Package ‘qte’

May 9, 2026

Title Quantile Treatment Effects

Version 1.3.1

Description Provides several methods for computing the Quantile Treatment Effect (QTE) and Quantile Treatment Effect on the Treated (QTT). The main cases covered are (i) Treatment is randomly assigned, (ii) Treatment is as good as randomly assigned after conditioning on some covariates (also called conditional independence or selection on observables) using the methods developed in Firpo (2007) <doi:10.1111/j.1468-0262.2007.00738.x>, (iii) Identification is based on a Difference in Differences assumption (several varieties are available in the package e.g. Athey and Imbens (2006) <doi:10.1111/j.1468-0262.2006.00668.x> Callaway and Li (2019) <doi:10.3982/QE935>, Callaway, Li, and Oka (2018) <doi:10.1016/j.jeconom.2018.06.008>).

Depends R (>= 3.5)

Imports Hmisc, parallel, quantreg, BMisc, formula.tools, ggplot2, texreg, pbapply, data.table

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 7.2.1

VignetteBuilder knitr

Suggests rmarkdown, knitr, msm

NeedsCompilation no

Author Brantly Callaway [aut, cre]

Maintainer Brantly Callaway `<brantly.callaway@uga.edu>`

Repository CRAN

Date/Publication 2022-09-01 14:30:02 UTC

Contents

`bounds` *bounds*

Description

bounds estimates bounds for the Quantile Treatment Effect on the Treated (QTET) using the method of Fan and Yu (2012).

Usage

```
bounds(
  formla,
  xformla = NULL,
  t,
  tmin1,
  tname,
```

```
  data,
  idname,
  probs = seq(0.05, 0.95, 0.05)
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary), d should be equal to one in all time periods for individuals that are eventually treated
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`t`	The 3rd time period in the sample. Treated individuals should be treated in this time period and untreated individuals should not be treated. The code attempts to enforce this condition, but it is good try to handle this outside the panel.qtet method.
`tmin1`	The 2nd time period in the sample. This should be a pre-treatment period for all individuals in the sample.
`tname`	The name of the column containing the time periods
`data`	A data.frame containing all the variables used
`idname`	The individual (cross-sectional unit) id name
`probs`	A vector of values between 0 and 1 to compute the QTET at

Value

A `BoundsObj` object

References

Fan, Yanqin and Zhengfei Yu. “Partial Identification of Distributional and Quantile Treatment Effects in Difference-in-Differences Models.” Economics Letters 115.3, pp.511-515, 2012.

Examples

```
## load the data
data(lalonde)

## Run the bounds method with no covariates
b1 <- bounds(re ~ treat, t=1978, tmin1=1975, data=lalonde.psid.panel,
  idname="id", tname="year")
summary(b1)
```

`ci.qte` *ci.qte*

Description

The `ci.qtet` method implements estimates the Quantile Treatment Effect (QTE) under a Conditional Independence Assumption (sometimes this is called Selection on Observables) developed in Firpo (2007). This method using propensity score re-weighting and minimizes a check function to compute the QTET. Standard errors (if requested) are computed using the bootstrap.

Usage

```
ci.qte(
  formla,
  xformla = NULL,
  x = NULL,
  data,
  w = NULL,
  probs = seq(0.05, 0.95, 0.05),
  se = TRUE,
  iters = 100,
  alp = 0.05,
  method = "logit",
  retEachIter = FALSE,
  printIter = FALSE,
  pl = FALSE,
  cores = 2
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary), d should be equal to one in all time periods for individuals that are eventually treated
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`x`	Vector of covariates. Default is no covariates
`data`	A data.frame containing all the variables used
`w`	an additional vector of sampling weights
`probs`	A vector of values between 0 and 1 to compute the QTET at
`se`	Boolean whether or not to compute standard errors
`iters`	The number of iterations to compute bootstrap standard errors. This is only used if se=TRUE
`alp`	The significance level used for constructing bootstrap confidence intervals

`method`	Method to compute propensity score. Default is logit; other option is probit.
`retEachIter`	Boolean whether or not to return list of results from each iteration of the bootstrap procedure (default is FALSE). This is potentially useful for debugging but can cause errors due to running out of memory.
`printIter`	For debugging only; should leave at default FALSE unless you want to see a lot of output
`pl`	boolean for whether or not to compute bootstrap error in parallel. Note that computing standard errors in parallel is a new feature and may not work at all on Windows.
`cores`	the number of cores to use if bootstrap standard errors are computed in parallel

Value

QTE object

References

Firpo, Sergio. “Efficient Semiparametric Estimation of Quantile Treatment Effects.” Econometrica 75.1, pp. 259-276, 2015.

