

Package ‘mquantreg’

December 16, 2017

Type Package

Title An R package for M-quantile Models in Small Area Estimation

Version 0.1.0

Author Enno Tammena

Maintainer Enno Tammena <tammenae@hu-berlin.de>

Description This package provides functions to conduct linear M-quantile regression, obtain pseudo random effects from mixed M-quantile models and provides an M-quantile based approach to estimate linear and non-linear estimators in small area estimation. It is intended to be integrated in the R package emdi.

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 6.0.1

Imports data.table,
ggplot2,
MASS,
Hmisc

R topics documented:

mmqm	2
mq	3
mquantile	4
mq_sae	5
plot.mmqm	7
plot.mq	8
predict.mmqm	9
predict.mq	9
print.emdi	10
print.mmqm	11
print.mq	11
print.summary.emdi	12
summary.mmqm	12
summary.mq	13
Index	15

mmqm

*Pseudo Mixed M-quantile Models***Description**

Fit (pseudo) linear mixed M-quantile models based on the Huber proposal 2 loss function. Pseudo random effects are obtained based on the finding the average M-quantile regression plane in each domain. This procedure was introduced by *Chambers and Tzavidis (2006)*.

Usage

```
mmqm(formula, data, domains, grid = seq(0.001, 0.999, 0.05), ...)
```

Arguments

formula	a formula of the form $y \sim x_1 + x_2 + \dots$
data	a data frame containing the variables in the model.
domains	name of the variable in data that specifies the domain/group.
grid	numeric vector with the grid of tau-values. The argument defaults to <code>seq(0.001, 0.999, 0.05)</code> . The 0.5 M-quantile is always calculated.
...	additional arguments to be passed to the mq function, e.g. the tuning constant for <code>psi.huber</code> .

Value

An object of class "mmqm" which is a list with the following components:

area.coef	data.frame with the pseudo random effects per area.
all.coef	data.frame with the the MQ-coefficients corresponding to the grid.
area.tau	the average tau values corresponding to the areas.
area.coef	the results from the MQ-model, see mq .
domains	a vector of domains with length of data.

References

Breckling, J. and Chambers, R. (1988). *M-quantiles*. *Biometrika* 75, 761-71.

Chambers, R. and N. Tzavidis (2006): *M-quantile models for small area estimation*, *Biometrika*, 93, 255-268.

See Also

[mq](#)

Examples

```
mmqm(weight ~Time, data=ChickWeight, domains="Diet")
```

mq

*Linear M-quantile Regression***Description**

Fit linear M-quantile regression based on the Huber proposal 2 loss function. This function is a modification of the `rlm` function in the package [MASS](#)

Usage

```
mq(formula, data, k = 1.345, t = 0.5, w = rep(1, nrow(data)),
   case.weights = rep(1, nrow(data)), var.weights = rep(1, nrow(data)),
   init = "ls", maxit = 30, acc = 1e-04, test.vec = "resid",
   sparse = TRUE)
```

Arguments

<code>formula</code>	a formula of the form $y \sim x_1 + x_2 + \dots$
<code>data</code>	a data frame containing the variables in the model.
<code>k</code>	tuning constant used for Huber proposal 2 loss function.
<code>t</code>	vector of tau values specifying the M-quantiles.
<code>w</code>	a vector of prior weights for each case (for the IWLS procedure).
<code>case.weights</code>	(optional) initial down-weighting for each case.
<code>var.weights</code>	(optional) initial variable weights.
<code>init</code>	(optional) initial values for the coefficients or a method to find initial values or the result of a fit with a <code>coef</code> component. Known methods are "ls" (the default) for an initial least-squares fit using weights $w \cdot \text{weights}$, and "lts" for an unweighted least-trimmed squares fit with 200 samples.
<code>maxit</code>	the limit on the number of IWLS iterations.
<code>acc</code>	the accuracy for the stopping criterion.
<code>test.vec</code>	the stopping criterion is based on changes in this vector.
<code>sparse</code>	returns less results (to save RAM).

