M-quantile models for robust small area estimation of non-linear indicators

With development of an R-package and an application to poverty estimation

Master's Thesis submitted

to

Prof. Dr. Timo Schmid
Prof. Dr. Ulrich Rendtel

Freie Universität Berlin
Fachbereich Wirtschaftswissenschaft
Institut für Statistik und Ökonometrie

by

Enno Tammena
Graefestr. 58
10967 Berlin
tammenae@hu-berlin.de
Matriculation Nr.
572575

in partial fulfillment of the requirements
for the degree of
Master of Science
in Statistics
Berlin, December 20th, 2017

Acknowledgement

I would like to thank my supervisors Prof. Dr. Timo Schmid and Prof. Dr. Ulrich Rendtel. I am particularly grateful that Prof. Dr. Timo Schmid proposed this challenging but solvable topic for my Master's thesis. My special thanks goes to my colleague Felix Skarke for excellent econometrics tutoring and moral support in the first semester and to Alexander Gattig for proposing me for a scholarship at the "Studienstiftung des Deutschen Volkes", to which my gratitude goes for funding my studies. Last but not least I want to thank my family and friends.

Contents

List of Abbreviations

BLUE	Best linear unbiased estimator
BLUP	Best linear unbiased predictor
BULM	Basic unit level model
EBLUP	Empirical best linear unbiased predictor
EBP	Empirical best prediction
ELL	Elbers-Lanjouw-Lanjouw
EU-SILC	European Union Statistics on Income and Living Conditions
FGT	Foster-Greer-Thorbecke
HCR	Headcount ratio
HP2	Huber proposal 2
IWLS	Iteratively reweighted least squares
MQ	M-quantile
MQ (SAE)	M-quantile small area approach
MSE	Mean squared error
OLS	Ordinary least squares
PG	Poverty gap
QQ	Quantile-quantile
RB	Relative bias
RMSE	Root mean squared error
RRMSE	Relative root mean squared error
S.A.M.P.L.E	Small Area Methods for Poverty and Living Condition Estimates
SAE	Small area estimation
SDG	Sustainable Development Goals
SSR	Sum of squared residuals
SST	Total sum of squares

Mathematical Symbols

y	Vector of the dependent variable $(n \times 1)$
X	Matrix independent/auxiliary variables $(n \times p)$
x_i^T	Vector of independent/auxiliary variables for the i-th unit $(1 \times p)$
W	Matrix of weights $(n \times n)$
β	Vector of regression coefficients $(p \times 1)$
β_{OLS}	Vector of OLS regression coefficients
β_q	Vector of qth quantile regression coefficients
β_τ	Vector of τth M-quantile regression coefficients
Z	Design matrix for random effects $(n \times u)$
ϑ	Vector of random effects $(u \times 1)$
$\beta_{\bar{\tau}_j}$	Vector of pseudo random effects of area j $(p \times 1)$
ϵ	Vector of errors $(n \times 1)$
e	Vector of residuals $(n \times 1)$
σ^2_ϑ	Variance of the random effect
σ^2_e	Variance of the error term
γ	Correlation of error terms
$\hat{(\cdot)}$	Estimator
θ	Location parameter
μ_j	Linear indicator of interest in area j
$\hat{\mu}^{EBLUP}$	Empirical best linear unbiased predictor of μ
$\hat{\mu}^{PIMQ}$	Plug-in estimator of μ based on M-quantiles
ν_j	Non-linear indicator of interest in area j
$\hat{\nu}^{BP}$	Best predictor of ν
$\hat{\nu}^{EBP}$	Empirical best predictor of ν
$\hat{\nu}^{MQ}$	Estimator of ν based on M-quantiles
$F(t)$	Distribution function
$\hat{F}^{CD}(t)$	Chambers-Dunstan estimator of the distribution function
$F_\alpha^{FGT}(z)$	FGT poverty measure with power α and threshold z
x	Random variable
x_i	Value of x
$x_i^{(S)}$	Value of x_i in iteration S of an iterative algorithm
x_i^*	Value of x_i in a superpopulation

$\bar{x}$	Arithmetic mean of x
Ω	Finite population
i	Unit
j	Domain/area
s_j	Set of sampled units in area j
r_j	Set of non-sampled units in area j
n_j	Sample size of area j
N_j	Population size of area j
d	Number of domains
$sgn(\cdot)$	Signum function
$I(\cdot)$	Indicator function
$\rho(\cdot)$	Objective or loss function
$\psi(\cdot)$	Influence or score function
$w(\cdot)$	Weight function
k	Tuning constant in the Huber proposal 2 loss function
δ	Interval between lines or planes on a grid
$s_{mad}(x)$	Robust scale estimator based on mean absolute deviation
$E(x)$	Expected value of x
$MQ_\tau(x)$	τth M-quantile of x
$Q_q(x)$	qth quantile of x
$U[a,b]$	Uniform distribution in interval a,b
$N(\mu,\sigma)$	Normal distribution with expectation μ and variance σ
$\chi^2(df)$	Chi-squared distribution with df degrees of freedom

List of Figures

List of Tables

1 Introduction

> "Every journey has a beginning and an end. Plotting that journey and establishing key milestones along the way requires accessible, timely and reliable disaggregated data. The data requirements for the global indicators are almost as unprecedented as the SDGs [Sustainable Development Goals, A/N] themselves and constitute a tremendous challenge to all countries."
>
> (United Nations, 2016, p.3)

In the year 2015 the United Nations established the Sustainable Development Goals, which range from the the eradication of poverty[1] over the reduction of inequality to environmental goals. As the introductory quote shows, the availability of disaggregated data is considered to be a premise for their achievement and a major challenge to all countries.

Indeed, even in developed countries with sophisticated statistical institutions, the availability of survey data is often not sufficient for the direct estimation of disaggregated indicators at lower regional levels with the desired precision, because the number of available observations at that regional level is too small. However, because poverty and inequality will seldom show an even spatial distribution, only considering aggregate poverty indicators at country or state level might leave smaller regions with high poverty or inequality undetected, and therefore hinder the implementation of effective policies.

The use of small area methods to overcome such data limitations for the estimation of means and totals is well known since the last century (cf. e.g. Battese et al. (1988), Fay and Herriot (1979)). However, the small area estimation of poverty and inequality indicators is more difficult, because they are commonly measured with non-linear indicators. With the ELL method of Elbers et al. (2003) and the empirical best prediction (EBP) developed by Molina and Rao (2010) successful attempts towards small area estimation of non-linear poverty estimators have been made. However, the ELL method relies on very rigorous data requirements, and particulary the EBP is based on Gaussian assumptions on the distribution of error terms (cf. Haslett, 2016). Aditionally, when working with income data, researchers will often be challenged with outlying and influential data points. A proposal to outlier robust small area estimation without distributional assumptions and less requirements on the available data based on M-quantile models was therefore made about 10 years ago by

[1] There are different concepts for poverty and inequality, where some are only based on income, and others define poverty as a multidimensional concept based on several indicators like deprivation in food or access to water (cf. Betti and Lemmi, 2013). In this thesis, only income type measures of poverty and inequality are considered.

Chambers and Tzavidis (2006). Nonetheless, while there are user friendly implementations of the EBP method available in R today, this is not the case for M-quantile methods. Possibly, this limitation is a reason, why there has been not much dedication to the comparison of the EBP and M-quantile approach under different scenarios yet, since there are no published papers regarding this question available.

The aim of this thesis is therefore to contribute to the scientific progress in two ways: Firstly, to provide a user friendly implementation of M-quantile models in context of small area estimation, including a non-parametric bootstrap procedure to estimate the variation of the results. Secondly, to evaluate based on Monte-Carlo simulation studies the performance of the M-quantile approach compared to the EBP method regarding the small area estimation of poverty measures and other non-linear indicators.

The thesis is organized as follows: It begins with the theoretical foundations of the later implemented methods. For this, a comprehensive description of the M-estimation framework, and its generalization to M-quantiles and M-quantile regression is given. These sections are followed by a discussion of random effects and it is shown how so-called pseudo random effects can be obtained by M-quantile models. In section 3 a short overview of the small area framework is given, and the use of random effects as well as pseudo random effects in small area models is presented. Of particular importance is then the estimation of non-linear indicators, where a short description of the EBP is followed by a detailed exposition of small area estimation of non-linear indicators based on M-quantiles as well as a corresponding bootstrap procedure for mean squared error (MSE) estimation.

The implementation of these methods in R is then presented in section 4, followed by the setup and results of various Monte-Carlo simulations in section 5. These include a replication of the results of Marchetti et al. (2012) to show that the new implementation actually produces the correct results. Additional simulations then compare the new method to the EBP in a variety of scenarios, where point estimation as well as mean squared error estimation are considered.

The remainder of this thesis is used for an application based on the Austrian EU-SILC data, which gives an example how poverty mapping can be conducted based on the M-quantile approach (section 6), followed by a summary and concluding remarks in section 7.

Provided along with this thesis are also some results in digital form, which include the developed R-Package `mquantreg`, a users manual and the simulation results as well as routines for their analysis.

2 Robust Estimation with Loss Functions

> "It is interesting to look back to [...] Gauss and his theory of least squares. Gauss was fully aware that his main reason for assuming a [...] quadratic loss function was mathematical, i.e., computational, convenience."
>
> (Huber, 1964, p.73)

At the core of this Master thesis lies a theory of robust estimation proposed by Huber (1964) which is called *M-estimation*. In the following sections, this theory is explained and it is shown how well known estimators like the *mean* and the *median* can be regarded as special cases in this framework. Building on this, it is shown how M-estimation can be generalized to find estimates for other non-central points of a distribution, which are called M-quantiles, and include well known concepts like *quantiles* and *expectiles* as limiting solutions. In section 2.2 it is shown how M-estimation can be used in a regression context, finally leading to *M-quantile regression*. This method is then the foundation for a non-parametric procedure to obtain pseudo random effects described in section 2.4.

2.1 Location Parameter of a Distribution

The following subsection begins with the most simple case: the univariate and unconditional estimation of the central point of a distribution. In the next subsection, non-central points are discussed.

2.1.1 Symmetric Loss Functions: Mean, Median and M-estimation

Huber's theory of robust estimation is motivated by the question, how the central point for a variable can be robustly estimated if its true distribution function F differs slightly from an assumed normal distribution. For instance, when the true distribution is a contaminated normal distribution. In 1964, Huber focuses on estimators, henceforth called "M-estimators", that minimize certain corresponding loss functions ρ:

Definition 1. *M-estimator*

For a random variable x with distribution function $F(x)$ an M-estimator is the solution θ that minimizes

$$\min_{\theta} \int \rho(x-\theta)dF(x), \tag{1}$$

where ρ is a non-constant function (cf. Huber, 1964, p. 74), that is called the loss function or the objective function. In a finite sample $x_1, \ldots, x_n$ an M-estimator is obtained by solving

$$\min_{\theta} \sum_{i=1}^{N} \rho(x_i - \theta). \tag{2}$$

For the estimation of the central point of a distribution ρ has to be symmetric around 0.

Remark 1. *While it is always possible to find θ using the loss function, it will be more convenient when ρ is differentiable (and convex), as the solution to (2) can be found with the influence function ψ:*

$$\min_{\theta} \sum_{i=1}^{N} \psi(x_i - \theta) \stackrel{!}{=} 0, \tag{3}$$

where

$$\psi(x_i - \theta) = \frac{d\rho(x_i - \theta)}{d\theta}. \tag{4}$$

In addition, some estimators need to be calculated with an iterative re-weighting algorithm, for which the weight function w is of particular importance. Let $e_i = x_i - \theta$, it is defined as

$$w(e_i) = \frac{\psi(e_i)}{e_i}, \tag{5}$$

and shows, how residuals are weighed based on the distance to zero (cf. Fox and Weisberg, 2010). However, for non-differentiable choices of ρ the computation is more complex and some numerical procedure can be required.

Two well known estimators that fit in the M-estimation framework are the mean and the median. For example, the mean minimizes a quadratic loss function:

Example 1. The mean and the quadratic loss function

$$\begin{aligned} & min \sum_{i=1}^{N} \frac{1}{2}(x_i - \theta)^2 \\ \Rightarrow & \sum_{i=1}^{N} (x_i - \theta) \stackrel{!}{=} 0 \\ \Rightarrow & \hat{\theta}_{mean} = \frac{1}{n} \sum_{i=1}^{N} x_i. \end{aligned}$$

The median corresponds to an absolute loss function (see table 2). Because this function is not differentiable, its computation requires some numerical procedure.

The most important role in this thesis plays however a loss function which can be considered as an intermediate between the quadratic and absolute loss function:

Definition 2. *The Huber proposal 2 loss function*

$$\rho_{HP2}(e) = \begin{cases} \frac{1}{2}e^2 & for|e| \leq k \\ k|e| - \frac{1}{2}k^2 & for|e| > k, \end{cases} \tag{6}$$

where $k \in \mathbb{R}^+$ is a tuning parameter.

The estimator that corresponds to the HP2 loss function is called Huber M-estimator in this thesis.

For the right choice of k (which is in practice unknown), the Huber M-estimator can be shown to be the most robust[2] M-estimator (cf. Huber, 1964).

The HP2 loss function includes the quadratic and absolute loss function two limiting cases: For $k \to \infty$, the Huber loss function approaches the quadratic loss function, because $|e|$ will always be smaller than k. For $k \to 0$ it approaches the (compressed) loss function of the median, since k^2 approaches 0 more quickly than k (see figure 1). Therefore, mean and median are limiting cases of the Huber M-estimator.

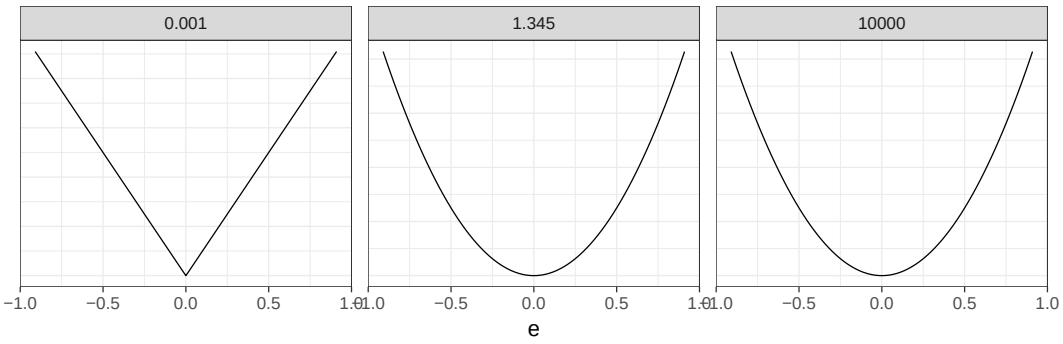

Figure 1: HP2 loss function for different choices of k

For choices of k in between, the HP2 loss function will be situated somewhere in between both functions and take the shape of the quadratic loss function for values close to zero. Depending on k, it then transitions to the shape of the absolute loss function. The choice of k can therefore be seen as a trade-off between (asymptotic) efficiency and robustness (cf. Tzavidis et al., 2016, p. 429), implying that the Huber M-estimator can be more efficient than the median, but also more robust than the mean. In practical applications, k is however

[2]Huber defines robustness as the supremum of the asymptotic variance, which he claims is a good measure of robustness also in finite samples (cf. Huber, 1964).

often set to 1.345. This goes back to Holland and Welsch (1977), who argued that if the underlying distribution is in fact the normal distribution, the loss in asymptotic efficiency compared to the use of ρ_{L2} is only around 5%, yet a certain robustness is achieved.