Examples

```
## Load the data
data(lalonde)

##Estimate the QTET of participating in the job training program;
##This is the no covariate case.  Note: Because individuals that participate
## in the job training program are likely to be much different than
## individuals that do not (e.g. less experience and less education), this
## method is likely to perform poorly at estimating the true QTET
q1 <- ci.qte(re78 ~ treat, x=NULL, data=lalonde.psid, se=FALSE,
 probs=seq(0.05,0.95,0.05))
summary(q1)

##This estimation controls for all the available background characteristics.
q2 <- ci.qte(re78 ~ treat,
 xformla=~age + I(age^2) + education + black + hispanic + married + nodegree,
 data=lalonde.psid, se=FALSE, probs=seq(0.05, 0.95, 0.05))
summary(q2)
```

`ci.qtet` *ci.qtet*

Description

The `ci.qtet` method implements estimates the Quantile Treatment Effect on the Treated (QTET) under a Conditional Independence Assumption (sometimes this is called Selection on Observables) developed in Firpo (2007). This method using propensity score re-weighting and minimizes a check function to compute the QTET. Standard errors (if requested) are computed using the bootstrap.

Usage

```
ci.qtet(
  formla,
  xformla = NULL,
  w = NULL,
  data,
  probs = seq(0.05, 0.95, 0.05),
  se = TRUE,
  iters = 100,
  alp = 0.05,
  method = "logit",
  retEachIter = FALSE,
  indsample = TRUE,
  printIter = FALSE,
  pl = FALSE,
  cores = 2
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary), d should be equal to one in all time periods for individuals that are eventually treated
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`w`	an additional vector of sampling weights
`data`	A data.frame containing all the variables used
`probs`	A vector of values between 0 and 1 to compute the QTET at
`se`	Boolean whether or not to compute standard errors
`iters`	The number of iterations to compute bootstrap standard errors. This is only used if se=TRUE
`alp`	The significance level used for constructing bootstrap confidence intervals
`method`	Method to compute propensity score. Default is logit; other option is probit.
`retEachIter`	Boolean whether or not to return list of results from each iteration of the bootstrap procedure (default is FALSE). This is potentially useful for debugging but can cause errors due to running out of memory.
`indsample`	Binary variable for whether to treat the samples as independent or dependent. This affects bootstrap standard errors. In the job training example, the samples are independent because they are two samples collected independently and then merged. If the data is from the same source, usually should set this option to be FALSE.
`printIter`	For debugging only; should leave at default FALSE unless you want to see a lot of output
`pl`	Whether or not to compute standard errors in parallel
`cores`	Number of cores to use if computing in parallel

Value

QTE object

References

Firpo, Sergio. “Efficient Semiparametric Estimation of Quantile Treatment Effects.” Econometrica 75.1, pp. 259-276, 2015.

Examples

```
## Load the data
data(lalonde)

##Estimate the QTET of participating in the job training program;
##This is the no covariate case.  Note: Because individuals that participate
## in the job training program are likely to be much different than
## individuals that do not (e.g. less experience and less education), this
## method is likely to perform poorly at estimating the true QTET
q1 <- ci.qtet(re78 ~ treat, x=NULL, data=lalonde.psid, se=FALSE,
 probs=seq(0.05,0.95,0.05))
summary(q1)

##This estimation controls for all the available background characteristics.
q2 <- ci.qtet(re78 ~ treat,
 xformla=~age + I(age^2) + education + black + hispanic + married + nodegree,
 data=lalonde.psid, se=FALSE, probs=seq(0.05, 0.95, 0.05))
summary(q2)
```

`CiC` *Change in Changes*

Description

`CiC` computes the Quantile Treatment Effect on the Treated (QTET) using the method of Athey and Imbens (2006). `CiC` is a Difference in Differences type method. It requires having two periods of data that can be either repeated cross sections or panel data.

The method can accommodate conditioning on covariates though it does so in a restrictive way: It specifies a linear model for outcomes conditional on group-time dummies and covariates. Then, after residualizing (see details in Athey and Imbens (2006)), it computes the Change in Changes model based on these quasi-residuals.

Usage

```
CiC(
  formla,
  xformla = NULL,
  t,
```

```
  tmin1,
  tname,
  data,
  panel = FALSE,
  se = TRUE,
  idname = NULL,
  alp = 0.05,
  probs = seq(0.05, 0.95, 0.05),
  iters = 100,
  pl = FALSE,
  cores = 2,
  retEachIter = FALSE
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary), d should be equal to one in all time periods for individuals that are eventually treated
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`t`	The 3rd time period in the sample. Treated individuals should be treated in this time period and untreated individuals should not be treated. The code attempts to enforce this condition, but it is good try to handle this outside the panel.qtet method.
`tmin1`	The 2nd time period in the sample. This should be a pre-treatment period for all individuals in the sample.
`tname`	The name of the column containing the time periods
`data`	A data.frame containing all the variables used
`panel`	Binary variable indicating whether or not the dataset is panel. This is used for computing bootstrap standard errors correctly.
`se`	Boolean whether or not to compute standard errors
`idname`	The individual (cross-sectional unit) id name
`alp`	The significance level used for constructing bootstrap confidence intervals
`probs`	A vector of values between 0 and 1 to compute the QTET at
`iters`	The number of iterations to compute bootstrap standard errors. This is only used if se=TRUE
`pl`	Whether or not to compute standard errors in parallel
`cores`	Number of cores to use if computing in parallel
`retEachIter`	Boolean whether or not to return list of results from each iteration of the bootstrap procedure (default is FALSE). This is potentially useful for debugging but can cause errors due to running out of memory.