Value

An object of class "mq" which is a list with the following components:

<code>fitted.values</code>	matrix of fitted values per M-quantile.
<code>residuals</code>	matrix of residuals per M-quantile.
<code>tau.values</code>	tau values for which regressions were run.
<code>coefficients</code>	matrix of coefficients per M-quantile.
<code>call</code>	the formula that was called.
<code>iterations</code>	number of iterations until convergence per M-quantile.
<code>scale</code>	results from scale estimation per M-quantile.
<code>iterations</code>	number of iterations until convergence per M-quantile.
<code>x</code>	model-matrix of the independent variables.
<code>y</code>	vector of the dependent variable.
<code>classes</code>	classes of the independent variables (needed for plotting).
<code>r_2</code>	pseudo R-squared per M-quantile. Interpret only as an approximate figure.

References

- Breckling, J. and Chambers, R. (1988). *M-quantiles*. Biometrika 75, 761-71.
- Chambers, R. and N. Tzavidis (2006): *M-quantile models for small area estimation*, Biometrika, 93, 255-268.

See Also

[rlm](#)

Examples

```
mq(stack.loss ~ ., stackloss, t=c(0.25,0.5,0.75))
```

mquantile	<i>Sample M-quantiles</i>
-----------	---------------------------

Description

M-quantiles are fitted to univariate samples based on the Huber proposal 2 loss function for asymmetries (quantile-like) between 0 and 1.

Usage

```
mquantile(x, k = 1.375, tau = seq(0, 1, 0.25), dec = 4)
```

Arguments

- | | |
|-----|--|
| x | numeric vector of univariate observations. |
| k | the tuning parameter for the loss function. |
| tau | numeric vector of asymmetries between 0 and 1. |
| dec | number of decimals remaining after rounding the results. |

Value

returns a numeric vector of M-quantiles.

References

- P. J. Huber (1981). *Robust Statistics*. Wiley.
- Breckling, J. and Chambers, R. (1988). *M-quantiles*. Biometrika 75, 761-71.

See Also

[mq](#)
[quantile](#)

Examples

```
x = rnorm(1000)
mquantile(x,tau=c(0.01,0.02,0.05,0.1,0.2,0.5,0.8,0.9,0.95,0.98,0.99))
```

mq_sae

MQ-Models for disaggregated indicators

Description

Function `mq_sae` estimates indicators using the M-quantile approach. Point predictions of indicators are obtained by Monte-Carlo approximations as proposed by *Marchetti et al. (2012)*. Additionally, mean squared error (MSE) estimation can be conducted by using a non-parametric bootstrap approach. The procedure is based on pseudo random effects that are obtained by linear M-quantile regression as introduced by *Chambers and Tzavidis (2006)*.

Usage

```
mq_sae(fixed, pop_data, pop_domains, smp_data, smp_domains, L = 50,
       threshold = NULL, MSE = FALSE, B = 10, S = 20, seed = 123,
       parallel_mode = ifelse(grepl("windows", .Platform$OS.type), "socket",
                              "multicore"), cpus = 1, custom_indicator = NULL, na.rm = FALSE, ...)
```

Arguments

<code>fixed</code>	a two-sided linear formula object describing the fixed-effects part of the nested error linear regression model with the dependent variable on the left of a <code>~</code> operator and the explanatory variables on the right, separated by <code>+</code> operators. The argument corresponds to the argument <code>fixed</code> in function <code>mmqm</code> .
<code>pop_data</code>	a data frame that needs to comprise the variables named on the right of the <code>~</code> operator in <code>fixed</code> , i.e. the explanatory variables, and <code>pop_domains</code> .
<code>pop_domains</code>	a character string containing the name of a variable that indicates domains in the population data. The variable can be numeric or a factor but needs to be of the same class as the variable named in <code>smp_domains</code> .
<code>smp_data</code>	a data frame that needs to comprise all variables named in <code>fixed</code> and <code>smp_domains</code> .
<code>smp_domains</code>	a character string containing the name of a variable that indicates domains in the sample data. The variable can be numeric or a factor but needs to be of the same class as the variable named in <code>pop_domains</code> .
<code>L</code>	a number determining the number of Monte-Carlo simulations. Defaults to 50.
<code>threshold</code>	a number defining a threshold. Alternatively, a threshold may be defined as a function of <code>y</code> returning a numeric value. Such a function will be evaluated once for the point estimation and in each iteration of the non-parametric bootstrap. A threshold is needed for calculation e.g. of head count ratios and poverty gaps. The argument defaults to <code>NULL</code> . In this case the threshold is set to 60% of the median of the variable that is selected as dependent variable similarly to the At-risk-of-poverty rate used in the EU (see also <i>Social Protection Committee 2001</i>). However, any desired threshold can be chosen.
<code>MSE</code>	if <code>TRUE</code> , MSE estimates using a non-parametric bootstrap approach are calculated (see also <i>Marchetti et al. (2012)</i>). Defaults to <code>FALSE</code> .
<code>B</code>	a number determining the number of bootstrap populations in the non-parametric bootstrap approach (see also <i>Marchetti et al. (2012)</i>) used in the MSE estimation. Defaults to 10.