To actually calculate the Huber M-estimate for some sample data, an iterative procedure is required, because the weights and the errors stand in an recursive relation to each other. Once the weights are known, the Huber M-estimator can be calculated as a weighted mean. The procedure is similar to the IWLS algorithm explained in section 2.2.2 and here only displayed in abbreviated form:

Algorithm/Procedure 1. *Calculation of Huber M-estimate for a variable x*

1. *initial guess, e.g.* $\hat{\theta}^{(0)} = \bar{x}$
2. *iterate until convergence*
   (a) *at iteration s, calculate residuals* $e^{(s-1)} = x - \hat{\theta}^{(s-1)}$
   (b) *calculate weights* $w_i^{(s-1)} = w_{HP2}(e_i^{(s-1)})$ *for* $i = 1, \ldots, n$
   (c) *calculate a weighted mean*

$$\hat{\theta}^{HP2,(s)} = \frac{\sum_i^N w_i^{(s-1)} x_i}{\sum_i^N w_i^{(s-1)}}.$$

Table 2 shows the loss and influence function of the so far discussed M-estimators. In figure 2, the respective functions are plotted.

Remark 2. *Other loss functions for M-estimation are of course possible, notably the bi-square and Hampel function. However, they are only used in one of the papers considered in this thesis where they are found to be computationally disadvantageous (cf. Chambers and Tzavidis, 2006). Therefore, the focus of this thesis lies on the HP2 loss function.*

	Mean	Median	Huber M-estimator
Loss Function ρ	$\rho_{L2}(e) = \frac{1}{2}e^2$	$\rho_{L1}(e) = \lvert e \rvert$	$\rho_{HP2}(e) = \begin{cases} \frac{1}{2}e^2 & for \lvert e \rvert \leq k \\ k\lvert e \rvert - \frac{1}{2}k^2 & for \lvert e \rvert > k \end{cases}$
Influence Function ψ	$\psi_{L2} = u$	$\psi_{L1} = sgn(e)$	$\psi_{HP2}(e) = \begin{cases} e & for \lvert e \rvert \leq k \\ ksgn(e) & for \lvert e \rvert > k \end{cases}$
Weight Function w	$w_{L2} = 1$		$w_{HP2}(e) = \begin{cases} 1 & for \lvert e \rvert \leq k \\ \frac{k}{\lvert e \rvert} & for \lvert e \rvert > k \end{cases}$

Table 2: Important M-Estimators and corresponding ρ, ψ, w- functions. Note: Influence function ψ_{L1} per definition (cf. Rao and Molina, 2015, p. 200), as ρ_{L1} is not differentiable.

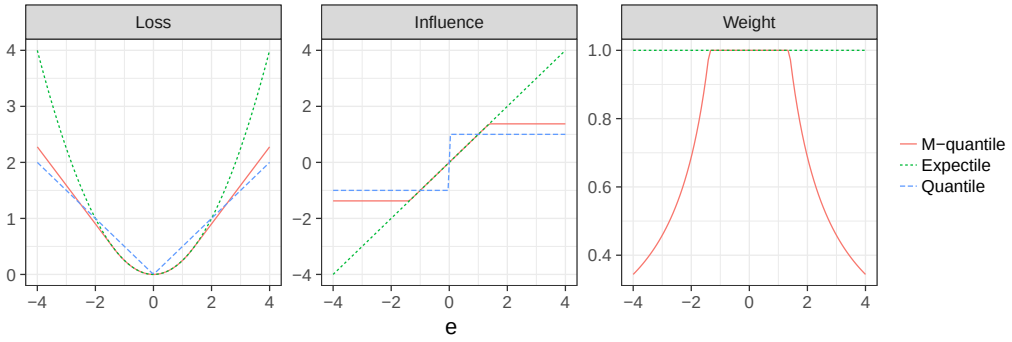

Figure 2: Loss, influence and weight function of important M-estimators

2.1.2 Asymmetric Loss Functions: Expectiles, Quantiles and M-quantiles

With the Huber loss function, a robust estimate of the central point of a distribution can be retrieved. However, the central point only provides an incomplete picture of a distribution. To characterize non-central parts of a distribution one can rely on quantiles, expectiles and M-quantiles. These quantities can also be defined in the manner of M-estimation, but now with loss functions, that are in general asymmetric around zero.

Definition 3. *M-quantiles under the HP2 loss function*

For a random variable x with distribution function $F(x)$ the τth M-quantile is the solution θ that minimizes

$$\min_{\theta} \int \rho_{MQ}(x - \theta) dF(x), \tag{7}$$

where

$$\rho_{MQ}(e) = \begin{cases} \rho_{HP2}(e)(1-\tau) & for\, e \leq 0 \\ \rho_{HP2}(e)\tau & for\, e > 0 \end{cases}$$

$$\rho_{HP2}(e) = \begin{cases} \frac{1}{2}e^2 & for |e| \leq k \\ (k|u| - \frac{1}{2}k^2) & for |e| > k. \end{cases}$$

In a finite sample $x_1, \ldots, x_n$ an M-quantile is obtained by solving

$$\min_{\theta} \sum_{i=1}^{N} \rho_{MQ}(x_i - \theta). \tag{8}$$

Remark 3. *The relationship between ρ, ψ and w also hold in the asymmetric case.*

Remark 4. *Note that if instead of ρ_{HP2} the absolute function ρ_{L1} is used in* (7)*, quantiles are obtained, and for the quadratic loss function ρ_{L2} a generalization of the mean called expectiles are retrieved. Alternatively, these quantities can be approximated with the tuning constant k in ρ_{HP2}, as the solution approaches expectiles when $k \to \infty$ and quantiles for $k \to 0$.*

To visualize the difference between quantiles, expectiles and M-quantiles the respective loss, influence and weight functions are plotted for $\tau \in \{0.25, 0.5, 0.75\}$ in figure 3. For $\tau = 0.5$, the curves show the special cases leading to the mean, median and Huber M-estimator. For other choices of τ, the loss functions weigh for $\tau < 0.5$ positive residuals less than negative residuals, while the opposite is true for $\tau > 0.5$.

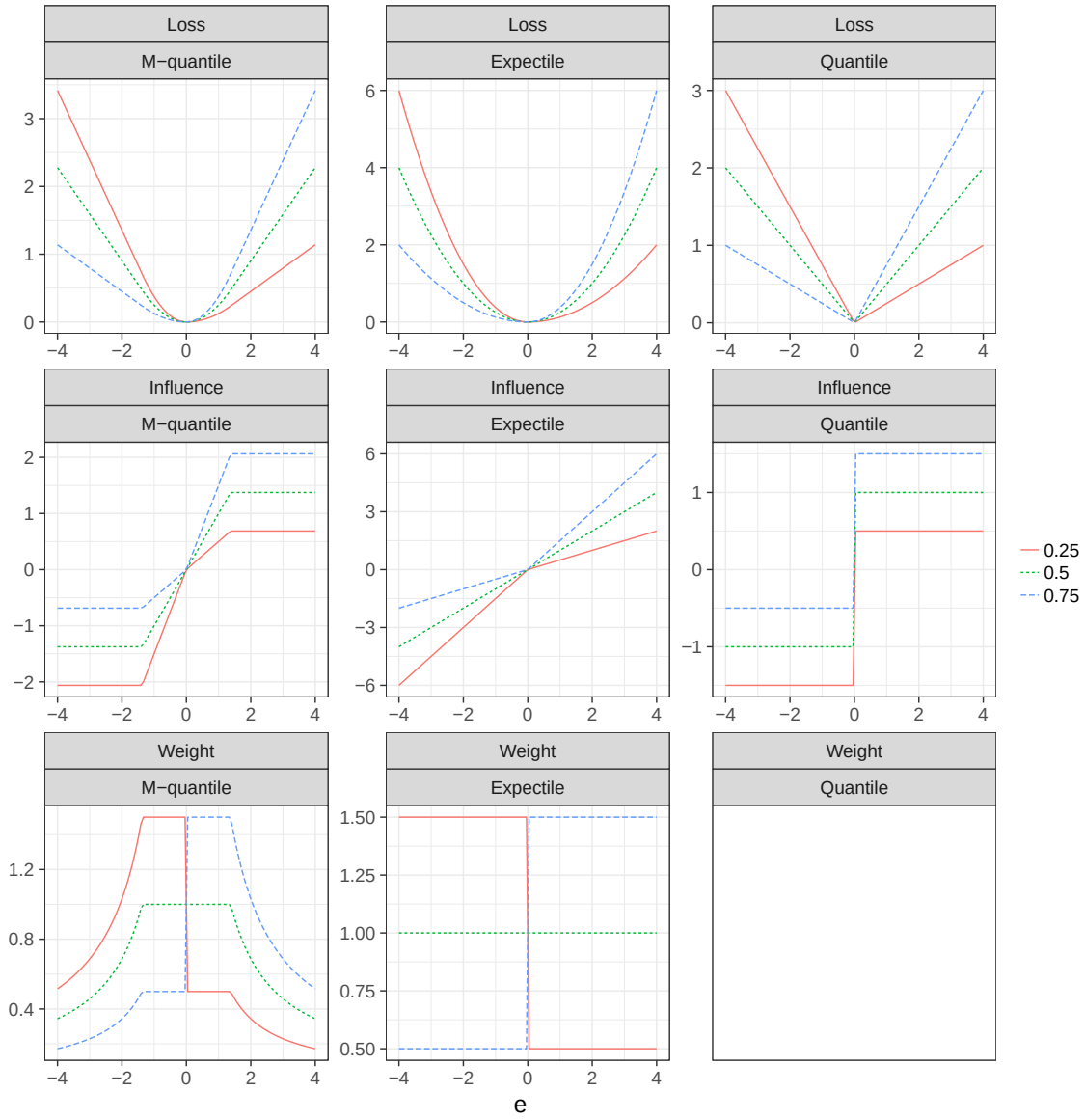

Figure 3: Asymmetric loss, influence and weight function of important M-estimators

Regarding the interpretation, there is only an intuitive interpretation for quantiles, where it holds in a finite sample that a certain percentage of values lies below the value of the qth quantile. Expectiles and M-quantiles only have an intuitive interpretation for $\tau = 0.5$ (the mean or the Huber estimator of the central point).

To provide some intuition for the three quantities, the values for $\tau \in \{0, 0.1, ...1\}$ are calculated for the examples of $X \sim N(0, 1)$ and $X \sim exp(1)$ plotted in figure 4. For the normal distribution, M-quantiles are located between expectiles and quantiles, while the values of quantiles, expectiles and M-quantiles are close to each and coincide for $\tau = 0.5$.

For the exponential distribution, such behavior cannot be observed, while the values are still close to each other.

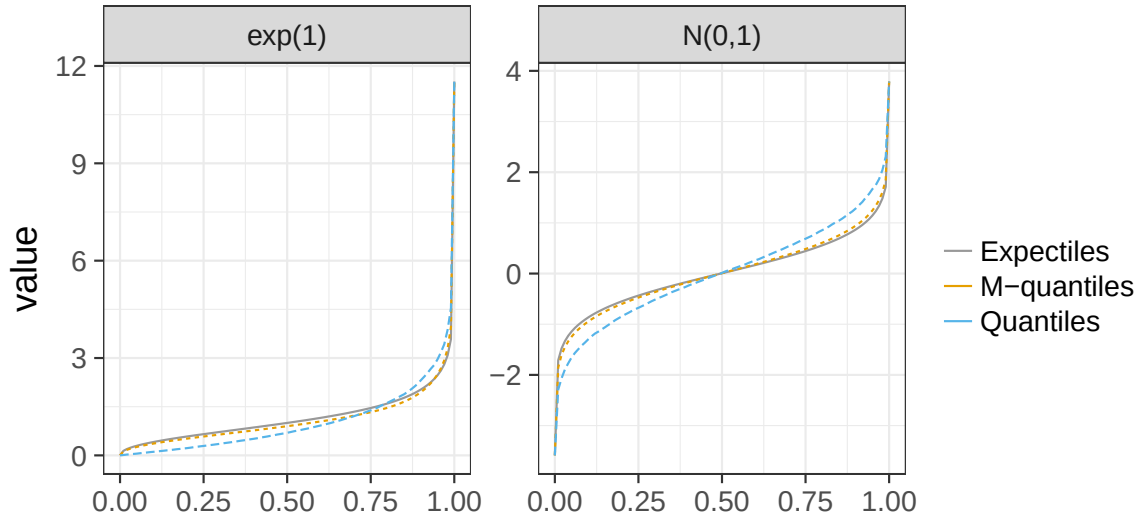

Figure 4: Quantiles, expectiles and M-quantiles for $exp(1)$ and $N(0,1)$

2.2 Conditional Location Parameter of a Distribution

Until now only *unconditional* location parameters are considered. However, with M-estimation also *conditional* location parameters for y given X can be found which are like in the unconditional case defined by the corresponding loss function. Therefore, well known regression problems can be formulated in the way of M-estimation, based on symmetric and asymmetric loss functions.

2.2.1 Regression based on Symmetric Loss Functions

Consider the following model

$$y = X\beta + \epsilon, \tag{9}$$

where ϵ denotes a random error term with $E[\epsilon] = 0$. An estimator of β is for example given by the ordinary least squares (OLS) estimator. It is well known, that in OLS regression the sum of squared errors is minimized, which is in a sample the squared distance between the regression line (or plane, in the multivariate case) and the observations. Speaking in terms of M-estimation, a quadratic loss function is minimized, as the following example shows:

Example 2. Regression with the quadratic loss function

Consider model (9). Let $\theta = X\beta$. In the notation of M-estimation, the linear regression problem takes the form

$$\min_{\beta} \sum_{i=1}^{N} \rho_{L2}(y_i - \theta). \tag{10}$$

Because ρ is differentiable, an estimator for β can then be obtained by exploiting the relationship between the objective and the influence function. Let $e_i = y_i - x_i^T\beta$, the estimator for β is derived with:

$$\min_{\beta}(e_i)^2 \Rightarrow \frac{d}{d\beta}\sum(e_i)^2 \Rightarrow \frac{d}{d\beta}ee^T \Rightarrow X^T(y - X\beta) \stackrel{!}{=} 0,$$

which obviously is just the OLS estimator $\hat{\beta}_{OLS} = (X^TX)^{-1}X^Ty$.