Value

QTE Object

References

Athey, Susan and Guido Imbens. “Identification and Inference in Nonlinear Difference-in-Differences Models.” Econometrica 74.2, pp. 431-497, 2006.

Examples

```
## load the data
data(lalonde)
## Run the Change in Changes model conditioning on age, education,
## black, hispanic, married, and nodegree
c1 <- CiC(re ~ treat, t=1978, tmin1=1975, tname="year",
 xformla=~age + I(age^2) + education + black + hispanic + married + nodegree,
 data=lalonde.psid.panel, idname="id", se=FALSE,
 probs=seq(0.05, 0.95, 0.05))
summary(c1)
```

`compute.panel.qtet` *compute.panel.qtet*

Description

`compute.panel.qtet` uses third period of data, combined with Distributional Difference in Differences assumption (Fan and Yu, 2012) to point identify QTET.

Usage

```
compute.panel.qtet(qp)
```

Arguments

qp	QTEparams object containing the parameters passed to ciqte

Value

QTE object

`computeDiffSE` *computeDiffSE*

Description

Takes two sets of initial estimates and bootstrap estimations (they need to have the same number of iterations) and determines whether or not the estimates are statistically different from each other. It can be used to compare any sets of estimates, but it is particularly used here to compare estimates from observational methods with observations from the experimental data (which also have standard errors because, even though the estimates are cleanly identified, they are still estimated).

Usage

```
computeDiffSE(est1, bootIters1, est2, bootIters2, alp = 0.05)
```

Arguments

`est1`	A QTE object containing the first set of estimates
`bootIters1`	A List of QTE objects that have been bootstrapped
`est2`	A QTE object containing a second set of estimates
`bootIters2`	A List of QTE objects that have been bootstrapped using the second method
`alp`	The significance level used for constructing bootstrap confidence intervals

---

`ddid2` *ddid2*

Description

`ddid2` computes the Quantile Treatment Effect on the Treated (QTET) using the method of Callaway, Li, and Oka (2015).

Usage

```
ddid2(
  formla,
  xformla = NULL,
  t,
  tmin1,
  tname,
  data,
  panel = TRUE,
  dropalwaystreated = TRUE,
  idname = NULL,
  probs = seq(0.05, 0.95, 0.05),
```

```
  iters = 100,
  alp = 0.05,
  method = "logit",
  se = TRUE,
  retEachIter = FALSE,
  seedvec = NULL,
  pl = FALSE,
  cores = NULL
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary)
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`t`	The 3rd time period in the sample (this is the name of the column)
`tmin1`	The 2nd time period in the sample (this is the name of the column)
`tname`	The name of the column containing the time periods
`data`	The name of the data.frame that contains the data
`panel`	Boolean indicating whether the data is panel or repeated cross sections
`dropalwaystreated`	How to handle always treated observations in panel data case (not currently used)
`idname`	The individual (cross-sectional unit) id name
`probs`	A vector of values between 0 and 1 to compute the QTET at
`iters`	The number of iterations to compute bootstrap standard errors. This is only used if se=TRUE
`alp`	The significance level used for constructing bootstrap confidence intervals
`method`	The method for estimating the propensity score when covariates are included
`se`	Boolean whether or not to compute standard errors
`retEachIter`	Boolean whether or not to return list of results from each iteration of the bootstrap procedure
`seedvec`	Optional value to set random seed; can possibly be used in conjunction with bootstrapping standard errors.
`pl`	boolean for whether or not to compute bootstrap error in parallel. Note that computing standard errors in parallel is a new feature and may not work at all on Windows.
`cores`	the number of cores to use if bootstrap standard errors are computed in parallel

Value

`QTE` object

References

Callaway, Brantly, Tong Li, and Tatsushi Oka. “Quantile Treatment Effects in Difference in Differences Models under Dependence Restrictions and with Only Two Time Periods.” Working Paper, 2015.

Examples

```
##load the data
data(lalonde)

## Run the ddid2 method on the observational data with no covariates
d1 <- ddid2(re ~ treat, t=1978, tmin1=1975, tname="year",
 data=lalonde.psid.panel, idname="id", se=FALSE,
 probs=seq(0.05, 0.95, 0.05))
summary(d1)

## Run the ddid2 method on the observational data with covariates
d2 <- ddid2(re ~ treat, t=1978, tmin1=1975, tname="year",
 data=lalonde.psid.panel, idname="id", se=FALSE,
 xformla=~age + I(age^2) + education + black + hispanic + married + nodegree,
 probs=seq(0.05, 0.95, 0.05))
summary(d2)
```

`diffQ` *diffQ*

Description

takes a single set of quantiles:
(not qtes as in diffquantiles) and returns the difference between particular ones

Usage

```
diffQ(qvec, tauvec, hightau, lowtau)
```

Arguments

`qvec`	vector of quantiles
`tauvec`	vector of tau (should be same length as qvec)
`hightau`	upper quantile
`lowtau`	lower quantile

Value

scalar difference between quantiles

DR *DR*

Description

A distribution regression object

Usage

```
DR(yvals, drlist)
```

Arguments

`yvals`	A vector of values that y can take
`drlist`	A list where for each value of y, a distribution regression

---

ggqte *ggqte*

Description

Makes somewhat nicer plots of quantile treatment effects by using ggplot

Usage