S	a number determining the number of samples taken from each bootstrap Population (see also <i>Marchetti et al. (2012)</i>) used in the MSE estimation. Defaults to 20.
seed	an integer to set the seed for the random number generator. For the usage of random number generation see details. If seed is set to NULL, seed is chosen randomly. Defaults to 123.
parallel_mode	modus of parallelization, defaults to an automatic selection of a suitable mode, depending on the operating system, if the number of CPUs is chosen higher than 1. For details see parallelStart
cpus	number determining the kernels that are used for the parallelization. Defaults to 1. For details see parallelStart
custom_indicator	a list of functions containing the indicators to be calculated additionally. Such functions must and must only depend on the target variable y and the threshold. Defaults to NULL.
na.rm	if TRUE, observations with NA values are deleted from the population and sample data. For the mq_sae procedure complete observations are required. Defaults to FALSE.
...	additional arguments to be passed to the mqmm function, e.g. the vector of values for the grid, or the tuning parameter for the Huber loss function.

Details

For Monte-Carlo approximations and in the non-parametric bootstrap approach random number generation is used. Thus, a seed is set by the argument seed.

The set of predefined indicators includes the mean, median, four further quantiles (10%, 25%, 75% and 90%), head count ratio, poverty gap, Gini coefficient and the quintile share ratio.

Value

An object of class "emdi" that provides estimators for regional disaggregated indicators and optionally corresponding MSE estimates. Generic functions such as [estimators](#), [print](#), [plot](#), and [summary](#) have methods that can be used to obtain further information. See [emdiObject](#) for descriptions of components of objects of class "emdi".

References

Chambers, R. and N. Tzavidis (2006): M-quantile models for small area estimation, *Biometrika*, 93, 255-268.

Marchetti, S., N. Tzavidis, and M. Pratesi (2012): Non-parametric bootstrap mean squared error estimation for -quantile estimators of small area averages, quantiles and poverty indicators, *Computational Statistics & Data Analysis*, 56, 2889-2902.

Social Protection Committee (2001). Report on indicators in the field of poverty and social exclusions, Technical Report, European Union.

See Also

[emdiObject](#), [mq](#), [mmqm](#), [estimators.emdi](#), [print.emdi](#), [plot.emdi](#), [summary.emdi](#)

Examples

```
## Not run:
# Loading data - population and sample data
require("emdi") #the datasets are in the emdi package
data("eusilcA_pop")
data("eusilcA_smp")

# Example 1: With default setting but na.rm=TRUE
mqemdi_model <- mq_sae(fixed = eqIncome ~ gender + eqsize + cash + self_empl +
  unempl_ben + age_ben + surv_ben + sick_ben + dis_ben + rent + fam_allow +
  house_allow + cap_inv + tax_adj, pop_data = eusilcA_pop,
  pop_domains = "district", smp_data = eusilcA_smp, smp_domains = "district",
  na.rm = TRUE)

# Example 2: With MSE, two additional indicators and MSE estimation
mqemdi_model <- mq_sae(fixed = eqIncome ~ gender + eqsize + cash +
  self_empl + unempl_ben + age_ben + surv_ben + sick_ben + dis_ben + rent +
  fam_allow + house_allow + cap_inv + tax_adj, pop_data = eusilcA_pop,
  pop_domains = "district", smp_data = eusilcA_smp, smp_domains = "district",
  L = 10, MSE = TRUE, B = 5, S=10, custom_indicator =
  list( my_max = function(y, threshold){max(y)},
  my_min = function(y, threshold){min(y)}), na.rm = TRUE, cpus = 1)

## End(Not run)
```

plot.mmqm

Plot visualization for an mmqm Object

Description

Creates two types of plots of pairwise combinations of the dependent and independent variable that visualize the pseudo random effects. A list of plots is returned, one for each numeric independent variable which is plotted against the dependent variable. Factor variables are automatically excluded.

