Package ‘xtreg2way’

August 11, 2020

Type Package

Title Feasible Estimation of the Two-Way Fixed Effect Model

Version 1.0.0

Description Implemented is an algorithm to estimate the two-way fixed effect linear model. The coefficients of interest are computed using the residuals from the projection of all variables on the two sets of fixed effects. Our algorithm has three desirable features. First, it manages memory and computational resources efficiently which speeds up the computation of the estimates. Second, it allows the researcher to estimate multiple specifications using the same set of fixed effects at a very low computational cost. Third, the asymptotic variance of the parameters of interest can be consistently estimated using standard routines on the residualized data.


License MIT + file LICENSE

URL https://github.com/paulosomaini/xtreg2way

Encoding UTF-8

LazyData true

Imports pracma, Matrix, stats, MASS

RoxygenNote 7.1.1

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

NeedsCompilation no

Author Paulo Somaini [aut, cre, cph],
Frank Wolak [aut],
Brian Chivers [aut]

Maintainer Paulo Somaini <soma@stanford.edu>

Repository CRAN

Date/Publication 2020-08-11 08:40:03 UTC
**R topics documented:**

avar .................................................. 2
ids_with_multiple_obs ................................ 3
nonredundant ........................................ 3
projdummies ......................................... 4
projvar ................................................ 4
regress1 ............................................... 5
where_id_with_single_obs ........................... 6
xtreg2way ............................................. 6

Index 9

---

**avar**  
Asymptotic variance of Estimator

---

**Description**

avar calculated the asymptotic variance of the regression estimation

**Usage**

avar(X, e, group = NULL, J = NULL)

**Arguments**

- **X**: A matrix or vector of independent variable(s)
- **e**: The residuals from the regression
- **group**: (optional) The cluster identifier (hhid from projdummies)
- **J**: (optional) This is assumed to be $X'X$, and can be input if pre-calculated

**Value**

A matrix of the covariates

**Examples**

```r
hhid <- c("a","b","c","a","b","c","a","b","c","a","b","c")
tid <- c("1","1","1","2","2","3","3","3","4","4","5","5","6","6")
w <- rep(1, 15)
x1 <- rnorm(15, mean=50, sd=10)
x2 <- rnorm(15, mean=50, sd=10)
y <- x1 + rnorm(15, mean=50, sd=10)
struc <- projdummies(hhid, tid, w)
projvar_list <- projvar(x1, struc)
x1p <- projvar_list$var
projvar_list <- projvar(x2, struc)
x2p <- projvar_list$var
```
```r
projvar_list <- projvar(y, struc)
yp <- projvar_list$var
reg <- regress1(yp, data.frame(x1p, x2p))
matCov <- avar(data.frame(x1p, x2p), reg$res, struc$hhid, reg$XX)
```

### ids_with_multiple_obs

**Description**

`ids_with_multiple_obs` returns a vector of values that occur multiple times.

**Usage**

```r
ids_with_multiple_obs(id)
```

**Arguments**

- `id`: A vector of dummy indicators.

**Value**

A vector of repeated values.

### nonredundant

**Description**

`nonredundant` flags for redundant dummy levels.

**Usage**

```r
nonredundant(iid, tid, w)
```

**Arguments**

- `iid`: A vector of group dummy indicators.
- `tid`: A vector of time dummy indicators.
- `w`: A vector of non-negative weights.

**Value**

A list will be returned with the following named values: `flag` - Are there redundant dummy levels? `nr` - A listing of
projdummies  

**Projection Dummies**

**Description**

projdummies returns necessary matrices to project variables on fixed effect dummies. The input parameters all need to be of the same length.

**Usage**

projdummies(hhid, tid, w)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>hhid</td>
<td>A vector of individual effect identifiers</td>
</tr>
<tr>
<td>tid</td>
<td>A vector of time effect identifiers</td>
</tr>
<tr>
<td>w</td>
<td>A vector of weights for each observation</td>
</tr>
</tbody>
</table>

**Value**

A list will be returned with necessary matrices to project upon. If the time effect has more levels, the matrices B, C, invDD, and invDDDH will be returned. If the individual effect has more levels, the matrices A, B, invHH and invHHDH will be returned.

hhid and tid as factors will always be returned, as well as the original weights w that are passed.

**Examples**

```r
hhid <- c("a", "b", "c", "a", "b", "c", "a", "b", "c", "a", "b", "c")
tid <- c("1", "1", "2", "2", "3", "3", "3", "4", "4", "5", "6", "6", "6")
w <- rep(1, 15)
projdummies(hhid, tid, w)
```

projvar  

**Project Variables**

**Description**

projvar uses the matrices from projdummies to project a variable. In practice, all dependent and independent variables must be project for the two way regression.