```
ggqte(
  qteobj,
  main = "",
  ylab = "QTE",
  ylim = NULL,
  ybreaks = NULL,
  xbreaks = c(0.1, 0.3, 0.5, 0.7, 0.9),
  setype = "pointwise"
)
```

Arguments

`qteobj`	a QTE object
`main`	optional title
`ylab`	optional y axis label
`ylim`	optional limits of y axis
`ybreaks`	optional breaks in y axis
`xbreaks`	optional breaks in x axis

`setype`	options are "pointwise", "uniform" or both; pointwise confidence intervals cover the QTE at each particular point with a fixed probability, uniform confidence bands cover the entire curve with a fixed probability. Uniform confidence bands will tend to be wider. The option "both" will plot both types of confidence intervals

Value

a ggplot object

`lalonde` *Lalonde (1986)'s NSW Dataset*

Description

`lalonde` contains data from the National Supported Work Demonstration. This program randomly assigned applicants to the job training program (or out of the job training program). The dataset is discussed in Lalonde (1986). The experimental part of the dataset is combined with an observational dataset from the Panel Study of Income Dynamics (PSID). Lalonde (1986) and many subsequent papers (e.g. Heckman and Hotz (1989), Dehejia and Wahba (1999), Smith and Todd (2005), and Firpo (2007) have used this combination to study the effectiveness of various 'observational' methods (e.g. regression, Heckman selection, Difference in Differences, and propensity score matching) of estimating the Average Treatment Effect (ATE) of participating in the job training program. The idea is that the results from the observational method can be compared to results that can be easily obtained from the experimental portion of the dataset.

To be clear, the observational data combines the observations that are treated from the experimental portion of the data with untreated observations from the PSID.

Usage

`data(lalonde)`

Format

Four data.frames: (i) lalonde.exp contains a cross sectional version of the experimental data, (ii) lalonde.psid contains a cross sectional version of the observational data, (iii) lalonde.exp.panel contains a panel version of the experimental data, and (iv) lalonde.psid.panel contains a panel version of the observational data. Note: the cross sectional and panel versions of each dataset are identical up to their shape; in demonstrating each of the methods, it is sometimes convenient to have one form of the data or the other.

References

LaLonde, Robert. "Evaluating the Econometric Evaluations of Training Programs with Experimental Data." The American Economics Review, pp. 604-620, 1986. @source The dataset comes from Lalonde (1986) and has been studied in much subsequent work. The `qte` package uses a version from the `causalsens` package (`https://CRAN.R-project.org/package=causalsens`)

`lalonde.exp` *Lalonde's Experimental Dataset*

Description

The cross sectional verion of the experimental part of the `lalonde` dataset. It is loaded with all the datasets with the command `data(lalonde)`

`lalonde.exp.panel` *Lalonde's Panel Experimental Dataset*

Description

The panel verion of the experimental part of the `lalonde` dataset. It is loaded with all the datasets with the command `data(lalonde)`

`lalonde.psid` *Lalonde's Observational Dataset*

Description

The cross sectional verion of the observational part of the `lalonde` dataset. It is loaded with all the datasets with the command `data(lalonde)`

`lalonde.psid.panel` *Lalonde's Experimental Dataset*

Description

The panel verion of the observational part of the `lalonde` dataset. It is loaded with all the datasets with the command `data(lalonde)`

`MDiD` *Mean Difference in Differences*

Description

`MDiD` is a Difference in Differences type method for computing the QTET.

The method can accommodate conditioning on covariates though it does so in a restrictive way: It specifies a linear model for outcomes conditional on group-time dummies and covariates. Then, after residualizing (see details in Athey and Imbens (2006)), it computes the Change in Changes model based on these quasi-residuals.

Usage

```
MDiD(
  formla,
  xformla = NULL,
  t,
  tmin1,
  tname,
  data,
  panel = FALSE,
  se = TRUE,
  idname = NULL,
  alp = 0.05,
  probs = seq(0.05, 0.95, 0.05),
  iters = 100,
  retEachIter = FALSE
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary), d should be equal to one in all time periods for individuals that are eventually treated
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`t`	The 3rd time period in the sample. Treated individuals should be treated in this time period and untreated individuals should not be treated. The code attempts to enforce this condition, but it is good try to handle this outside the panel.qtet method.
`tmin1`	The 2nd time period in the sample. This should be a pre-treatment period for all individuals in the sample.
`tname`	The name of the column containing the time periods
`data`	A data.frame containing all the variables used

`panel`	Binary variable indicating whether or not the dataset is panel. This is used for computing bootstrap standard errors correctly.
`se`	Boolean whether or not to compute standard errors
`idname`	The individual (cross-sectional unit) id name
`alp`	The significance level used for constructing bootstrap confidence intervals
`probs`	A vector of values between 0 and 1 to compute the QTET at
`iters`	The number of iterations to compute bootstrap standard errors. This is only used if se=TRUE
`retEachIter`	Boolean whether or not to return list of results from each iteration of the bootstrap procedure (default is FALSE). This is potentially useful for debugging but can cause errors due to running out of memory.