It is now easy to see, why OLS regression models the conditional *mean* of y given X:

$$E[y|X] = X\beta_{OLS}, \tag{11}$$

because the minimization problem (10) is just that of the mean (cf. equation (2)) with now $\theta = X\beta$.

While OLS regression is easy to compute and interpret, few outliers at the tail of the conditional distribution of $y|X$ might alter the regression line considerable. Like the mean, OLS regression is not very *robust*.

However, in the mindset of M-estimation it follows immediately, that one is not limited to the quadratic loss function, but is able to use other valid loss functions like the HP2 loss function to retrieve possibly more robust conditional location parameters. For example, with the absolute loss function ρ_{L1} a median regression and with ρ_{HP2} a robust Huber regression can be estimated.

Nonetheless, while choosing a robust procedure can be beneficial in the presence of outliers, modeling only the central point of the conditional distribution might provide an incomplete picture of the relationship between X and y. For instance, "the effect of a job- training program on the length of participants' current unemployment spell might be to lengthen the shortest spells while dramatically reducing the probability of very long spells" (Koenker, 2000, p.5).

An proposal to overcome this shortcoming is given with quantile regression, and alternatives include expectile and M-quantile regression. These forms of regression have in common that they are based on asymmetric loss functions.

2.2.2 Regression based on Asymmetric Loss Functions

Koenker and Bassett (1978) introduced the concept of quantile regression: "Least squares

estimation of mean regression models asks the question, 'How does the conditional mean of Y depend on the covariates X?' Quantile regression asks this question at each quantile of the conditional distribution enabling one to obtain a more complete description of how the conditional distribution of Y given $X = x$ depends on x." (Koenker, 2000, p.5).

Quantile regression can be considered as a generalization of median regression, where the loss function is allowed to be asymmetric:

Example 3. Quantile regression

Instead of the conditional mean, in quantile regression the q_{th} quantile conditional on X is modeled:

$$Q_q(y|X) = X\beta_q. \tag{12}$$

Let $e_i = y_i - x_i^T\beta_q$. The regression coefficients β_q are then found with solving

$$\min_{\beta_q} \sum_{i=1}^{N} \rho_q(e_i),$$

where ρ_q is an asymmetric loss function based on ρ_{L1}:

$$\rho_q(e) = \begin{cases} \rho_{L1}(e)(1-\tau) & for\, e \leq 0 \\ \rho_{L1}(e)\tau & for\, e > 0. \end{cases}$$

Solving this minimization problem requires in practice a reformulation to one that can be solved by linear programming (cf. Koenker (2000), Schmid (2011)).

Analogous to the perception of quantile regression as a generalization of median regression, Newey and Powell (1987) suggest expectile regression as a generalization of least squares regression, which can be more efficient than quantile regression if the distribution of the error term is in fact near normal (cf. Newey and Powell, 1987, p. 821). However, expectile regression is not robust. A robust yet efficient alternative once more based on the HP2 loss function is then M-quantile regression (cf. Breckling and Chambers, 1988) which is now presented in further detail:

Example 4. M-quantile regression

A linear M-quantile model has the form:

$$MQ_\tau(y|X) = X\beta_\tau. \tag{13}$$

Note that ρ is set to the HP2 loss function in this thesis and a fixed tuning parameter $k \in \mathbb{R}^+$ is assumed.

In M-quantile regression, the residuals need to be rescaled using a robust estimator of scale, which is in this thesis the mean absolute deviation:

$$s_{mad}(e) = \frac{med|e|}{0.6745}. \tag{14}$$

Hence, the ith rescaled residual is defined as

$$e_{i,mad} = \frac{y_i - x_i^T \beta_\tau}{s_{mad}}. \tag{15}$$

The now well known minimization problem (2) then leads to β_τ, i.e. using the influence function

$$\sum_{i=1}^{N} \psi_\tau(\frac{y_i - x_i^T \beta_\tau}{s_{mad}}) x_i \stackrel{!}{=} 0, \tag{16}$$

with

$$\psi_\tau(e) = 2 * \begin{cases} \psi_{HP2}(e)(1-\tau) & fore \leq 0 \\ \psi_{HP2}(e)\tau & fore > 0. \end{cases}$$

For a given M-quantile $\tau \in (0,1)$ the minimization problem can be solved using IWLS. The solution is then given by

$$\hat{\beta}_\tau = (X^T W_\tau X)^{-1} X^T W_\tau y, \tag{17}$$

where W is a diagonal matrix containing the weights that result from the IWLS procedure (cf. Chambers and Tzavidis, 2006, p. 260).

Remark 5. *A unique solution is guaranteed when a continuous monotone influence like the HP2 function is used (cf. Kokic et al., 1997).*

Since the weights depend on the estimation of β_τ, while β_τ depends at the same time on the weights, an iterative procedure like the IWLS is required. The key idea is to use the calculated residuals from the previous iteration until convergence:

Algorithm/Procedure 2. *IWLS procedure (cf. Fox and Weisberg (2010))*

1. *Calculate a starting solution e.g.* $\hat{\beta}^{(0)} = \hat{\beta}_{OLS}$

2. repeat s times until convergence:

(a) at iteration s calculate $e^{(s-1)} = y - \hat{y}^{(s-1)}$

(b) calculate estimator of scale $s_{mad}^{(s-1)}$ *and rescaled residuals* $e_{mad}^{(s-1)}$

(c) calculate $w_i^{(s-1)} = w_{HP2,\tau}(e_{i,mad}^{(s-1)})$ *for* $i = 1 \ldots, n$

(d) calculate a weighted regression for iteration s: $\hat{\beta_\tau}^{(s)} = (X^T W_\tau^{(s-1)} X)^{-1} X^T W_\tau^{(s-1)} y$, *where* $W_\tau^{(s-1)} = diag(w_1^{(s-1)}, \ldots, w_n^{(s-1)})$

Note that different convergence criteria are possible (e.g weights or residuals).

Alternatively, the estimates for β_τ can be obtained with maximum likelihood using the so called *Generalized Asymmetric Least Informative distribution*, as recently shown by Bianchi et al. (2015). This approach is not considered here, because the implementation in this thesis is based on the IWLS procedure. Results for asymptotic properties as well as an asymptotic variance estimator for M-quantile regression are derived by Bianchi and Salvati (2015), but not displayed here.

2.3 Random Effect Models

In many areas of research data can have a hierarchical nature, such that units of measurement can be attributed to some clusters or domains. For instance, in a longitudinal setting measurements at different time points can be clustered by the measured individual, or in small area estimation individuals can be clustered by some regional unit. Often this implies, that a part of the variance of the dependent variable can be attributed to varying intercepts and/or slopes in between these clusters, which can be modeled by random effects models:

Definition 4. *The random effects model*

Assume n individual units can be attributed to d clusters or domains. The random effects model is then given by

$$y_{ij} = x_{ij}^T \beta + z_{ij}^T \vartheta_j + \epsilon_{ij} \qquad for \quad i = 1, ..., n, \; j = 1, ..., d \; , \tag{18}$$

where x_{ij}^T *is ith row of the design matrix X (at unit level) and* z_{ij}^T *the ith row of the design matrix Z (at cluster level). Note that the dimensions of both matrices generally differ, as X has dimension* $n \times p$ *and Z has dimension* $n \times u$*. If Z is a vector of ones, the model reduces to a random intercept model, where there are only cluster specific intercepts:*

$$y_{ij} = x_{ij}^T \beta + \vartheta_j + \epsilon_{ij}, \tag{19}$$

otherwise it also incorporates random slopes, where the effects of some or all independent variables are varying in between clusters (cf. Chambers and Tzavidis, 2006).

As an example, assume a researcher wants to explain the grades of students clustered in classes. It might be, that average grades vary in between the classes, but the considered explanatory variables as such have the same effects in between the classes. This could be modeled by a random intercept model. If the effect of one or more independent variables on the grades also varies in between the classes, a random slope model could be more appropriate.

The random intercept model is in the focus of the small area models discussed in section 3. Note that this model is called basic nested error linear regression model in the work of Rao and Molina (2015).

To estimate mixed models, maximum likelihood estimation (MLE) or restricted maximum likelihood (REML) can be used. However, assumptions about the distribution of ϵ as well as ϑ are required. If the distribution is miss-specified or contaminated by outlying values, this can lead to inefficient and biased results (cf. Tzavidis et al., 2016, p.34). To avoid this, a possibility is to use a robust likelihood function that works in a similar manner like the Huber loss function, as it has been proposed by Welsh and Richardson (1997). However, as Schoch (2012) points out, there are no reliable algorithms to estimate these models, as convergence issues occur frequently. An alternative approach is to retrieve pseudo random effects based on M-quantiles which is now presented.

2.4 Pseudo Random Effect Models

Chambers and Tzavidis (2006) propose to use M-quantiles to compute quantities that they refer to with "pseudo-random effects". The general idea is to model random effects by an "average" M-quantile regression hyperplane in each domin (or cluster).

The approach of Chambers and Tzavidis (2006) exploits the property that M-quantiles characterize the distribution of a random variable, such that each value can be identified by a certain M-quantile. In a regression environment, where the conditional M-quantiles of y given X are modeled, that means that each value y_i will lie on a certain τth regression line or plane. Hence, $\exists\tau\forall i$, such that $y_i = MQ_\tau(y|X)$.

To find the pseudo random effect, Chambers and Tzavidis (2006) propose to define the corresponding $MQ_\tau(y|X)$ of every unit in y based on the whole population without any reference to the hierarchical structure. Once it is known which value of y is identified by which $MQ_\tau(y|X)$, for each domain j the average value of the τ values is calculated leading to

$\bar{\tau}_j$. $\beta_{\bar{\tau}_j}$ then represent the "pseudo random effects". Note that these can be pseudo random intercepts as well as pseudo random slopes.

Formally, the pseudo random effects model is therefore given by

$$y_{ij} = x_{ij}^T \beta_{\bar{\tau}_j} + \epsilon_{ij} \qquad \text{for} \quad i = 1, ..., n, \; j = 1, ..., d \; , \tag{20}$$

Note that each observation lies on one and only one regression plane (cf. Schmid, 2011), but of course several observations can lie on the same plane. A possible issue that can occur in practice is the crossing of planes or lines, but it is not covered in this thesis. As Chambers and Tzavidis (2006) state, crossing lines are often a result of a model-misspecification. A correction algorithm for crossing regression lines is described in Pratesi et al. (2009).

In practice, the corresponding τth M-quantile of a unit y_{ij} is found using an interpolation procedure. For this, a fine grid of M-quantile planes is estimated. The residuals between lines (or planes) and observations then serve as the measure of distance to each plane. If an observation lies exactly on a line, it is identified by that M-quantile regression line. If it lies in between two lines, a linear interpolation is used to determine the corresponding τ (cf. Schmid, 2011, p. 64). The τ_i values in each domain are then averaged. Using these values the pseudo random effects are calculated in a second set of M-quantile regressions. In short the procedure is given by:

Algorithm/Procedure 3. *Pseudo random effects with M-quantiles*

1. *run M-quantile regressions for a fine grid of τ values*
2. *for each y_i find the corresponding τ_i, using an interpolation procedure*
3. *average the τ_i for each domain j using the mean $\bar{\tau}_j = \frac{1}{N_j} \sum_{i \in j} \tau_i$*
4. *run d M-quantile regressions for $\bar{\tau}_j$ values resulting from (3) to obtain the pseudo random effects denoted by $\beta_{\bar{\tau}_j}$.*

Remark 6. *The pseudo random effects can be interpreted such that $\bar{\tau}_j > 0.5$ is a positive pseudo random effect, $\bar{\tau}_j < 0.5$ is a negative pseudo-random effect.*

Note that the interval δ between τ values in step (1) is supposed to be small to achieve a sufficiently fine grid. See also remark 7.

Example 5. Estimation of pseudo random effects

To illustrate procedure 3 assume the following simplified example with a population of $N = 500$ divided in 10 equally sized domains ($N_j = 50$) and the following model with random intercept and slope:

$$
\begin{aligned}
y_{ij} &= x_{ij} + x_{ij}\vartheta_j + \vartheta_j + \epsilon_{ij} \\
x_{ij} &\sim N(0, 1) \\
\epsilon_{ij} &\sim N(0, 1) \\
\vartheta_j &\text{ fixed to } 1, 2, \ldots, 10
\end{aligned}
$$

Figure 5 then shows step (1): The regression lines that result from estimating M-quantile regressions on fine grid. Here, a regular grid with interval $\delta = 0.01$ is used. The lines are plotted over a scatter plot of x and y.

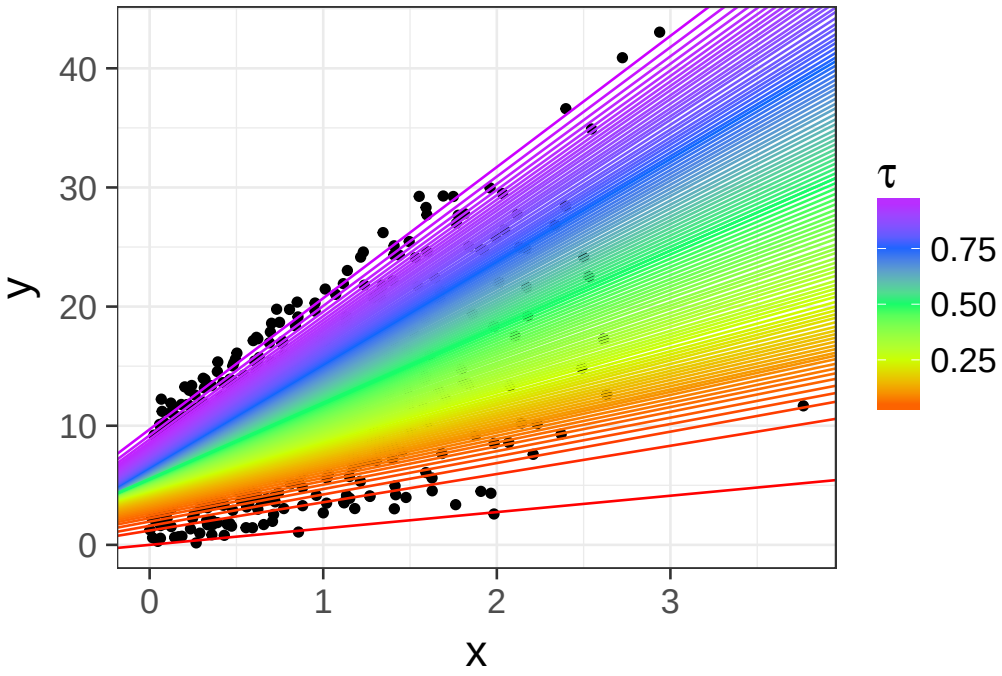

Figure 5: Grid of fitted M-quantile regression lines

Step (4) is then visualized in figure 6, where the regression lines plot the estimated pseudo random effects for each domain. The corresponding scatter points are likewise colored per domain. Visual inspection of this simple scenario confirms that the true intercepts and slopes are well approximated. Note that these plots are created using the inbuilt plot functions implemented in the newly developed R-package `mquantreg`. More sophisticated scenarios are simulated by Chambers and Tzavidis (2006) and provide support that pseudo random effects

can be used to estimate area means in random slope and random intercept models.