**Usage**

projvar(var, struc)
Arguments

var A vector of a single variable
struc The output of projdummies, containing matrices necessary to project

Value

A list will be returned with the following named values: var - the projected variable delta, tau - intermediate variables

Examples

hhid <- c("a", "b", "c", "a", "b", "c", "a", "b", "c")
tid <- c("1", "1", "2", "2", "3", "3", "4", "5", "5", "6", "6")
w <- rep(1, 15)
x1 <- rnorm(15, mean=50, sd=10)

struc <- projdummies(hhid, tid, w)
x1p <- projvar(x1, struc)

regress1

Regression

Description

regress1 performs an OLS regression based on the projected variables y and X.

Usage

regress1(y, X)

Arguments

y The dependent variable
X A matrix or vector of independent variable(s)

Value

A list which contains X'X, the returned coefficients beta, and residuals res

Examples

hhid <- c("a", "b", "c", "a", "b", "c", "a", "b", "c")
tid <- c("1", "1", "2", "2", "3", "3", "4", "5", "5", "6", "6")
w <- rep(1, 15)
x1 <- rnorm(15, mean=50, sd=10)
x2 <- rnorm(15, mean=50, sd=10)
y <- x1 + rnorm(15, mean=50, sd=10)
struc <- projdummies(hhid, tid, w)
projvar_list <- projvar(x1, struc)
x1p <- projvar_list$var
projvar_list <- projvar(x2, struc)
x2p <- projvar_list$var
projvar_list <- projvar(y, struc)
yp <- projvar_list$var

reg <- regress1(yp, data.frame(x1p,x2p))

---

where_id_with_single_obs

where_id_with_single_obs returns a true/false vector, Does this dummy occur multiple times?

---

### Description

where_id_with_single_obs returns a true/false vector, Does this dummy occur multiple times?

### Usage

where_id_with_single_obs(id)

### Arguments

- **id**  
  A vector of dummy indicators

### Value

A true/false vector

---

**xtreg2way**  

**Estimation of Two Way Fixed Effects Model**

### Description

xtreg2way estimates a 2-way fixed effect model absorbing the two set of dummies and reports standard errors.
Usage

xtreg2way(y, ...)

## S3 method for class 'formula'
xtreg2way(
  formula,
  data,
  iid = NULL,
  tid = NULL,
  w = NULL,
  struc = NULL,
  se = "",
  noise = "",
  ...
)

## Default S3 method:
xtreg2way(
  y,
  X,
  iid = NULL,
  tid = NULL,
  w = NULL,
  struc = NULL,
  se = "",
  noise = "",
  ...
)

Arguments

y          Either a numeric of data, or a formula
...
formul    Other parameters, based on method used
forma      Model specifications
data       A dataframe with labels from the formula y
iid        (optional) The group ID, size N-by-1 - not needed if struc is provided
tid        (optional) The time ID, size N-by-1 - not needed if struc is provided
w          (optional) The vector of weights, size N-by-1. If omitted, w will be set to 1 for
            all observations
struc      (optional) This list contains the results from the first step of the algorithm. To
            save computational time, you can rerun the algorithm on different columns by
            providing this struc.
se         (optional) This indicates standard error estimate to be calculated. Possible val-
            ues include: se=="0" : standard errors assuming homoscedasticity and no within
            group correlation or serial correlation. se=="1" : standard errors proposed by
            Arellano (1987) robust to heteroscedasticity and serial correlation. se=="2" :
standard errors robust to heteroscedasticity but assumes no correlation within group or serial correlation. se="11": Arellano standard errors with a degree of freedom correction performed by Stata xtreg, fe. If se is omitted or set to [] then it is set to 1 and the Arellano (1987) estimator is computed.

noise (optional) If noise is set to "1", then results are displayed

X A matrix of data

Value
betaHat (K-by-1) vector of estimated coefficients
aVarHat (K-by-K) estimate of the matrix of variances and covariance of the estimator.
y (N-by-1) the residual of the projection of y on the two sets of dummies.
X (N-by-K) the residual of the projection of each column of X on the two sets of dummies.
struc (list) results of the first step of the algorithm.

Methods (by class)
• formula: This function ingests a formula as the first argument, and requires data as a data.frame
• default: Default Method

Examples
hhid <- c("a","b","c","a","b","c","a","b","c","a","b","c")
tid <- c("1","1","1","2","2","3","3","3","4","4","5","5","6","6")
w <- rep(1, 15)
x1 <- rnorm(15, mean=50, sd=10)
x2 <- rnorm(15, mean=50, sd=10)
y <- x1 + rnorm(15, mean=50, sd=10)
#The most basic way to use this function
output <- xtreg2way(y, x1, hhid, tid, w, se="2", noise="1")
#You can rerun faster with different columns using output$struc
output2 <- xtreg2way(y, data.frame(x1,x2), struc=output$struc)
#Or you can use a formula and specify data=
output3 <- xtreg2way(y~x1+x2, data=data.frame(x1,x2,y), iid=hhid, tid=tid, w=w, se="2", noise="1")
Index

avar, 2
ids_with_multiple_obs, 3
nonredundant, 3
projdummies, 4
projvar, 4
regress1, 5
where_id_with_single_obs, 6
xtreg2way, 6