
<code>...</code>	other arguments.

See Also[get_APEs](#)

psid	<i>Female labor force participation</i>
------	---

Description

The sample was obtained from the "Panel Study of Income Dynamics" and contains information about $N = 1461$ women that were observed over $T = 9$ years.

Usage

psid

Format

A data frame with 13,149 rows:

ID individual identifier

LFP labor force participation

KID1 # of kids aged between 0 and 2

KID2 # of kids aged between 3 and 5

KID3 # of kids aged between 6 and 17

INCH income husband

AGE age of woman

TIME time identifier

References

Hyslop, D. (1999). "State Dependence, Serial Correlation and Heterogeneity in Intertemporal Labor Force Participation of Married Women". *Econometrica* 67(6), 1255-1294.

Fernandez-Val, I. (2009). "Fixed effects estimation of structural parameters and marginal effects in panel probit models". *Journal of Econometrics* 150(1), 71-85.

See Also[bife](#)

summary.bife	<i>Summarizing models of class bife</i>
--------------	---

Description

Summary statistics for objects of class "bife".

Usage

```
## S3 method for class 'bife'
summary(object, type = c("sp", "fe"), corrected = NULL,
        fixed = NULL, ...)
```

Arguments

object	an object of class "bife".
type	the type of parameter estimates the summary statistics are related to: structural parameters or fixed effects. Default is "sp" referring to the structural parameters.
corrected, fixed	deprecated.
...	other arguments.

Value

Returns an object of class "summary.bife" which is a list of summary statistics of object.

See Also

[bife](#)

summary.bifeAPEs	<i>Summarizing models of class bifeAPEs</i>
------------------	---

Description

Summary statistics for objects of class "bifeAPEs".

Usage

```
## S3 method for class 'bifeAPEs'
summary(object, ...)
```

Arguments

object an object of class "bifeAPEs".
... other arguments.

Value

Returns an object of class "summary.bifeAPEs" which is a list of summary statistics of object.

See Also

[get_APEs](#)

vcov.bife

Extract estimates of the covariance matrix

Description

[vcov.bife](#) computes an estimate of the covariance matrix of the estimator of the structural parameters from objects returned by [bife](#). The estimate is obtained using the inverse of the negative Hessian after convergence.

Usage

```
## S3 method for class 'bife'  
vcov(object, ...)
```

Arguments

object an object of class "bife".
... other arguments.

Value

The function [vcov.bife](#) returns a named matrix of covariance estimates.

See Also

[bife](#)

vcov.bifeAPes	<i>Extract estimates of the covariance matrix</i>
---------------	---

Description

[vcov.bifeAPes](#) computes an estimate of the covariance matrix of the estimator of the average partial parameters from objects returned by [get_APES](#).

Usage

```
## S3 method for class 'bifeAPes'  
vcov(object, ...)
```

Arguments

object	an object of class "bifeAPes".
...	other arguments.

Value

The function [vcov.bifeAPes](#) returns a named matrix of covariance estimates.

See Also

[get_APES](#)

Index

*Topic **datasets**

psid, [13](#)

apeff_bife, [8](#)

apeff_bife (get_APEs), [8](#)

bias_corr, [2](#), [2](#), [3](#), [4](#), [8](#), [9](#)

bife, [3](#), [3](#), [4–15](#)

bife_control, [4](#), [5](#), [6](#)

coef.bife, [6](#), [6](#)

coef.bifeAPEs, [7](#), [7](#)

fitted.bife, [7](#), [7](#)

get_APEs, [7](#), [8](#), [8](#), [11](#), [13](#), [15](#), [16](#)

glm, [4](#)

predict.bife, [9](#), [9](#)

print.bife, [10](#), [10](#)

print.bifeAPEs, [11](#), [11](#)

print.summary.bife, [12](#), [12](#)

print.summary.bifeAPEs, [12](#), [12](#)

psid, [13](#)

summary.bife, [12](#), [14](#)

summary.bifeAPEs, [12](#), [14](#)

vcov.bife, [15](#), [15](#)

vcov.bifeAPEs, [16](#), [16](#)