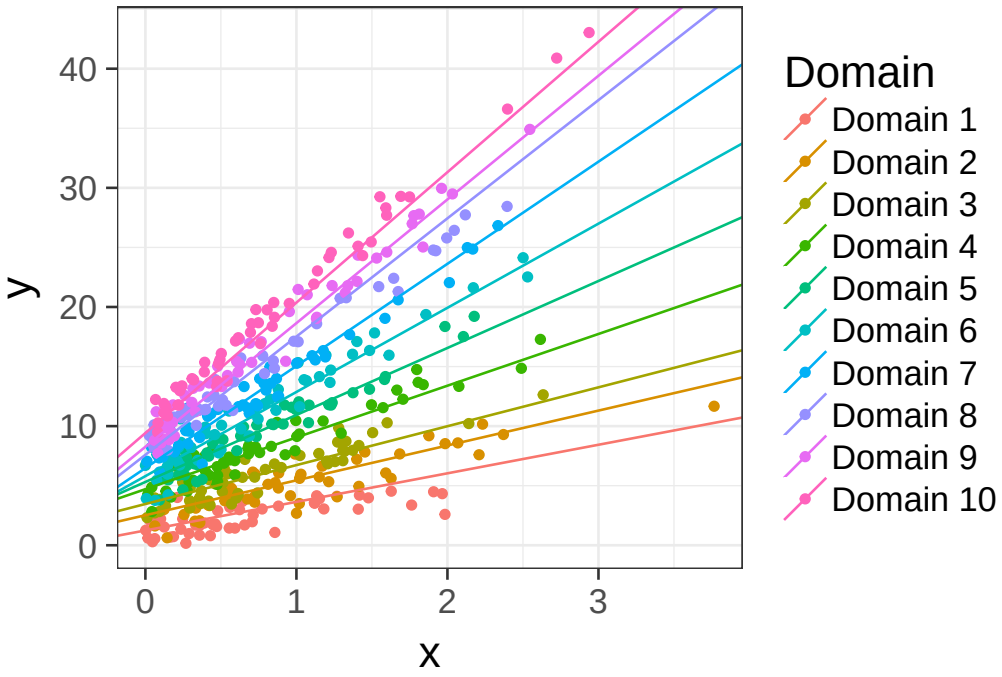

Figure 6: Plot of the pseudo random effects per domain

Remark 7. *The grid interval*

To my knowledge, there is no discussion in the literature, how the grid should be specified. However, as the following example shows, there can be a considerable relation between grid interval and bias as well as impreciseness of the pseudo random effects estimation.

To investigate how the interval δ between the τ values impacts the quality of the estimated random effects, a small simulation with 50 different grids $0.001, 0.001 + \delta, \ldots, 0.999$ is run, where $\delta = 0.001, 0.005, ...0.1$ For each grid, 50 populations with the following model are generated:

$$
\begin{aligned}
y_{ij} &= x_{ij} + \vartheta_j + \epsilon_{ij} \\
x_{ij} &\sim N(0, 10^2) \\
\epsilon_{ij} &\sim N(0, 1) \\
\vartheta_j &\text{ fixed to } 1, \ldots, 10.
\end{aligned}
$$

There are 30 domains with domain size fixed to $n_j = 50$. For each δ, relative bias and mean squared error are calculated over the 50 populations based on the later presented formulas (42) and (43).

Figure 7 indicates an almost linear relationship between the grid interval and relative bias/MSE. Hence, a grid of a lower resolution yields higher biases and variation. Therefore, in applications a rather fine grid should be chosen. Of course, this increases the computational time, and how fine the grid needs to be depends on the data and the application. In the replication studies in section 5.1 a $\delta \approx 0.05$ *already provides sufficient results. More research regarding this question can be considered as desirable.*

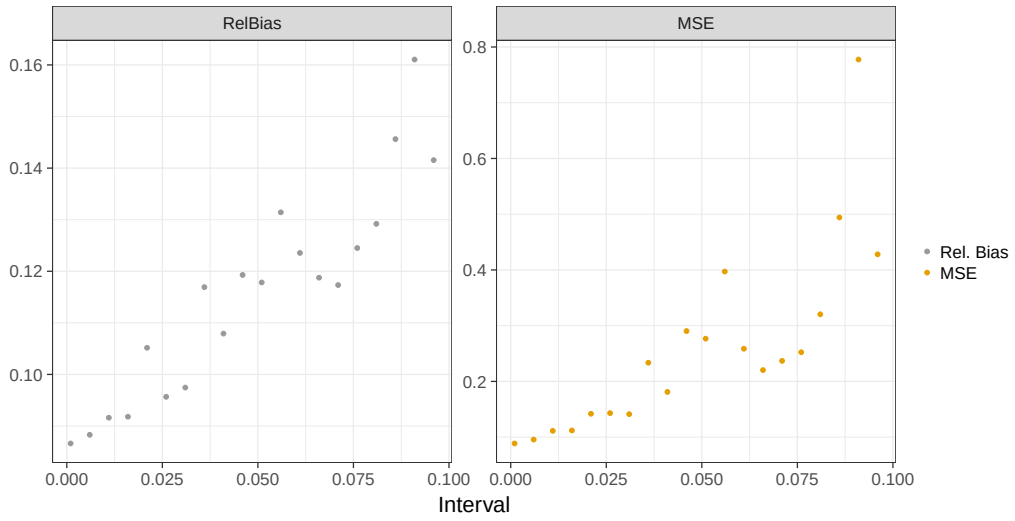

Figure 7: Absolute value of relative bias and MSE depending on grid interval

While quantile regression is rather widely used in applied research, expectile regression and M-quantile regression remain in a niche.[3] One reason for this is likely the better interpretability of quantiles and quantile regression results with regard to the β-coefficients, as there is no intuitive interpretation for M-quantiles and expectiles apart from the central point. Hence, it might be harder to convey what it means that a change in the independent variable leads to a change of the conditional τth M-quantile (or expectile) of the dependent variable. Another potential issue is that user friendly software for M-quantiles is not available until now. However, when the aim is not to estimate coefficients, but a robust prediction of unsampled values, M-quantile regression can be a relatively efficient yet robust method. An important area of research where this is of relevance is small area estimation, which is the subject of the next sections.

[3]For instance, searching on "Google Scholar" for "quantile regression", "expectile regression" and "M-quantile regression" returns 174.000, 1.510 and 689 results respectively as of the 26th September 2017.

3 M-quantile Models in Small Area Estimation

> "Reliable statistics are a key infrastructure element of a democratic society in any country of the world and thus also in developing countries. Governments need timely and reliable information to implement existing policies and programmes, to develop new policies across all sectors and to monitor their impact."
>
> (Eurostat, 2013, p.3)

In this section, it is shown how the previously presented concepts can be used in small area estimation (SAE). Firstly, the concept of small area estimation and its notation is presented. Then, with point estimation of means the following section is motivated where the estimation of non-linear indicators is discussed. For both linear and non-linear indicators emphasis is put on models that are based on (pseudo) random effects. Finally, an approach to non-parametric mean squared error (MSE) estimation is presented.

3.1 Foundations of Small Area Estimation

The importance of small area estimation has already been discussed in the introduction, but, a definition is still pending:

Definition 5. *Small area*

For a given sample of data, a domain or area is regarded as a small area, when direct estimators of sufficient precision cannot be obtained, because the number of sampled observations in the area is too low (cf. Rao and Molina, 2015).

Examples for small areas can be smaller geographical units in a country, but also firms in a branch or classes in schools. The research interest in these areas or domains is to retrieve reliable estimates of a target indicator on domain level, e.g. a linear indicator like the mean or a non-linear indicator like on of the poverty indicators defined in the next section. However, often the available number of observations in certain domains at lower regional levels are too small, because cost constraints do not allow to survey a reasonable large number of individuals in each domain, a survey was initially designed to give accurate estimates for the overall population and interest in the small areas grew at a later point or official statistics are still in its infancy, as it might be the case in developing countries (cf. Rao and Molina, 2015).

In the absence of a sufficiently large number of observations, using only the sampled data will lead to too imprecise results and there might be even no observations at all in a domain.

Small area estimation subsumes a variety of methods of which some "borrow strength" from population data or auxiliary information to increase the precision of the estimation.

In what follows, it is assumed that a finite population Ω of size N can be partitioned into D known domains $\Omega_1, \Omega_2, \ldots, \Omega_D$ of sizes N_j, where $j = 1, \ldots, D$ refers to an jth domain and $i = 1, \ldots, n_j$ to the ith unit in area j. Let s_j denote the set of sampled and r_j the set of remaining, unsampled units in domain j. The sample sizes for each area are denoted by $n_1, \ldots, n_d$ leading to an overall sample of size n. With y the target variable is referred to, while X denotes a matrix of available auxiliary variables that are related to y. It is generally assumed, that information on y is only available for the sampled data, but X is measured for sampled and unsampled units .

In Molina and Rao's framework for SAE several types of estimators can be distinguished. For the economy of space, only the distinctions important for this thesis are briefly explained. For a more detailed taxonomy see Rao and Molina (2015).

Direct estimators are based only on the available data in a domain, that is the units $i \in s_j$. These estimators can be design based (cf. Rao and Molina, 2015, p.1), such that they take into account non-random inclusion probabilities for the sampled units that result from the survey design. For instance, when certain surveyed groups are over- or undersampled. An example is the Horvitz-Thompson estimator (cf. Horvitz and Thompson, 1952), which is basically a weighted mean:

Example 6. Horvitz-Thompson estimator

Let w_i denote the inverse of the inclusion probability of a unit i, then the mean of area j is given by

$$\hat{\mu}_j^{HT} = \frac{\sum_{i \in s_j} w_i y_i}{\sum_{i \in s_j} w_i}. \tag{21}$$

Indirect estimators include information from large areas (e.g. the whole population) to increase the effective sample size in the small area. In the here presented concepts this is done with exploiting a relationship between the target variable y and some auxiliary information X. Auxillary information can in practice be taken from other surveys (i.e. a census) or even unconventional data sources like mobile phone data (see e.g. Schmid et al. (2017)). This data has to be available for the set r and s.

If an indirect estimator explicitly accounts for variation in between the areas, i.e. based on random effects, Rao and Molina (2015) refer to it as *small area model*. Depending on which level the auxiliary information is available, one can also distinguish area and unit level

models, albeit area level models like the Fay-Herriot model are not considered in this thesis. This is because the in section 3.3.2 ultimately presented *M-quantile approach for small area estimation* is essentially a unit level estimator.

For the unit model, recall that in section 2.3 the random effects model is introduced. This model is the core of the basic unit level model. In this model, one can "borrow strength" based on auxiliary information X, where the fixed part describes the relation between y and X without reference to the domains, and domain specific effects are modeled using random effects.

Definition 6. *The basic unit level model (BULM)*

Consider the random intercept model (19)*. Assume auxiliary information X is given for the sampled as well as non-sampled units. The basic unit level model is given by*

$$y_{ij} = x_{ij}^T\beta + \vartheta_j + \epsilon_{ij} \qquad for \;\; i = 1, ..., n, \; j = 1, ..., d \; , \tag{22}$$

where ϵ_{ij} is a random error term with $E[\epsilon_{ij}] = 0, V[\epsilon_{ij}] = \sigma^2_{\epsilon_{ij}}$, ϑ_{ij} is a random effect with $E[\vartheta_{ij}] = 0, V[\vartheta] = \sigma^2_\vartheta > 0$. The correlation between ϵ_{ij} and ϑ_j is denoted by γ_j. Often, normality is assumed for both error terms (cf. Rao and Molina, 2015).

The BULM was firstly used in the SAE context by Battese et al. (1988) in an application in agriculture.

3.2 Small Area Estimation of Means

Under the assumption of known variances in the basic unit level model one can derive estimators for linear indicators for the mean that have the property to be the best linear unbiased predictor (BLUP). This is the pendant for random effect models to being the best linear unbiased estimator (BLUE) in linear regression models. In practice, the variances σ_e and σ_ϑ, as well as their possible correlation, are however unknown parameters that need to be estimated. Replacing these unknown population quantities by some estimated analogues, leads to estimators that are no longer BLUP, but only *empirical* BLUP or EBLUP (cf. Gonzalez-Manteiga et al., 2008, p. 444).

For example, under a consistent estimator for variances and β in the basis unit model the EBLUP estimator for the mean of area j is given by (cf. Schmid, 2011, p.31):

$$\hat{\mu}_j^{EBLUP} = \frac{1}{N_j}\left[\sum_{i\in s_j} y_{ij} + \sum_{i\in r_j}(x_{ij}^T\hat{\beta} + \hat{\vartheta}_j)\right] . \tag{23}$$

The estimation of a linear indicator like the small area mean is therefore essentially a question of how the non-sampled values are estimated. Estimates for variances and β can be obtained under distributional assumptions by MLE, REML or the method of moments (cf. Schmid et al., 2017, p. 32). Note that the additional uncertainty from the variance estimation needs to be accounted for in the MSE estimation, which often makes analytic derivation of the MSE impossible (cf. Gonzalez-Manteiga et al., 2008, p. 444).

An alternative to EBLUP estimation without distributional assumptions is based on M-quantiles. In section 2.4 it was shown that pseudo random effects can be found based on M-quantile regression. Because predictions for $y \in r_j$ can be retrieved using these quantities, one might expect there should be an analogous estimator to the EBLUP estimator (23) for the mean. Indeed, a simple M-quantile based plug-in estimator is proposed by Chambers and Tzavidis (2006) and given by

$$\hat{\mu}_j^{PIMQ} = \frac{1}{N_j}\left(\sum_{i \in s_j} y_i + \sum_{i \in r_j} x_i^T \hat{\beta}_{\bar{\tau}_j}\right), \tag{24}$$

with $\beta_{\bar{\tau}_j}$ being the pseudo random effects of area j (see section 2.4). Chambers and Tzavidis (2006) derive this estimator based on a first-order Taylor approximation, but in a later paper Tzavidis et al. (2010) show that $\hat{\mu}_j^{PIMQ}$ is the expected value of the naive estimator of the distribution function in an area. Because this naive estimator for the distribution function is not generally consistent, the plug-in estimator for the mean can be biased (cf. Tzavidis et al., 2010, p. 167). The authors also propose a bias corrected estimator for the mean, which is based on the Chambers and Dunstan estimator for the distribution function. We will return to both distribution estimators in 3.3.2 in the context of non-linear indicators, but firstly the importance of non-linear estimation is motivated in the now following section.