Value

A `QTE` object

References

Athey, Susan and Guido Imbens. “Identification and Inference in Nonlinear Difference-in-Differences Models.” Econometrica 74.2, pp. 431-497, 2006.

Thuysbaert, Bram. “Distributional Comparisons in Difference in Differences Models.” Working Paper, 2007.

Examples

```
## load the data
data(lalonde)

## Run the Mean Difference in Differences method conditioning on
## age, education, black, hispanic, married, and nodegree
md1 <- MDiD(re ~ treat, t=1978, tmin1=1975, tname="year",
 xformla=~age + I(age^2) + education + black + hispanic + married + nodegree,
 data=lalonde.psid.panel, idname="id", se=FALSE,
 probs=seq(0.05, 0.95, 0.05))
summary(md1)
```

`panel.checks` *panel.checks*

Description

Does some checking that data setup is valid for using methods in qte package

Usage

```
panel.checks(qp)
```

Arguments

`qp`	QTEparams object containing the parameters passed to ciqte

---

`panel.qtet` *panel.qtet*

Description

`panel.qtet` computes the Quantile Treatment Effect on the Treated (QTET) using the method of Callaway and Li (2015). This method should be used when the researcher wants to invoke a Difference in Differences assumption to identify the QTET. Relative to the other Difference in Differences methods available in the `qte` package, this method's assumptions are more intuitively similar to the identifying assumptions used in identifying the Average Treatment Effect on the Treated (ATT).

Additionally, this method can accommodate covariates in a more flexible way than the other Difference in Differences methods available. In order to accommodate covariates, the user should specify a vector `x` of covariate names. The user also may specify a method for estimating the propensity score. The default is logit.

`panel.qtet` can only be used in some situations, however. The method requires three periods of panel data where individuals are not treated until the last period. The data should be formatted as a panel; the names of columns containing time periods and ids for each cross sectional unit need to be passed to the method.

Usage

```
panel.qtet(
  formla,
  xformla = NULL,
  t,
  tmin1,
  tmin2,
  tname,
  data,
  idname,
  probs = seq(0.05, 0.95, 0.05),
  iters = 100,
  alp = 0.05,
  method = c("qr", "pscore"),
  se = TRUE,
  retEachIter = FALSE,
  pl = FALSE,
  cores = NULL
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary), d should be equal to one in all time periods for individuals that are eventually treated
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`t`	The 3rd time period in the sample. Treated individuals should be treated in this time period and untreated individuals should not be treated. The code attempts to enforce this condition, but it is good try to handle this outside the panel.qtet method.
`tmin1`	The 2nd time period in the sample. This should be a pre-treatment period for all individuals in the sample.
`tmin2`	The 1st time period in the sample. This should be a pre-treatment period for all individuals in the sample.
`tname`	The name of the column containing the time periods
`data`	A data.frame containing all the variables used
`idname`	The individual (cross-sectional unit) id name
`probs`	A vector of values between 0 and 1 to compute the QTET at
`iters`	The number of iterations to compute bootstrap standard errors. This is only used if se=TRUE
`alp`	The significance level used for constructing bootstrap confidence intervals
`method`	The method for including covariates, should either be "QR" for quantile regression or "pscore" for propensity score
`se`	Boolean whether or not to compute standard errors
`retEachIter`	Boolean whether or not to return list of results from each iteration of the bootstrap procedure (default is FALSE). This is potentially useful for debugging but can cause errors due to running out of memory.
`pl`	Whether or not to compute standard errors in parallel
`cores`	Number of cores to use if computing in parallel

Value

`QTE` object

References

Callaway, Brantly and Tong Li. “Quantile Treatment Effects in Difference in Differences Models with Panel Data.” Working Paper, 2019.

Examples

```
##load the data
data(lalonde)

## Run the panel.qtet method on the experimental data with no covariates
pq1 <- panel.qtet(re ~ treat, t=1978, tmin1=1975, tmin2=1974, tname="year",
 data=lalonde.exp.panel, idname="id", se=FALSE,
 probs=seq(0.05, 0.95, 0.05))
summary(pq1)

## Run the panel.qtet method on the observational data with no covariates
pq2 <- panel.qtet(re ~ treat, t=1978, tmin1=1975, tmin2=1974, tname="year",
 data=lalonde.psid.panel, idname="id", se=FALSE,
 probs=seq(0.05, 0.95, 0.05))
summary(pq2)

## Run the panel.qtet method on the observational data conditioning on
## age, education, black, hispanic, married, and nodegree.
## The propensity score will be estimated using the default logit method.
pq3 <- panel.qtet(re ~ treat, t=1978, tmin1=1975, tmin2=1974, tname="year",
 xformla=~age + I(age^2) + education + black + hispanic + married + nodegree,
 data=lalonde.psid.panel, idname="id", se=FALSE, method="pscore",
 probs=seq(0.05, 0.95, 0.05))
summary(pq3)

pq4 <- panel.qtet(re ~ treat, t=1978, tmin1=1975, tmin2=1974, tname="year",
 xformla=~age + I(age^2) + education + black + hispanic + married + nodegree,
 data=lalonde.psid.panel, idname="id", se=FALSE, method="qr",
 probs=seq(0.05, 0.95, 0.05))
summary(pq4)
```

`panelize.data` *panelize.data*

Description

get data in correct format for using panel methods in qte package

Usage