Usage

```
plot.mmqm(object, type = "domain", legend = TRUE)
```

Arguments

object	mmqm object, a result of mmqm .
type	either "domain" or "overall". Defaults to domain.
legend	either TRUE or FALSE. Set FALSE when many domains make the plot unreadable. Defaults to true.
	type="overall" plots the grid (as specified in mmqm) of M-quantile regression lines over the data. type="domain" plots the regression lines that visualize the pseudo random effects.

Value

Returns a list of ggplot2 objects.

References

- Breckling, J. and Chambers, R. (1988). *M-quantiles*. Biometrika 75, 761-71.
- Chambers, R. and N. Tzavidis (2006): *M-quantile models for small area estimation*, Biometrika, 93, 255-268.

See Also

[mmqm](#)

Examples

```
m_model <- mmqm(weight ~Time, data=ChickWeight, domains="Diet")
plot(m_model, type="overall")
plot(m_model, type="domain")
```

plot.mq

Residual plot for an mq object.

Description

Creates a plot of residuals versus fitted values.

Usage

```
plot.mq(object)
```

Arguments

object mq object, a result of [mq](#).

Value

Returns a list of one plot per tau.

References

- Breckling, J. and Chambers, R. (1988). *M-quantiles*. Biometrika 75, 761-71.
- Chambers, R. and N. Tzavidis (2006): *M-quantile models for small area estimation*, Biometrika, 93, 255-268.

See Also

[mq](#)

Examples

```
plot(mq(stack.loss ~ ., stackloss, t=c(0.25,0.5,0.75)))
```

predict.mmqm	<i>Predict method for Pseudo mixed M-quantile Model Fits</i>
--------------	--

Description

Predicted values for class "mmqm".

Usage

```
predict.mmqm(object, newdata, na.action = na.pass, ...)
```

Arguments

object	mq object, a result of mq .
newdata	An optional data frame in which to look for variables with which to predict. If omitted, the fitted values are used.

Value

Returns a vector of predicted values.

References

Breckling, J. and Chambers, R. (1988). *M-quantiles*. Biometrika 75, 761-71.

Chambers, R. and N. Tzavidis (2006): *M-quantile models for small area estimation*, Biometrika, 93, 255-268.

See Also

[mmqm](#)

Examples

```
# use first 200 observations in chickweight data to predict remaining observations:
mmqm_model <- mmqm(weight ~Time, data=ChickWeight[1:200,], domains="Diet")
predict(mmqm_model, ChickWeight[-c(1:200),])
```

predict.mq	<i>Predict method for Linear M-quantile Model Fits</i>
------------	--

Description

Predicted values for class "mq".

Usage

```
predict.mq(object, newdata, na.action = na.pass, ...)
```

Arguments

object	mq object, a result of mq .
newdata	An optional data frame in which to look for variables with which to predict. If omitted, the fitted values are used.

Value

The function calculates the predicted values.

References

Breckling, J. and Chambers, R. (1988): "M-quantiles", *Biometrika*, 75, 761–71.
 Chambers, R. and N. Tzavidis (2006): "M-quantile models for small area estimation", *Biometrika*, 93, 255–268.

See Also

[mq](#)

Examples

```
##Predictions
x=rnorm(15)
e=rnorm(15)
y=x+e
df <- data.frame(x=x, y=y)
predict(mq(y ~ x, data=df))
new <- data.frame(x = seq(-3, 3, 0.5))
predict(mq(y ~ x, data=df), new)
```

print.emdi	<i>Prints an emdiObject</i>
------------	-----------------------------

Description

Basic information of an emdi object is printed.

Usage

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

Arguments

x	an x of type "emdi", representing point and MSE estimates obtained by direct estimation (see also direct) or Empirical Best Prediction (see also ebp). or M-quantile Small Area Models (see also mq_sae). # <– NEW
...	optional arguments passed to print.default .