3.3 Small Area Estimation of Distributions and non-linear Indicators

Research interest is not limited to linear indicators. One might be interested in finding other parts of the distribution in small areas, like quantiles. Moreover, an important application of small area estimation is the estimation of poverty and inequality which is generally based on non-linear indicators. Common indicators for poverty measured in terms of income or expenditure are the Foster-Greer-Thorbecke (FGT) poverty measures (Foster et al., 1984):

Definition 7. *The FGT poverty measures*

Let y be a welfare variable and z be a poverty threshold. The FGT measures for area j

are given by

$$F_{\alpha j}^{FGT} = \frac{1}{N_j} \sum_{i=1}^{N_j} \left[\left(\frac{z - y_{ij}}{z} \right)^{\alpha} I(y_{ij<z}) \right], \alpha \in \{0, 1, 2\}. \tag{25}$$

For $\alpha = 0$, the head count ratio (HCR) is obtained and gives the ratio of individuals in poverty to the population. Alternatively, $\alpha = 1$, defines the poverty gap (PG) which gives the average distance individuals have to the poverty threshold, as proportion of the threshold. Both indicators range from 0 to 1 (cf. Rao and Molina, 2015, p. 293). Poverty severity ($\alpha = 2$) is not considered in this thesis.

Note that the threshold z can be a relative value, e.g. 60% of the median of the target variable or an absolute value.

For the measurement of inequality, the Gini coefficient is a well known indicator:

Definition 8. *The Gini coefficient*

Suppose $y_1, ..., y_N$ are the values of y ordered in ascending order.

$$Gini = \frac{2 \sum_{i=1}^{N} i y_i}{N \sum_{i=1}^{N} y_i} - \frac{N+1}{N}. \tag{26}$$

The Gini coefficient ranges from 0 to 1, with 1 being maximum inequality (one unit owns everything, the remaining units nothing)

However, it will also become important in this thesis that the Gini can be defined based on the distribution function of y. Let μ be the mean of F, then it holds

$$Gini = \frac{1}{\mu} \int_0^{\infty} F(y)(1 - F(y))dy \tag{27}$$

(cf. Gastwirth, 1972).

For non-linear indicators, it is not generally feasible to analytically derive point-estimates and measures for their precision. Therefore, the EBP method and an approach based on M-quantiles is now presented. Both methods make use of Monte-Carlo approximations, but differ in their assumptions and how predictions for the non-sampled units $y_{ij} \in r_j$ are obtained. As stated in the introduction, a further alternative is the ELL method of Elbers et al. (2003), but it is not considered in this paper.

3.3.1 Empirical Best Prediction (EBP)

Molina and Rao (2010) propose an estimation procedure called empirical best prediction (EBP) which can be used for non-linear indicators in a small area context, i.e. poverty mea-

sures. For the economy of space, it is only briefly explained why in this method Monte-Carlo approximation is required and the most important steps of the algorithm are summarized. This allows to see similarities and differences to the M-quantile approach presented in the following section. The theory is described in more detail by Molina and Rao (2010).

Assume the target indicator ν is a non-linear indicator, which is a function h of the variable y. ν can be for example an FGT poverty measure. Given the distinction between sampled and non sampled units, the non-linear indicator can be written as (cf. Molina and Rao, 2010, p.373):

$$\nu_j = \frac{1}{N_j} h_j(y_j) = \frac{1}{N_j} \left[\sum_{i \in s_j} h(y_{ij}) + \sum_{i \in r_j} h(y_{ij}) \right]. \tag{28}$$

An estimator for the in practice unknown second term $h(y_{ij}), i \in r_j$ with desirable properties is the best predictor (BP), as this is the function $\hat{h}$ that minimizes the mean squared error

$$MSE(\hat{h}(y)) = E_y \left[(\hat{h}(y) - h(y))^2 \right].$$

However, finding the best predictor for the unknown part of h requires to solve

$$\hat{h}^{BP}(y) = E_{y_{i \in r}}(h(y)|y_{i \in s}),$$

but calculating this expected value (an integral) is not explicitly possible due to the complexity of the non-linear transformation induced by h (cf. Molina and Rao, 2010, p. 373).

To overcome this issue, a Monte-Carlo approximation can be used that is described now. An implementation of this procedure in `R` in the package `emdi` is provided by Kreutzmann et al. (2017). Note that the EBP approach requires to transform the variable y to a normally distributed dependent variable. For this, some additional steps are implemented and described in more detail in Kreutzmann et al. (2017).

Algorithm/Procedure 4. *The EBP algorithm*

1. *using the sampled data, obtain estimators* $\hat{\beta}, \hat{\sigma}^2_\vartheta, \hat{\sigma}^2_e, \hat{\vartheta}_j = E[\vartheta_j|y_j], \hat{\gamma}_j = \frac{\hat{\sigma}^2_\vartheta}{\hat{\sigma}^2_\vartheta + \frac{\hat{\sigma}^2_e}{n_j}}$ *for the assumed BULM.*

2. *for* $l = 1, .., L$

   (a) *generate pseudo-populations:* $y^{*(l)}_{ij} = x^T_{ij}\hat{\beta} + \hat{\vartheta}_j + \vartheta^*_j + e^*_{ij}$ *where*

   - *for sampled domains:* $e^*_{ij} \sim N(0, \hat{\sigma}^2_e)$ *and* $\vartheta^*_j \sim N(0, \hat{\sigma}^2_\vartheta(1 - \hat{\gamma}_j))$

- *for non-sampled domains:* $e_{ij}^* \sim N(0, \hat{\sigma}_e^2)$ *and* $\vartheta_j^* \sim N(0, \hat{\sigma}_\vartheta^2), \hat{\vartheta}_j = 0$

(b) calculate linear or non-linear target indicator $\hat{\nu}_j^{(l)}$

3. *calculate* $\hat{\nu}_j^{EBP} = \frac{1}{L} \sum_{l=1}^{L} \hat{\nu}_j^{(l)}$ *for each domain.*

To estimate the MSE, a paramteric bootstrap procedure can be used (see Molina and Rao (2010), Kreutzmann et al. (2017)).

Note that the EBP is based on the assumption of normally distributed error terms for ϑ and ϵ. An alternative without distributional assumptions is given by the now presented M-quantile approach for non-linear indicators.

3.3.2 M-quantile Small Area Approach (MQ (SAE))

A non-parametric procedure based on M-quantiles and Monte-Carlo approximation is then proposed by Marchetti et al. (2012). The theoretical foundation of this method is different to the EBP, as the M-quantile approach does not minimize a criterion comparable to the MSE (cf. Rao and Molina, 2015). The work of Marchetti et al. (2012) is rather motivated by Tzavidis et al. (2010), who introduce the M-quantile based estimation of quantiles based on the distribution function in an area, which is therefore discussed first.

Consider the distribution function in an area with distinction between sampled and non-sampled units:

$$F_j(t) = \frac{1}{N_j} \left[\sum_{i \in s_j} I(y_{ij} \leq t) + \sum_{i \in r_j} I(y_{ij} \leq t) \right]. \tag{29}$$

A naive estimator of F_j is then given by plugging in some predicted values for $y_{ij} \in r_j$, which could be obtained by M-quantile pseudo random effects based on the sample data.

Remark 8. *Integrating over the naive estimator F yields the plugin-in estimator of the mean described in section 3.2.*

However Tzavidis et al. (2010) argue, that the naive estimator of F can be inconsistent, and particularly it should not be used to estimate location parameters like quantiles that are distant from the median. The authors suggest to use an alternative, model-consistent estimator proposed by Chambers and Dunstan (1986), which includes a term that the authors refer to as *"smearing argument"*. It is given by

$$\hat{F}_j^{CD}(t) = \frac{1}{N_j} \left[\sum_{i \in s_j} I(y_{ij} \leq t) + \frac{1}{n_j} \sum_{k \in r_j} \sum_{i \in s_j} I(\hat{y}_{kj} + e_{ij} \leq t) \right], \tag{30}$$

where e_{ij} are the residuals from the estimating model (these can be conditional or unconditional to the area, see remark 9), $\hat{y}_{kj}$ are the fitted values based on that model and n_j is the number of sampled units.

The Chambers and Dunstan estimator can be used in conjunction with M-quantile based pseudo random effects. To obtain fitted values for the non-sampled units, the area specific pseudo random effects are estimated based on the sample data. The predictions for $\hat{y}_{kj}$, $k \in r_j$ are then calculated with $\hat{y}_{kj} = x_{kj}^T \hat{\beta}_{\bar{\tau}_j}$. Residuals are also retrieved from the model calculated with the sample data: $e_{ij} = y_{ij} - x_{ij}^T \hat{\beta}_{\bar{\tau}_j}$. The "smearing argument" then works as follows: Note that the second term in equation (30) can be written as

$$\sum_{k \in r_j} \frac{1}{n_j} \sum_{i \in s_j} I(\hat{y}_{kj} + e_{ij} \leq t).$$

Hence, instead of simply evaluating the indicator function for the predicted values $\hat{y}_{kj}$, as it would be done in the naive estimator, each of the residuals corresponding to the n_j elements in s_j is added to each predicted value. Then, the arithmetic mean over these n_j indicator functions is calculated.

With the estimated empirical distribution function in an area it is then possible to calculate indicators that can be derived from this function. For instance, the qth quantile can then be obtained based on integration, which in practice requires a numerical procedure:

$$\int_{-\infty}^{Q_{q,j}} d\hat{F}_j^{CD}(t) = q. \tag{31}$$

To obtain FGT measures the "smearing argument" can also be introduced in equation (25) as described in Marchetti et al. (2012):

Consider the FGT measures for area j with threshold z.

$$F_{\alpha j}^{FGT} = \frac{1}{N_j} \sum_{i=1}^{N_j} \underbrace{\left[\left(\frac{z - y_{ij}}{z} \right)^{\alpha} I(y_{ij} < z) \right]}_{h_{\alpha,z}(y_{ij})}, \alpha \in \{0, 1\}. \tag{32}$$

For introducing the distinction between sampled and non-sampled areas, rewrite

$$F_{\alpha j}^{FGT} = \frac{1}{N_j} \left[\sum_{i \in s_j} h_{\alpha,z}(y_{ij}) + \sum_{i \in r_j} h_{\alpha,z}(y_{ij}) \right]. \tag{33}$$

An estimator with "smearing argument" for the non-sampled units is then given by

$$\hat{F}_{\alpha j}^{FGT} = \frac{1}{N_j}\left[\sum_{i\in s_j} h_{\alpha,z}(y_{ij}) + \frac{1}{n_j}\sum_{k\in r_j}\sum_{i\in s_j} h_{\alpha,z}(y_{ij}+e_{ij})\right]. \tag{34}$$

It is now possible to approximate this estimator with a Monte-Carlo procedure. While the authors use this Monte-Carlo procedure only for the FGT Measures, it can be expected to also work for other measures that can be derived from the distribution function, such as quantiles and non-linear indicators like the Gini coefficient. The reason for this is that the Chambers Dunstan estimator for the distribution function evaluated at the threshold z coincides with the FGT poverty measure for $\alpha = 0$ (cf. Marchetti et al., 2012), because it holds:

$$h_{\alpha=0,z}(y_{ij}) = I(y_{ij} < z). \tag{35}$$

Note that the authors do not propose how non-sampled areas are dealt with. In the implementation of the `mquantreg`-package, I therefore follow the suggestion of Rao and Molina (2015) and use the $\tau = 0.5$ M-quantile for these areas. The algorithm for the Monte-Carlo estimation of non-linear indicators based on M-qunatiles is then given by:

Algorithm/Procedure 5. *The MQ (SAE) algorithm*

1. *using the sampled data, obtain M-quantile pseudo random effects (see section 2.4). Calculate vector of residuals* $e_r = (e_{11}, \ldots, e_{n_j j_d})^T$ *with* $e_{ij} = y_{ij} - x_{ij}^T\hat{\beta}_{\bar{\tau}_j}$
2. *for* $l = 1, .., L$
   (a) *generate pseudo population values* $y_{ij}^* \in r_j$
      - *for* $e_{11}^*, \ldots, e_{ij}^*, i \in r_j$*, take a random sample with replacement of length* $N_j - n_j$ *from* e_r
      - *generate* $y_{ij}^* \in r_j$ *of length* $N_j - n_j$ *with* $y_{ij}^{*(l)} = x_{ij}^T\hat{\beta}_{\bar{\tau}_j} + e_{ij}^*$*:*
        - *for sampled domains:* $\bar{\tau}_j$ *results from step (1)*
        - *for non-sampled domains:* $\bar{\tau}_j = 0.5$

   (b) *combine vector of sampled values* $y_{ij} \in s_j$ *and* $y_{ij}^* \in r_j$

   (c) *calculate linear or non-linear target indicator* $\hat{\nu}_j^{(l)}$
3. *calculate* $\hat{\nu}_j^{MQ} = \frac{1}{L}\sum_{l=1}^{L}\hat{\nu}_j^{(l)}$ *for each domain.*

as proposed by Marchetti et al. (2012).

Remark 9. *The authors also propose an alternative way to sample $e^*_{11}, \ldots, e^*_{ij}$ which is called the "conditional approach". Here the values are sampled from area specific vectors of errors, e_{rj}. This approach is not considered here, as it is not recommended by the authors for small samples in areas (the very reason, why small area estimation is done in the first place). In addition, it would not work for non-sampled areas.*

As Marchetti et al. (2012) state, one can also calculate expression (34) without the Monte-Carlo approach and that the point estimates as well as the computational time are almost identical. However, if the number of sampled units and unsampled units is rather large in one area, it is likely that it is computationally advantageous not to cross every sampled with every unsampled unit as required by the "smearing argument". Assume there are 1000 sampled units, and 5000 non-sampled units in an area, this would require to calculate in equation (34) five million indicator functions and that is only in one area. Using the Monte-Carlo approach, this reduces to $5000L$ calculations, where L can be a value around 50. It is imaginable, that in such situations the Monte-Carlo approach is faster, especially when it is bootstrapped to calculate the MSE (see section 5.3).

While the MQ (SAE) approach is in some way similar to the EBP, there is a notable difference: for the EBP, because of the parametric assumptions about the distribution of the error terms, the pseudo populations can be generated with sampling from a normal distribution based on the estimates for the individual and area specific error terms. In the MQ-based approach, one can only draw from the vector of residuals of step (1).

An empirical comparison of both estimators under different simulated scenarios is presented in section 5.2.

3.4 Non-parametric Bootstrap MSE Estimation for non-linear Indicators

> "[...] every serious estimate deserves a reliable assessment of precision."
>
> (Koenker and Hallock, 2001, p.153)

It was already stated in section 3.3.1 that MSE estimation for non-linear estimators is difficult. Even for EBLUP estimation of the mean there is in general no analytic solution for the MSE (though, under certain assumptions, large sample approximations can be made (cf. Gonzalez-Manteiga et al., 2008, p. 444)). Instead, MSE estimation can be done by some form of bootstrapping. For example, for the EBP a parametric bootstrap as proposed by Molina and Rao (2010) or a semi- parametric wild bootstrap (cf. Kreutzmann et al., 2017)

can be used to obtain MSE estimates. Note that the parametric bootstrap procedure relies once more on Gaussian assumptions about the error distributions.