```
panelize.data(data, idname, tname, t, tmin1, tmin2 = NULL)
```

Arguments

`data`	A data.frame containing all the variables used
`idname`	The individual (cross-sectional unit) id name
`tname`	The name of the column containing the time periods

`t`	The 3rd time period in the sample. Treated individuals should be treated in this time period and untreated individuals should not be treated. The code attempts to enforce this condition, but it is good try to handle this outside the panel.qtet method.
`tmin1`	The 2nd time period in the sample. This should be a pre-treatment period for all individuals in the sample.
`tmin2`	The 1st time period in the sample. This should be a pre-treatment period for all individuals in the sample.

Value

data.frame

`plot.BoundsObj` *Plot Bounds*

Description

Plots a BoundObj Object

Usage

```
## S3 method for class 'BoundsObj'
plot(
  x,
  plotate = FALSE,
  plot0 = FALSE,
  qtecol = "black",
  atecol = "black",
  col0 = "black",
  ylim = NULL,
  uselegend = FALSE,
  legloc = "topright",
  ...
)
```

Arguments

`x`	A BoundsObj Object
`plotate`	Boolean whether or not to plot the ATE
`plot0`	Boolean whether to plot a line at 0
`qtecol`	Color for qte plot. Default "black"
`atecol`	Color for ate plot. Default "black"
`col0`	Color for 0 plot. Default "black"

`ylim`	The ylim for the plot; if not passed, it will be automatically set based on the values that the QTE takes
`uselegend`	Boolean whether or not to print a legend
`legloc`	String location for the legend. Default "topright"
`...`	Other parameters to be passed to plot (e.g lwd)

`plot.QTE` *plot.QTE*

Description

Plots a QTE Object

Usage

```
## S3 method for class 'QTE'
plot(
  x,
  plotate = FALSE,
  plot0 = FALSE,
  qtecol = "black",
  atecol = "black",
  col0 = "black",
  xlab = "tau",
  ylab = "QTE",
  legend = NULL,
  ontreated = FALSE,
  ylim = NULL,
  uselegend = FALSE,
  legendcol = NULL,
  legloc = "topright",
  ...
)
```

Arguments

`x`	a QTE Object
`plotate`	Boolean whether or not to plot the ATE
`plot0`	Boolean whether to plot a line at 0
`qtecol`	Color for qte plot. Default "black"
`atecol`	Color for ate plot. Default "black"
`col0`	Color for 0 plot. Default "black"
`xlab`	Custom label for x-axis. Default "tau"
`ylab`	Custom label for y-axis. Default "QTE"

`legend`	Vector of strings to add to legend
`ontreated`	Boolean whether parameters are "on the treated group"
`ylim`	The ylim for the plot; if not passed, it will be automatically set based on the values that the QTE takes
`uselegend`	Boolean whether or not to print a legend
`legendcol`	Legend Colors for plotting
`legloc`	String location for the legend. Default "topright"
`...`	Other parameters to be passed to plot (e.g lwd)

`print.summary.BoundsObj` *Print a summary.BoundsObj*

Description

Prints a Summary QTE Object

Usage

```
## S3 method for class 'summary.BoundsObj'
print(x, ...)
```

Arguments

`x`	A summary.BoundsObj
`...`	Other objects to pass (not used)

`print.summary.QTE` *Print*

Description

Prints a Summary QTE Object

Usage

```
## S3 method for class 'summary.QTE'
print(x, ...)
```

Arguments

`x`	A summary.QTE object
`...`	Other params (required as generic function, but not used)

`QDiD` *Quantile Difference in Differences*

Description

`QDiD` is a Difference in Differences type method for computing the QTET.

The method can accommodate conditioning on covariates though it does so in a restrictive way: It specifies a linear model for outcomes conditional on group-time dummies and covariates. Then, after residualizing (see details in Athey and Imbens (2006)), it computes the Change in Changes model based on these quasi-residuals.

Usage

```
QDiD(
  formla,
  xformla = NULL,
  t,
  tmin1,
  tname,
  data,
  panel = FALSE,
  se = TRUE,
  idname = NULL,
  alp = 0.05,
  probs = seq(0.05, 0.95, 0.05),
  iters = 100,
  retEachIter = FALSE,
  pl = FALSE,
  cores = NULL
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary), d should be equal to one in all time periods for individuals that are eventually treated
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`t`	The 3rd time period in the sample. Treated individuals should be treated in this time period and untreated individuals should not be treated. The code attempts to enforce this condition, but it is good try to handle this outside the panel.qtet method.
`tmin1`	The 2nd time period in the sample. This should be a pre-treatment period for all individuals in the sample.