See Also

[emdiObject](#), [ebp](#), [mq_sae](#) # <– NEW

`print.mmqm`*Printing Pseudo Linear Mixed M-quantile Model Fits*

Description

Printing method for class "mmqm".

Usage

```
## S3 method for class 'mmqm'  
print(object)
```

Arguments

`object` mmqm object, a result of `mmqm`.

Value

The function determines the printing behaviour for the `mmqm` object.

See Also

`mmqm`

Examples

```
print(mmqm(weight ~Time, data=ChickWeight, domains="Diet"))  
mmqm(weight ~Time, data=ChickWeight, domains="Diet") #equivalent
```

`print.mq`*Printing Linear M-quantile Model Fits*

Description

Printing method for class "mq".

Usage

```
## S3 method for class 'mq'  
print(object)
```

Arguments

`object` mq object, a result of `mq`.

Value

The function determines the printing behaviour for the `mq` object.

References

- Breckling, J. and Chambers, R. (1988). *M-quantiles*. Biometrika 75, 761-71.
- Chambers, R. and N. Tzavidis (2006): *M-quantile models for small area estimation*, Biometrika, 93, 255-268.

See Also

[mq](#)

Examples

```
print(mq(stack.loss ~ ., stackloss, t=c(0.25,0.5,0.75)))
mq(stack.loss ~ ., stackloss, t=c(0.25,0.5,0.75)) #equivalent
```

```
print.summary.emdi      Prints a summary.emdi object
```

Description

The elements described in summary.emdi are printed.

Usage

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

Arguments

x	an object of type "summary.emdi", generally resulting from applying summary to an object of type "emdi"
...	optional arguments passed to print.default; see the documentation on that method functions

See Also

[summary.emdi](#)

```
summary.mmqm      Summarizing Linear Mixed M-quantile Model Fits
```

Description

Summary method for class "mmqm".

Usage

```
## S3 method for class 'mmqm'
summary(object)
```

Arguments

object mmqm object, a result of [mmqm](#).

Value

The function determines the summary behaviour for the mmqm object.

References

Breckling, J. and Chambers, R. (1988): M-quantiles, *Biometrika*, 75, 761–71.

Chambers, R. and N. Tzavidis (2006): M-quantile models for small area estimation, *Biometrika*, 93, 255–268.

See Also

[mmqm](#)

Examples

```
mqmm_mod <- mmqm(weight ~Time, data=ChickWeight, domains="Diet")
summary(mqmm_mod)
```

summary.mq

Summarizing Linear M-quantile Model Fits

Description

Summary method for class "mq".

Usage

```
## S3 method for class 'mq'
summary(object)
```

Arguments

object mq object, a result of [mq](#).

Value

The function summary.mq computes a list of summary statistics of the fitted M-quantile model. Coefficients are returned per M-quantile.

References

Breckling, J. and Chambers, R. (1988). *M-quantiles*. *Biometrika* 75, 761-71.

Chambers, R. and N. Tzavidis (2006): *M-quantile models for small area estimation*, *Biometrika*, 93, 255-268.

See Also

[mq](#)

Examples

```
summary(mq(stack.loss ~ ., stackloss, t=c(0.25,0.5,0.75)))
```

Index

direct, [10](#)

ebp, [10](#)

emdiObject, [6](#), [10](#)

estimators, [6](#)

estimators.emdi, [6](#)

MASS, [3](#)

mmqm, [2](#), [5–9](#), [11](#), [13](#)

mq, [2](#), [3](#), [4](#), [6](#), [8–13](#)

mq_sae, [5](#), [10](#)

mqmm, [6](#)

mquantile, [4](#)

parallelStart, [6](#)

plot, [6](#)

plot.emdi, [6](#)

plot.mmqm, [7](#)

plot.mq, [8](#)

predict.mmqm, [9](#)

predict.mq, [9](#)

print, [6](#)

print.default, [10](#)

print.emdi, [6](#), [10](#)

print.mmqm, [11](#)

print.mq, [11](#)

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

quantile, [4](#)

rlm, [3](#), [4](#)

summary, [6](#)

summary.emdi, [6](#), [12](#)

summary.mmqm, [12](#)

summary.mq, [13](#)