For the M-quantile approach the situation is not too different. Chambers and Tzavidis (2006) derive an analytic solution to estimate the MSE for the naive mean (see section 3.2), however this MSE estimator can be unstable when there are only few observation in an area (cf. Marchetti et al., 2012) and deriving an analytic solution for the non-linear poverty estimators or quantiles is not feasible. Alternatively, Tzavidis et al. (2010) and Marchetti et al. (2012) present a non-parametric bootstrap approach that can be used for linear, as well as non-linear indicators and is based on the work of Lombardia et al. (2004).

The approach described in Marchetti et al. (2012) is based on the generation of a number of super- or bootstrap populations, from which samples are drawn. In each sample, the target indicator is estimated. Based on these estimations and the true bootstrap population values bias and variance can be estimated, as shown shortly.

The bootstrap procedure is now presented for the FGT measures. Because the FGT measures are closely tied to the empirical distribution function (as shown in the previous section), the now presented bootstrapping procedure can also be used for any measures derived from the distribution function like quantiles and the Gini.

The bootstrap procedure starts with an initial calculation of the pseudo random effects model. Recall that this model is given for area j by

$$y_{ij} = x_{ij}^T \beta_{\bar{\tau}_j} + \epsilon_{ij}$$

.

Note that there are no distributional *assumptions* about ϵ_{ij}, however, of course the error term *does* have a distribution, which is henceforth denoted by G.

To generate the bootstrap or superpopulations, it is necessary to sample from that distribution of G. An estimator $\hat{G}$ is given by the area-unconditional empirical distribution function of the *centered* residuals that result from the estimation of model (20).

$$\hat{G}(t) = \frac{1}{n}[\sum_{i \in j} \sum_{i \in s_j} I(e_{ij} - \bar{e} \leq t)]. \tag{36}$$

Now denote the superpopulation by

$$y_{ij}^* = x_{ij}^T \hat{\beta}_{\bar{\tau}_j} + e_{ij}^*, \tag{37}$$

where e_{ij}^* is drawn from the empirical distribution function $\hat{G}$ of the residuals.

Remark 10. *The authors alternatively propose to sample from a smoothed distribution function of the residuals, where the smoothing is done e.g. with an Epanechnikov kernel. Additionally, but not recommended by the authors, the distribution can be estimated conditional to the areas. See remark 9.*

Based on the superpopulations, stratified samples are drawn. An estimator for the FGT measures is then given by the superpopulation analogue to the already known expression (34):

$$\hat{F}_{\alpha j}^{FGT,*} = \frac{1}{N_j}\left[\sum_{i\in s_j} h_{\alpha,z}(y_{ij}^*) + \frac{1}{n_j}\sum_{k\in r_j}\sum_{i\in s_j} h_{\alpha,z}(y_{ij}^* + e_{ij}^*)\right]. \tag{38}$$

An approximate of (38) can be calculated based on the Monte-Carlo procedure described in algorithm 5. Hence, estimates for variance and bias can be obtained using the following algorithm:

Algorithm/Procedure 6. *MSE estimation with non-parametric bootstrapping*

1. *using the sampled data, obtain the M-quantile pseudo random effects (see section 2.4). Retrieve vector of residuals* $e = (e_{11}, \ldots, e_{n_j j_d})^T$ *with* $e_{ij} = y_{ij} - x_{ij}^T\hat{\beta}_{\bar{\tau}_j}$
2. *center vector of residuals to obtain* $\hat{G}$
3. *for* $b = 1, \ldots, B$
   (a) *generate superpopulation* Ω^**, maintaining small area sizes* N_j*:*
      i. *to obtain* $e_{11}^*, \ldots, e_{ij}^*, i \in r_j \bigcup s_j$*, take a random sample with replacement of size* N_j *from* $\hat{G}$
      ii. *generate* $y_{ij}^{*(b)} = x_{ij}^T\hat{\beta}_{\bar{\tau}_j} + e_{ij}^*$*:*
         - *for sampled domains:* $\bar{\tau}_j$ *results from step (1)*
         - *for non-sampled domains:* $\bar{\tau}_j = 0.5$
      iii. *for* $s = 1 \ldots S$
         A. *take a stratified random sample without replacement from superpopulation such that* $n_j^* = n_j$
         B. *calculate target indicator* $\hat{\nu}_j^{*(bs)}$ *for bootstrap b, sample s using Monte-Carlo approach with L iterations (see algorithm 5)*

(b) calculate mean of the S estimated target indicators in bootstrap b: $\bar{\hat{\nu}}_j^{*(bs)}$

(c) calculate true indicator for bootstrap b: $\nu_j^{*(b)}$

4. calculate bias, variance and MSE.

To estimate variance and bias, consider the following formulas:

$$\hat{Bias}(\hat{\nu}_j) = \frac{1}{B}\frac{1}{S}\sum_{b=1}^{B}\sum_{s=1}^{S}\left(\hat{\nu}_j^{*(bs)} - \nu_j^{*(b)}\right), \tag{39}$$

$$\hat{Var}(\hat{\nu}_j) = \frac{1}{B}\frac{1}{S}\sum_{b=1}^{B}\sum_{s=1}^{S}\left(\hat{\nu}_j^{*(bs)} - \bar{\hat{\nu}}_j^{*(bs)}\right)^2. \tag{40}$$

The estimator for the MSE is then given by

$$\hat{MSE}(\hat{\nu}_j) = \hat{Var}(\hat{\nu}_j) + \hat{Bias}(\hat{\nu}_j)^2. \tag{41}$$

To achieve stable results, the authors recommend to choose B and S between 50 and 100. As the authors highlight, the asymptotic properties for the MSE-estimator, i.e. for its consistency are not yet theoretically derived in general (cf. Marchetti et al., 2012). Some empirical results for bias and uncertainty are given in section 5, but firstly the implementation of the so far described M-quantile based methods in `R` is presented.

4 Implementation in R

> "The diffusion of technological change throughout statistics is closely tied to its embodiment in statistical software."
>
> (Koenker and Hallock, 2001, p.153)

In this part, the implementation of M-quantile regression, mixed M-quantile models (to obtain pseudo random effects) and the M-quantile small area approach in R is described. R is a widely used open source programming language, mostly in the context of statistical computing (cf. R Core Team, 2017). To begin with, some related R-packages and available functions are described, followed by a description of the general approach and of the functions and arguments that are now implemented in the `mquantreg` package. Furthermore, some suggestions for code improvement are presented.

4.1 Available R-Packages and unpublished R-Functions

For expectile and quantile regression the packages `expectreg` (cf. Sobotka et al., 2014) and `quantreg` (cf. Koenker, 2017) are available and some of their functions have been used as a orientation for the now developed package. The function `rlm` in the `MASS` (cf. Venables and Ripley, 2002) package allows to fit regression with the HP2 loss function for the central point, but not for quantile like measures. This function can however be altered to allow for asymmetric loss functions (cf. Chambers and Tzavidis, 2006).

For small area estimation, some robust methods for linear indicators based on unit and area levels are implemented in the `rsae` package (cf. Schoch, 2014). However, this package does not work for quantiles or non-linear indicators. Non-linear indicators based on the EBP method can be estimated by the `emdi` package (cf. Kreutzmann et al., 2017).

Additionally, there are some solitary functions available from the project "Small Area Methods for Poverty and Living Condition Estimates" (S.A.M.P.L.E, http://sample-project.eu). These functions are based on the work of Chambers and Tzavidis (2006), Tzavidis et al. (2010), Marchetti et al. (2012) and have been cordially provided by Stefano Marchetti, PhD.:

`mq.sae(y, x, regioncode.s, m, p, x.outs, regioncode.r, tol.value, maxit.value, k.value)`: This function calculates SAE averages and an analytic MSE

Note: Contrary to the ***`mquantreg`*** *package, the averages are estimated with the naive or a bias corrected estimator, and an analytic MSE is calculated.*

`mq.sae.quant(q, y, x, x.outs, regioncode.s, regioncode.r, MSE, B, R, method, maxit)`: Estimation of small area quantiles with mq.sae.quant and a bootstrap MSE

Note: Contrary to the ***mquantreg*** *package, the quantiles are estimated based on a numerical integration procedure.*

`mq.sae.poverty(y, x, x.outs, regioncode.s, regioncode.r, L, MSE, B, R, method)`: This function estimates the HCR and PG, based on a Monte-Carlo approximation.

Note: For these indicators, the same approach implemented in the ***mquantreg*** *package is used. See section 5.1 for the replication results.*

4.2 The new R-Package `mquantreg`

A major contribution of this thesis is the development of the now presented R-package `mquantreg`. It is intended to be at least partially integrated in the `emdi` package, but a part could also be used as the basis for a new package that focuses on M-quantile regression outside the small area context. Therefore, the functions for M-quantile regression, mixed M-quantile models and M-quantile small area models are programmed in such a way that they can be run independently. This leads to a considerable increase in programming effort, because argument checks and S3 methods[4] that allow to use standard R functions like plot, print, predict and summary needed to be implemented where applicable.

The contributed R-package now consists mainly of three sources:

- Own programming
- S.A.M.P.L.E. R-Code provided by Prof. Timo Schmid and Stefano Marchetti, PhD. Some of this code is based on a modification of the `rlm` package
- R-Code from the `emdi` package, that could be easily modified to be used for the `mquantreg` package

The following table shows the new functions, and their main origin. It should be highlighted that the code from the S.A.M.P.L.E project was also very helpful to clarify the required steps for those functions, that were finally reprogrammed. Also, the `emdi` package served as very important framework for the `mq_sae` functions. Many parts like the argument checking, the data framework or parallelization, could be adapted with little modifications. Additionally, it was a helpful draft to maintain a consistent structure of the code.

[4] S3 is a special case of object oriented programming implemented in R, where generic functions decide based on the class of an object, which function is called (cf. Wickham, 2014). For example, for an object of class `lm`, the generic function `print` will call the function `print.lm`.

	Own programming	emdi	S.A.M.P.L.E	MASS
bootstrap_monte_carlo_mq	✓	✓		
bootstrap_mq_sae	✓	✓		
framework_mq_sae	✓	✓		
fw_check1	✓	✓		
fw_check2	✓	✓		
fw_check3	✓	✓		
G	✓			
G_hat	✓			
icc		✓		
irls.delta				✓
irls.rrxwr				✓
IWLS	✓		✓	✓
mmqm	✓			
mmqm.check	✓			
monte_carlo_mq	✓	✓		
mq	✓		✓	✓
mq.check	✓			
mq_sae	✓	✓		
mq_sae_check1	✓	✓		
mq_sae_check2	✓	✓		
mquantile	✓			
mse_estim_mq	✓	✓		
mse_estim_wrapper_mq	✓	✓		
plot.mmqm	✓			
plot.mq	✓			
point_estim_mq	✓	✓		
predict.mmqm	✓			
predict.mq	✓			
prediction_y_mq	✓			
print.emdi	✓	✓		
print.mmqm	✓			
print.mq	✓			

print.summary.emdi	✓			
psi.huber				✓
summary.emdi	✓	✓		
summary.mmqm	✓			
summary.mq	✓			
superpopulation_mq	✓	✓		
transform_x	✓			
zerovalinter			✓	

Table 3: Main origin of functions and subfunctions in the `mquantreg` package

4.2.1 Coding Philosophy

Where possible, the style of code writing is oriented at the `emdi` package. Thus, important steps in all functions are again divided into "sub-functions". Two advantages come along with this proceeding: Firstly, the readability of the code can be facilitated, because loops and programming details are not distracting from the sequence of implemented steps. Secondly, it can be beneficial for the performance, because only those objects are returned from the sub-functions that are needed at later points, since calculation steps are only stored temporarily.

A small difference to the `emdi` package until now is, that at some points in the package `data.table` (cf. Dowle and Srinivasan, 2017) is used, which is supposed to be faster than `data.frame`. Because the grid calculation requires to store, order and match possibly large data frames, `data.table` can be considered to show higher performance here. Note that the notation of `data.table` is different than that of `data.frame`. To further speed up the code, an option "sparse" is included in the the `mq` function, which can be used to suppress some results not important for the `mmqm` procedure.

The functions `mmqm` and `mq` return results of their own class. For these functions, S3 methods are created where applicable. Argument checks are included to ensure that the correct input is given.

The `mq_sae` results are of class `emdi, mq_model`. This allows that the new function can be almost seamlessly integrated in the `emdi` package, and important functions of this package like map plotting or returning indicators already work with the objects returned from the `mq_sae` function.

As one example, the `mq_sae` function (without the `roxygen2` block) is given in the appendix. For the remaining functions, please refer to the digitally attached package.

Additionally, help files based on `roxygen2` (cf. Wickham et al., 2017) and a user manual are created. An excerpt of the manual is now presented, and shows the functions and their arguments in detail (in alphabetical order, S3 methods in the end).

4.2.2 Functions and Arguments

`mmqm` *Pseudo Mixed M-quantile Models*

Description

Fit (pseudo) linear mixed M-quantile models to calculate pseudo random effects. These quantities are obtained based on the finding the average M-quantile regression plane in each domain. This non-parametric 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 ~ x1 + x2 + ...
`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 `seq(0.001,0.999,0.05)`. The 0.5 M-quantile is always calculated.
`...`	additional arguments to be passed to the mq function, e.g. the tuning constant for psi.huber.

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

`formula` a formula of the form y ~ x1 + x2 + ...

`data` a data frame containing the variables in the model.

`k` tuning constant used for Huber proposal 2 loss function.

`t`	vector of tau values specifying the M-quantiles.
`w`	a vector of prior weights for each case (for the IWLS procedure).
`case.weights`	(optional) initial down-weighting for each case.
`var.weights`	(optional) initial variable weights.
`init`	(optional) initial values for the coefficients or a method to find initial values or the result of a fit with a coef component. Known methods are "ls" (the default) for an initial least-squares fit and "lts" for an unweighted least-trimmed squares fit with 200 samples.
`maxit`	the limit on the number of IWLS iterations.
`acc`	the accuracy for the stopping criterion.
`test.vec`	the stopping criterion is based on changes in this vector.
`sparse`	returns less results (to save RAM).