`tname`	The name of the column containing the time periods
`data`	A data.frame containing all the variables used
`panel`	Binary variable indicating whether or not the dataset is panel. This is used for computing bootstrap standard errors correctly.
`se`	Boolean whether or not to compute standard errors
`idname`	The individual (cross-sectional unit) id name
`alp`	The significance level used for constructing bootstrap confidence intervals
`probs`	A vector of values between 0 and 1 to compute the QTET at
`iters`	The number of iterations to compute bootstrap standard errors. This is only used if se=TRUE
`retEachIter`	Boolean whether or not to return list of results from each iteration of the bootstrap procedure (default is FALSE). This is potentially useful for debugging but can cause errors due to running out of memory.
`pl`	Whether or not to compute standard errors in parallel
`cores`	Number of cores to use if computing in parallel

Value

QTE Object

References

Athey, Susan and Guido Imbens. “Identification and Inference in Nonlinear Difference-in-Differences Models.” Econometrica 74.2, pp. 431-497, 2006.

Examples

```
## load the data
data(lalonde)

## Run the Quantile Difference in Differences method conditioning on
## age, education, black, hispanic, married, and nodegree
qd1 <- QDiD(re ~ treat, t=1978, tmin1=1975, tname="year",
 xformla=~age + I(age^2) + education + black + hispanic + married + nodegree,
 data=lalonde.psid.panel, idname="id", se=FALSE,
 probs=seq(0.05, 0.95, 0.05))
summary(qd1)
```

`QTE` *qte: A package for computating quantile treatment effects*

Description

Main class of objects. A `QTE` object is returned by all of the methods that compute the QTE or QTET.

Usage

```
QTE(
  qte,
  ate = NULL,
  qte.se = NULL,
  qte.lower = NULL,
  qte.upper = NULL,
  ate.se = NULL,
  ate.lower = NULL,
  ate.upper = NULL,
  c = NULL,
  pscore.reg = NULL,
  probs,
  type = "On the Treated",
  F.treated.t = NULL,
  F.untreated.t = NULL,
  F.treated.t.cf = NULL,
  F.treated.tmin1 = NULL,
  F.treated.tmin2 = NULL,
  F.treated.change.tmin1 = NULL,
  F.untreated.change.t = NULL,
  F.untreated.change.tmin1 = NULL,
  F.untreated.tmin1 = NULL,
  F.untreated.tmin2 = NULL,
  condQ.treated.t = NULL,
  condQ.treated.t.cf = NULL,
  eachIterList = NULL,
  inffunct = NULL,
  inffuncu = NULL
)
```

Arguments

`qte`	The Quantile Treatment Effect at each value of probs
`ate`	The Average Treatment Effect (or Average Treatment Effect on the Treated)
`qte.se`	A vector of standard errors for each qte

`qte.lower`	A vector of lower confidence intervals for each qte (it is based on the bootstrap confidence interval – not the se – so it may not be symmyetric about the qte
`qte.upper`	A vector of upper confidence intervals for each qte (it is based on the bootstrap confidence interval – not the se – so it may not be symmetric about the qte
`ate.se`	The standard error for the ATE
`ate.lower`	Lower confidence interval for the ATE (it is based on the bootstrap confidence intervall – not the se – so it may not be symmetric about the ATE
`ate.upper`	Upper confidence interval for the ATE (it is based on the bootstrap confidence interval – not the se – so it may not be symmetric about the ATE
`c`	The critical value from a KS-type statistic used for creating uniform confidence bands
`pscore.reg`	The results of propensity score regression, if specified
`probs`	The values for which the qte is computed
`type`	Takes the values "On the Treated" or "Population" to indicate whether the estimated QTE is for the treated group or for the entire population
`F.treated.t`	Distribution of treated outcomes for the treated group at period t
`F.untreated.t`	Distribution of untreated potential outcomes for the untreated group at period t
`F.treated.t.cf`	Counterfactual distribution of untreated potential outcomes for the treated group at period t
`F.treated.tmin1`	Distribution of treated outcomes for the treated group at period tmin1
`F.treated.tmin2`	Distribution of treated outcomes for the treated group at period tmin2
`F.treated.change.tmin1`	Distribution of the change in outcomes for the treated group between periods tmin1 and tmin2
`F.untreated.change.t`	Distribution of the change in outcomes for the untreated group between periods t and tmin1
`F.untreated.change.tmin1`	Distribution of the change in outcomes for the untreated group between periods tmin1 and tmin2
`F.untreated.tmin1`	Distribution of outcomes for the untreated group in period tmin1
`F.untreated.tmin2`	Distribution of outcomes for the untreated group in period tmin2
`condQ.treated.t`	Conditional quantiles for the treated group in period t
`condQ.treated.t.cf`	Counterfactual conditional quantiles for the treated group in period t
`eachIterList`	An optional list of the outcome of each bootstrap iteration
`inffunct`	The influence function for the treated group; used for inference when there are multiple periods and in the case with panel data. It is needed for computing covariance terms in the variance-covariance matrix.
`inffuncu`	The influence function for the untreated group

`QTEparams` *QTEparams*

Description

`QTEparams` is an object that contains all the parameters passed to QTE methods

Usage

```
QTEparams(
  formla,
  xformla = NULL,
  t = NULL,
  tmin1 = NULL,
  tmin2 = NULL,
  tname = NULL,
  data,
  panel = FALSE,
  w = NULL,
  idname = NULL,
  probs,
  alp = NULL,
  method = NULL,
  plot = NULL,
  se = NULL,
  iters = NULL,
  retEachIter = NULL,
  bootstrapiter = NULL,
  seedvec = NULL,
  pl = NULL,
  cores = NULL
)
```