Value

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

`fitted.values`	matrix of fitted values per M-quantile.
`residuals`	matrix of residuals per M-quantile.
`tau.values`	tau values for which regressions were run.
`coefficients`	matrix of coefficients per M-quantile.
`call`	the formula that was called.
`iterations`	number of iterations until convergence per M-quantile.
`scale`	results from scale estimation per M-quantile.
`iterations`	number of iterations until convergence per M-quantile.
`x`	model-matrix of the independent variables.
`y`	vector of the dependent variable.
`classes`	classes of the independent variables (needed for plotting).
`r_2`	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` *Sample M-quantiles*

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`

`expectile`

`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 small area 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

`fixed`	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 ~ operator and the explanatory variables on the right, separated by + operators. The argument corresponds to the argument `fixed` in function `mmqm`.
`pop_data`	a data frame that needs to comprise the variables named on the right of the ~ operator in `fixed`, i.e. the explanatory variables, and `pop_domains`.
`pop_domains`	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 `smp_domains`.
`smp_data`	a data frame that needs to comprise all variables named in `fixed` and `smp_domains`.
`smp_domains`	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 `pop_domains`.
`L`	a number determining the number of Monte-Carlo simulations. Defaults to 50.
`threshold`	a number defining a threshold. Alternatively, a threshold may be defined as a `function` of `y` returning a numeric value. Such a function will be evaluated once for the point estimation and in each iteration of the none-parametric bootstrap. A threshold is needed for calculation e.g. of head count ratios and poverty gaps. The argument defaults to `NULL`. 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 *Social Protection Committee 2001*). However, any desired threshold can be chosen.
`MSE`	if `TRUE`, MSE estimates using a none-parametric bootstrap approach are calculated (see also *Marchetti et al. (2012)*). Defaults to `FALSE`.
`B`	a number determining the number of bootstrap populations in the none-parametric bootstrap approach (see also *Marchetti et al. (2012)*) used in the MSE estimation. Defaults to 10.

`S` a number determining the number of samples taken from each bootstrap population (see also *Marchetti et al. (2012)*) 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 mmqm 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 none-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", "mq_model" that provides estimators for regional disaggregated indicators and optionally corresponding MSE estimates. Generic functions such

as `estimators`, `print` 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
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 function as threshold
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",
threshold = function(y){0.6 * median(y)}, transformation = "log",
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 mixed M-quantile model. 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

```
## S3 method for class 'mmqm'
plot(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

```
## S3 method for class 'mq'
plot(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` *Predict method for Pseudo Mixed M-quantile Model Fits*

Description

Predicted values for class "mmqm".

Usage

```
## S3 method for class 'mmqm'
predict(object, newdata, na.action = na.pass, ...)
```

Arguments

`object`	mmqm object, a result of `mmqm`.
`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 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 *Predict method for Linear M-quantile Model Fits*

Description

Predicted values for class "mq".

Usage

```
## S3 method for class 'mq'
predict(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, 76171. Chambers, R. and N. Tzavidis (2006): M-quantile models for small area estimation, Biometrika, 93, 255268.

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))
new <- data.frame(x = seq(-3, 3, 0.5))
predict(mq(y ~ x, data=df), new)
```

`print.emdi` *Prints an emdiObject*

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 approach (see also `mq_sae`).
`...`	optional arguments passed to `print.default`.