Arguments

`formla`	Should be some y on treatment variable
`xformla`	a formula for the other covariates such as ~ x1 + x2
`t`	The last period (not always used)
`tmin1`	The last pre-treatment period (not always used)
`tmin2`	The 2nd to last pre-treatment period (not always used)
`tname`	The name of the column containing time periods (not always used)
`data`	The name of the data frame
`panel`	Whether or not the data is panel
`w`	Additional (usually sampling) weights

`idname`	The name of the id column used with panel data (not always used)
`probs`	Which quantiles to produce quantile treatment effects for
`alp`	The significance level
`method`	The method to compute the propensity score
`plot`	boolean for whether or not to plot qtes
`se`	boolean whether or not to compute standard errors
`iters`	The number of bootstrap iterations to use to compute standard errors
`retEachIter`	boolean whether or not to return the full results from each bootstrap iteration
`bootstrapiter`	Used internally for determining whether or not a call is part of computing standard errors via the bootstrap
`seedvec`	A seed to compute the same bootstrap standard errors each time the method is called (not always used)
`pl`	Boolean for whether or not computing bootstrap standard errrors in parallel
`cores`	The number of cores to use if computing standard errors in in parallel

---

`qtes2mat` *qtes2mat*

Description

Turn multiple qtes into a matrix for printing

Usage

```
qtes2mat(qteList, sset = NULL, se = TRUE, rnd = 3)
```

Arguments

`qteList`	a list of qte objects
`sset`	subset of qtes to keep
`se`	whether or not to include standard errors in the resulting matrix
`rnd`	how many disgits to round to

Value

matrix

`qteToTexreg` *diffQ*

Description

takes a single set of quantiles:
(not qtes as in diffquantiles) and returns the difference between particular ones

Usage

```
qteToTexreg(qteobj, tau = NULL, reportAte = T)
```

Arguments

`qteobj`	A QTE object
`tau`	Optional vector of taus to texreg results for
`reportAte`	Whether or not texreg the ATE (or ATT) as well

`setupData` *setupData*

Description

`setupData` sets up the data to use in each compute method in the QTE package

Usage

```
setupData(qteParams)
```

Arguments

`qteParams`	object holding the function parameters

Value

qteData object holding data to be used in QTE functions

`spatt` *spatt*

Description

`spatt` computes the Average Treatment Effect on the Treated (ATT) using the method of Abadie (2005)

Usage

```
spatt(
  formla,
  xformla = NULL,
  t,
  tmin1,
  tname,
  data,
  w = NULL,
  panel = FALSE,
  idname = NULL,
  iters = 100,
  alp = 0.05,
  method = "logit",
  plot = FALSE,
  se = TRUE,
  retEachIter = FALSE,
  seedvec = NULL,
  pl = FALSE,
  cores = 2
)
```

Arguments

`formla`	The formula y ~ d where y is the outcome and d is the treatment indicator (d should be binary)
`xformla`	A optional one sided formula for additional covariates that will be adjusted for. E.g ~ age + education. Additional covariates can also be passed by name using the x paramater.
`t`	The 3rd time period in the sample (this is the name of the column)
`tmin1`	The 2nd time period in the sample (this is the name of the column)
`tname`	The name of the column containing the time periods
`data`	The name of the data.frame that contains the data
`w`	an additional vector of sampling weights
`panel`	Boolean indicating whether the data is panel or repeated cross sections
`idname`	The individual (cross-sectional unit) id name

`iters`	The number of iterations to compute bootstrap standard errors. This is only used if se=TRUE
`alp`	The significance level used for constructing bootstrap confidence intervals
`method`	The method for estimating the propensity score when covariates are included
`plot`	Boolean whether or not the estimated QTET should be plotted
`se`	Boolean whether or not to compute standard errors
`retEachIter`	Boolean whether or not to return list of results from each iteration of the bootstrap procedure
`seedvec`	Optional value to set random seed; can possibly be used in conjunction with bootstrapping standard errors.
`pl`	boolean for whether or not to compute bootstrap error in parallel. Note that computing standard errors in parallel is a new feature and may not work at all on Windows.
`cores`	the number of cores to use if bootstrap standard errors are computed in parallel

Value

`QTE` object

References

Abadie (2005)

Examples

```
##load the data
data(lalonde)

## Run the panel.qtet method on the experimental data with no covariates
att1 <- spatt(re ~ treat, t=1978, tmin1=1975, tname="year",
 x=NULL, data=lalonde.psid.panel, idname="id", se=FALSE)
summary(att1)

## Run the panel.qtet method on the observational data with no covariates
```

`summary.BoundsObj` *Summary of BoundsObj*

Description

`summary.BoundsObj` is an object for holding bounds results

Usage

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

Arguments

`object`	A BoundsObj Object
`...`	Other params (for consistency as generic S3 method, but not used)

Value

summary.BoundsObj Object

`summary.QTE` *Summary*

Description

`summary.QTE` summarizes QTE objects

Usage

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

Arguments

`object`	A QTE Object
`...`	Other params (to work as generic method, but not used)

Index