See Also

`emdiObject`, `ebp`, `mq_sae`

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 behavior 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
```

`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 behavior for the mmqm object.

References

Breckling, J. and Chambers, R. (1988): "M-quantiles", Biometrika, 75, 76171. Chambers, R. and N. Tzavidis (2006): M-quantile models for small area estimation, Biometrika, 93, 255268.

See Also

`mmqm`

Examples

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

4.2.3 Suggestions for Code-Improvements and Extensions

Some possible improvements and extensions that were not in the focus of this thesis or require additional programming skills are suggested now to facilitate the improvement of the package for third persons:

- A major improvement would be to translate the `mq` and `mmqm` functions to `c++`, as these functions are the bottleneck for the `mq_sae` function, especially when bootstrapping is done.
- The inclusion of more diagnostic plots could increase user friendliness. For `mq_sae` there is currently no diagnostic plot implemented.
- The code is prepared for the integration of some further functionality:
  - Smoothing for the error distributions in the `mq_sae` procedure (see Marchetti et al. (2012)): This part is currently disabled, because there appears to issues with the `np` package, that the S.A.M.P.L.E project used for smoothing. However, the authors note that the smoothed and unsmoothed results are similar anyways.
  - Inclusion of other loss functions than the Huber propososal 2 or other estimators of scale: These functions are commented out at the moment, because Chambers and Tzavidis (2006) noted that the choice of other loss function has no beneficial impact on the pseudo random effects calculation. In addition, the Huber proposal 2 function is found to be computationally advantageous. Furthermore, the exclusion makes the code more readable, and additional checks of arguments would be required if other loss functions are included. Also, case weights do not work correctly with these functions, as highlighted in the sourcecode by the authors of the `rlm` function.

Naturally, there are many other functions like variance estimation that can or should be added depending on the further use of the package.

In the next section, replications and other simulation results verify the correctness of the implementation and the performance of the M-quantile approach in small area estimation.

5 Monte-Carlo Simulations

In this section, various Monte-Carlo simulation studies are presented. Firstly, the implemented R-functions are tested based on a replication of some results of Marchetti et al. (2012). In the second section, the performance of the MQ (SAE) approach versus the EBP and direct estimation is investigated. Most simulations were run on the compute servers of the *Humboldt Lab for Empirical and Quantitative Research* (https://leqr.wiwi.hu-berlin.de).

5.1 Replication of Marchetti et al. (2012)

Marchetti et al. (2012) provide several simulations for different indicators. For the estimation of HCR and PG they use the Monte-Carlo approximation described in section 3. For quantiles and mean the authors use however other algorithms (see section 4.1). This is why these indicators are not considered in the replication, but in other simulation studies reported in section 5.2.

5.1.1 Simulation Setup

Following Marchetti et al. (2012), two random intercept scenarios (Normal and χ^2) are implemented:

Scenario 1. *Normal scenario*

$$
\begin{aligned}
y_{ij} &= 3000 - 150 * x_{ij} + \vartheta_j + \epsilon_{ij} \\
\vartheta_j &\sim N(0, 200^2) \\
\epsilon_{ij} &\sim N(0, 800^2) \\
x_{ij} &\sim N(\mu_j, 1) \\
\mu_j &\sim U[4, 10]
\end{aligned}
$$

Scenario 2. χ^2 *scenario*

$$
\begin{aligned}
y_{ij} &= 11 - x_{ij} + \vartheta_j + \epsilon_{ij} \\
\vartheta_j &\sim \chi(1) \\
\epsilon_{ij} &\sim \chi(6) \\
x_{ij} &\sim N(\mu_j, 1) \\
\mu_j &\sim U[8, 11]
\end{aligned}
$$

Error terms are centered to mean 0

The mean values μ_j as well as the number of cases n_j and N_j in each of 30 areas is drawn at random in the beginning and held fixed.

The authors run each setup with a small and large population/sample size:

For $\lambda = 1$, $N = 8400$, $150 \leq N_j \leq 440$, $n = 840$, $15 \leq n_j \leq 44$.

For $\lambda = 2$, $N = 840$, $50 \leq N_j \leq 150$, $n = 282$, $5 \leq n_j \leq 15$.

Due to computational reasons, only the results for the smaller population/sample ($\lambda = 2$) are replicated.

$H = 500$ simulations are run. In each run, a population of N cases is generated with N_j cases in each area, from which a stratified random sample over 30 areas (without replacement) is drawn based on the fixed sample sizes n_j.

Based on the Monte-Carlo procedure described in 3.3.2, point estimates for the HCR and PG are retrieved in each domain j. The number of iterations for the Monte-Carlo approximation is set to $L = 50$. For the MSE estimation, only $B = 1$ bootstrap population is generated, from which $S = 400$ bootstrap samples are drawn. The authors argue, that the 500 populations that are generated over the course of the simulation are imitating the generation of individual bootstrap populations. Hence, the computationally very expensive step of generating additional bootstrap populations is not necessary (cf. Marchetti et al., 2012, p. 2895).

With regard to the point estimates, the authors provide a graphical comparison of the point estimates in each area, averaged over the 500 simulations, to the averaged true values for the smaller samples.

For the mean squared error, some results in tabular form are provided. In these tables, the quality of the bootstrap MSE is assessed with the relative bias (RB) and the root mean squared error (RMSE). Let ν_j denote the true value for an indicator and $\hat{\nu}_j$ the corresponding estimate. Over H Monte-Carlo simulations, the RB is estimated with

$$RB(\hat{\nu}_j) = \frac{1}{H} \sum_{h=1}^{H} \left(\frac{\hat{\nu}_{j,h} - \nu_j}{\nu_j} \right) \tag{42}$$

and it provides a measure of the deviation from the estimated value relative to the true value ranging from $-\infty$ to ∞. The RMSE is given by

$$RMSE(\hat{\nu}_j) = \left[\frac{1}{H} \sum_{h=1}^{H} (\hat{\nu}_{j,h} - \nu_j)^2 \right]^{0.5} \tag{43}$$

and it is the root of the mean squared deviation from the true value ranging from 0 to ∞. (cf. Marchetti et al., 2012, p. 2894).

Remark 11. *The "true" value corresponding to the MSE/RMSE estimates is the* ***empirical*** *MSE/RMSE over the Monte-Carlo simulations.*

Note that the authors report the results from the smoothed approach (with smoothing

of the error distribution). The here reported results are the unsmoothed results (see remark, 10). However, both approaches are supposed to return consistent results (cf. Marchetti et al., 2012, p. 2895). Regarding the sampling from the errors in the Monte-Carlo procedure, the unconditional approach is used in the original and replicated results (see remark 9).

5.1.2 Point Estimation Results

The point estimation results can be replicated. The original graph is displayed in figure 8, the replication results in figure 9. Marchetti et al. (2012) find that the estimated values are very close to the true values, which is also matched by the results presented here. In the original results the estimated values are in most cases slightly lower than the true values. This slight underestimation is also found in the replication results.

Note that the true values in the areas differ between the original and replicated results, because they result from differences in the initial random sampling of μ_j, however the original and replicated values are in the same y-axis range.

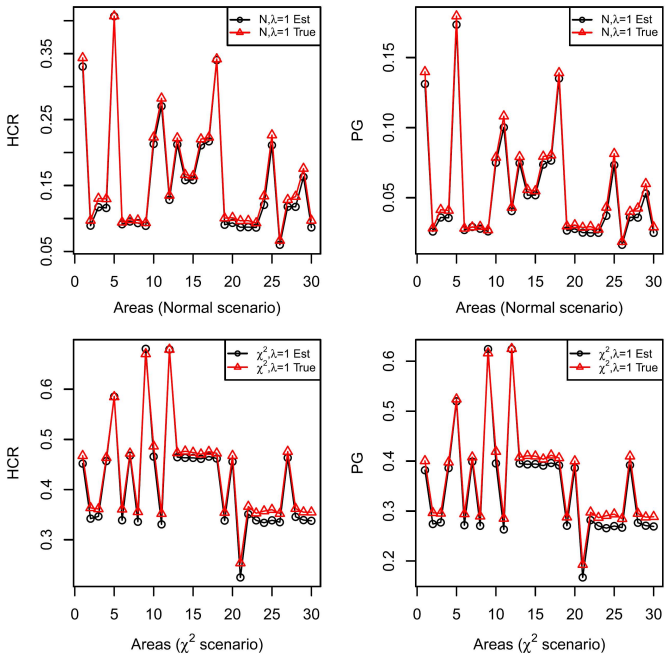

Figure 8: True vs. estimated point estimation results (Marchetti et al., 2012, p. 2897)

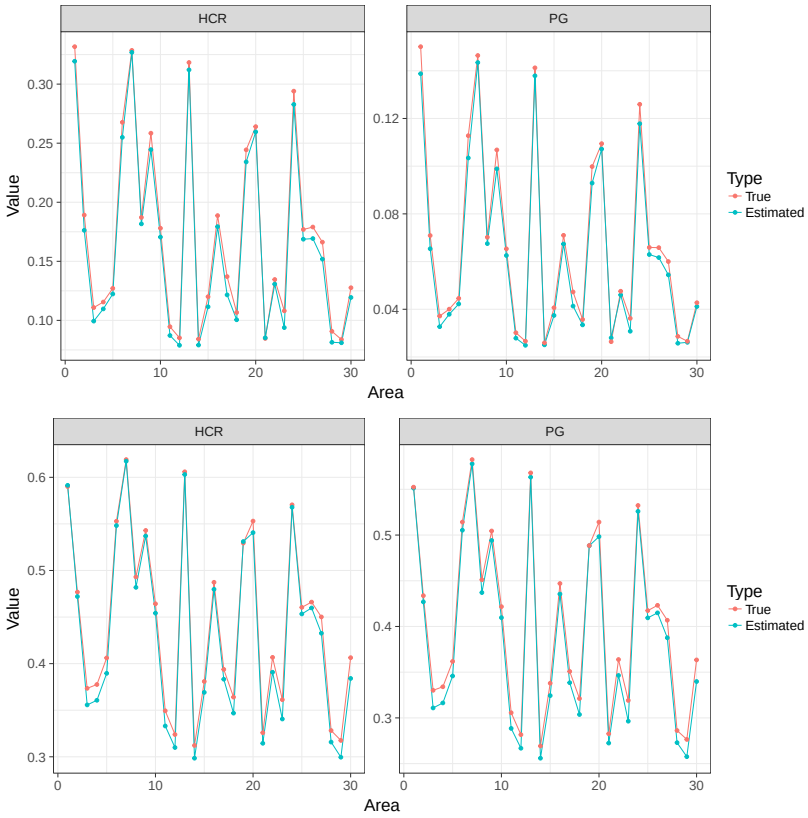

Figure 9: True vs. estimated point estimation results (replication)

5.1.3 MSE Estimation Results

The MSE results can also be replicated. Table 4 shows two columns for the HCR and two columns for the PG with the original and replicated results for the MSE estimation. Note that the values are averaged over the replications and then over the areas.

For the *Normal scenario* original and replicated results show a very good match. On average, the estimated values (original 0.059 vs. replicated 0.063) are very close to the true values (original 0.063 vs. replicated 0.065), which also applies to the relative bias. For the PG, the difference between the original and replicated results is slightly larger for the relative bias, however, with about 1.5 percentage points for a relative measure this deviation is still very small and can be attributed to differences in the random sampling of the sample sizes and/or mean values. For the RMSE the replicated values also match the original values (original 0.022/0.012 vs. replicated 0.022/0.014).

For the χ^2 *scenario* the replication was also successful. Regarding the HCR, the true (original 0.096 vs. replicated 0.098), as well as the estimated values (original 0.096 vs. replicated 0.099) are very close to each other, which also applies to the relative bias which deviates

only by around 2.4 percentage points. For the poverty gap, the true and estimated values show the same behavior, and the relative bias is again slightly larger, however, with about 3.25 percentage points the difference is also small and is again likely to result from the initial random sampling. Regarding the RMSE the replicated values are again almost identical (original 0.019/0.019 vs. replicated 0.021/0.022).

	HCR	HCR	PG	PG
χ^2 scenario	Original	Replicated	Original	Replicated
True	0.096	0.098	0.094	0.097
Estimated	0.096	0.099	0.095	0.099
Rel. Bias (%)	0.19	2.58	0.26	3.49
RMSE	0.019	0.021	0.019	0.022
	HCR	HCR	PG	PG
Normal scenario	Original	Replicated	Original	Replicated
True	0.063	0.065	0.028	0.033
Estimated	0.059	0.063	0.025	0.031
Rel. Bias (%)	-7.66	-7.82	-12.06	-10.47
RMSE	0.022	0.022	0.012	0.014

Table 4: Comparision of original and replicated RMSE estimation results

Given the successful replication results, the new functions can now be used to evaluate the MQ (SAE) approach in comparison to the EBP under different Monte-Carlo experiments. This is the content of the next section.

5.2 Comparision of direct, EBP and MQ (SAE) Point Estimation

In this section, Monte-Carlo simulations are presented, that evaluate the point estimation of EBP, MQ (SAE) and direct estimation. There are four Monte-Carlo scenarios implemented, all based on random intercept models with normally distributed area effects. The scenarios are the following:

- normally distributed individual error terms with truncated dependent variable (normal errors scenario)
- normally distributed individual error terms for the logarithm of the dependent variable (log-scale outcomes scenario)
- Pareto distributed individual error terms (Pareto errors scenario)
- contaminated normally distributed individual error terms with truncated dependent variable (contaminated normal errors scenario)

In all scenarios area sizes are fixed to $N_j = 200 \forall j$ in the population respectively $8 \leq n_j \leq 29$ in the sample, where $j = 1, \ldots, 50$.

The following process is repeated 500 times: In each run, a new population is generated according to the defined scenario. A stratified random sample without replacement of n_j per area is drawn. Based on the population data, the true value is calculated. Using the sample data, the following estimators are calculated:

- the direct estimator
- the EBP, with L=50 Monte-Carlo iterations and Box Cox transformation
- the MQ (SAE), with L=50 Monte-Carlo iterations[5]

For the economy of space, the analysis is limited to six indicators: Gini coefficient, HCR, PG, mean, median and $Q_{0.25}$. The bias is reported for Gini coefficient, HCR, PG, which is over H Monte-Carlo simulations estimated with

$$Bias(\hat{\nu}_j) = \frac{1}{H} \sum_{h=1}^{H} (\hat{\nu}_{j,h} - \nu_j) \,. \tag{44}$$

[5]The used grid is given by $\{0.006, 0.010, 0.020, 0.051, 0.096, 0.141, 0.186, 0.231, 0.276, 0.321, 0.366, 0.411, 0.456, 0.500, 0.501, 0.546, 0.591, 0.636, 0.681, 0.726, 0.771, 0.816, 0.861, 0.906, 0.951, 0.960, 0.980, 0.994\}$.

The relative bias is reported for mean, median, $Q_{0.25}$ and is calculated as in equation (42). As a measure of variation, the root mean squared error is reported for all indicators and is calculated like equation (43).

5.2.1 The Normal Errors Scenario

Scenario 3. *Normal errors*

$$
\begin{aligned}
y_{ij} &= max(4500 - 400 * x_{ij} + \vartheta_j + \epsilon_{ij}, 0) \\
x_{ij} &\sim N(\mu_j, 3^2) \\
\mu_j &\sim U[-3, 3] \\
\vartheta_j &\sim N(0, 500^2) \\
\epsilon_{ij} &\sim N(0, 1000^2)
\end{aligned}
$$

The box plots displayed in figure 10 display the distribution of bias/relative bias over the areas. Overall, the estimation is relatively unbiased for most indicators and estimators with a relative bias of on average less than 0.6% respectively an average bias of less then 0.004. Exceptions are $Q_{0.25}$, where the MQ (SAE) shows on average a negative bias of 1.5% and Gini coefficient, which is negatively biased if estimated directly (on average around -0.01). For direct estimation, there are also some outlying values, which means that in some areas the average bias/relative bias is higher.

The line plots of the bias/relative bias per area displayed in figure 11 additionally reveal, that in the individual areas the direct method is actually slightly biased for most areas, and only shows on average a low bias because the negative and positive biases even out. For $Q_{0.25}$ and Gini coefficient the direct estimator is even substantially biased in some areas with up to 6

In the box plots in figure 12 the distribution of the RMSE over the areas is displayed. Evidently, the direct method performs worse than EBP and MQ (SAE) as it has considerable amount of variation. The EBP has a slightly better performance than the MQ (SAE) for all indicators.

Regarding the areas, the line plots displayed in figure 13 additionally reveal, that the MQ (SAE) and EBP method show a similar behavior, as low and high deflections occur in the same areas.

Overall, the EBP method is least biased and most efficient, followed relative closely by the

MQ (SAE) method. The direct estimator does not provide desirable results at area level, as it is rather inefficient. These results are in line with the expectations, because the Gaussian assumptions for the EBP are fulfilled for both error terms. They also provide evidence, that both model based approaches successfully borrow strength especially in the smaller samples. The loss in efficiency by using the MQ (SAE) is in fact moderate.

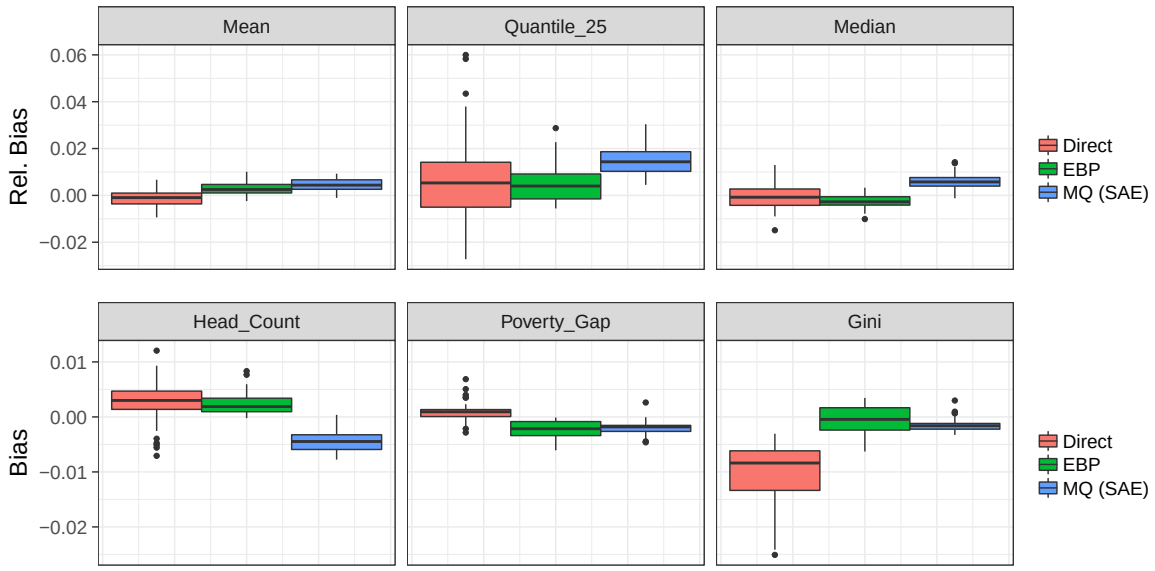

Figure 10: Distribution of bias/rel. bias of direct, EBP and MQ (SAE) point estimation results over the areas in the normal errors scenario

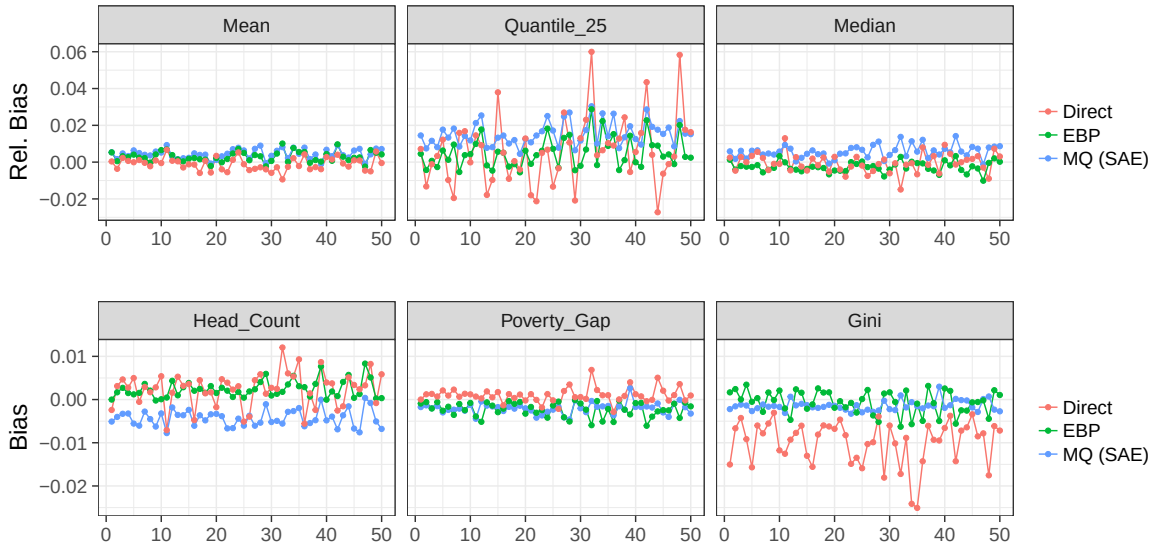

Figure 11: Bias/rel. bias per area of direct, EBP and MQ (SAE) point estimation results in the normal errors scenario

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Mean	MQ (SAE)	-0.001	0.003	0.004	0.004	0.007	0.009
Mean	EBP	-0.002	0.001	0.003	0.003	0.005	0.010
Mean	Direct	-0.009	-0.004	-0.001	-0.001	0.001	0.007
Median	MQ (SAE)	-0.001	0.004	0.006	0.006	0.008	0.014
Median	EBP	-0.010	-0.004	-0.003	-0.003	-0.001	0.003
Median	Direct	-0.015	-0.004	-0.001	-0.000	0.003	0.013
Quantile_25	MQ (SAE)	0.004	0.010	0.014	0.015	0.019	0.030
Quantile_25	EBP	-0.006	-0.001	0.004	0.005	0.009	0.029
Quantile_25	Direct	-0.027	-0.005	0.005	0.006	0.014	0.060
Gini	MQ (SAE)	-0.003	-0.002	-0.002	-0.001	-0.001	0.003
Gini	EBP	-0.006	-0.002	-0.000	-0.001	0.002	0.003
Gini	Direct	-0.025	-0.013	-0.008	-0.010	-0.006	-0.003
Head_Count	MQ (SAE)	-0.008	-0.006	-0.004	-0.004	-0.003	0.000
Head_Count	EBP	-0.000	0.001	0.002	0.002	0.003	0.008
Head_Count	Direct	-0.007	0.001	0.003	0.002	0.005	0.012
Poverty_Gap	MQ (SAE)	-0.005	-0.003	-0.002	-0.002	-0.002	0.003
Poverty_Gap	EBP	-0.006	-0.003	-0.002	-0.002	-0.001	-0.000
Poverty_Gap	Direct	-0.003	0.000	0.001	0.001	0.001	0.007

Table 5: Summary statistics over the areas for bias (upper half)/rel. bias (lower half) of direct, EBP and MQ (SAE) point estimation results in the normal errors scenario

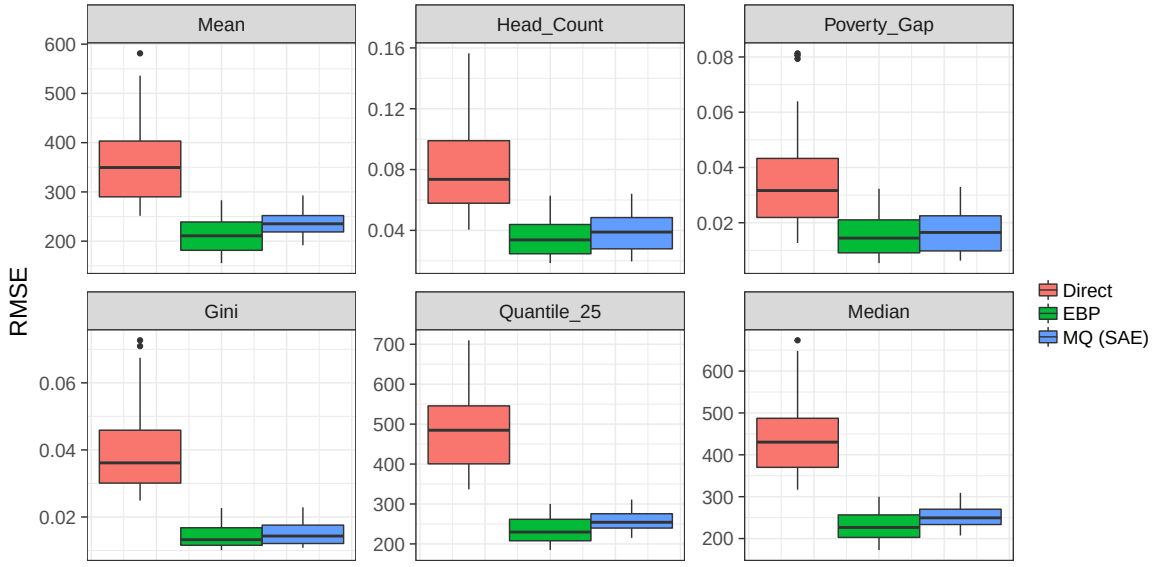

Figure 12: Distribution of RMSE of direct, EBP and MQ (SAE) point estimation results over the areas in the normal errors scenario

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Mean	MQ (SAE)	191.64	218.71	235.23	236.74	251.91	292.92
Mean	EBP	155.39	181.41	210.82	212.43	238.91	283.03
Mean	Direct	251.51	289.97	349.50	361.43	403.28	581.34
Median	MQ (SAE)	207.36	233.50	249.46	251.41	270.01	309.13
Median	EBP	172.49	202.72	226.50	229.95	256.50	299.87
Median	Direct	316.59	370.08	430.32	446.15	487.23	673.47
Quantile_25	MQ (SAE)	215.21	239.68	254.24	256.86	275.56	311.06
Quantile_25	EBP	184.59	207.85	229.66	235.31	261.68	300.19
Quantile_25	Direct	336.61	400.56	484.71	487.13	545.69	709.89
Gini	MQ (SAE)	0.011	0.012	0.014	0.015	0.018	0.023
Gini	EBP	0.010	0.012	0.013	0.014	0.017	0.023
Gini	Direct	0.025	0.030	0.036	0.039	0.046	0.073
Head_Count	MQ (SAE)	0.020	0.028	0.039	0.039	0.048	0.064
Head_Count	EBP	0.018	0.025	0.034	0.036	0.044	0.063
Head_Count	Direct	0.040	0.058	0.074	0.080	0.099	0.156
Poverty_Gap	MQ (SAE)	0.006	0.010	0.016	0.017	0.023	0.033
Poverty_Gap	EBP	0.005	0.009	0.014	0.016	0.021	0.032
Poverty_Gap	Direct	0.013	0.022	0.032	0.035	0.043	0.081

Table 6: Summary statistics over the areas for the RMSE of direct, EBP and MQ (SAE) point estimation results in the normal errors scenario

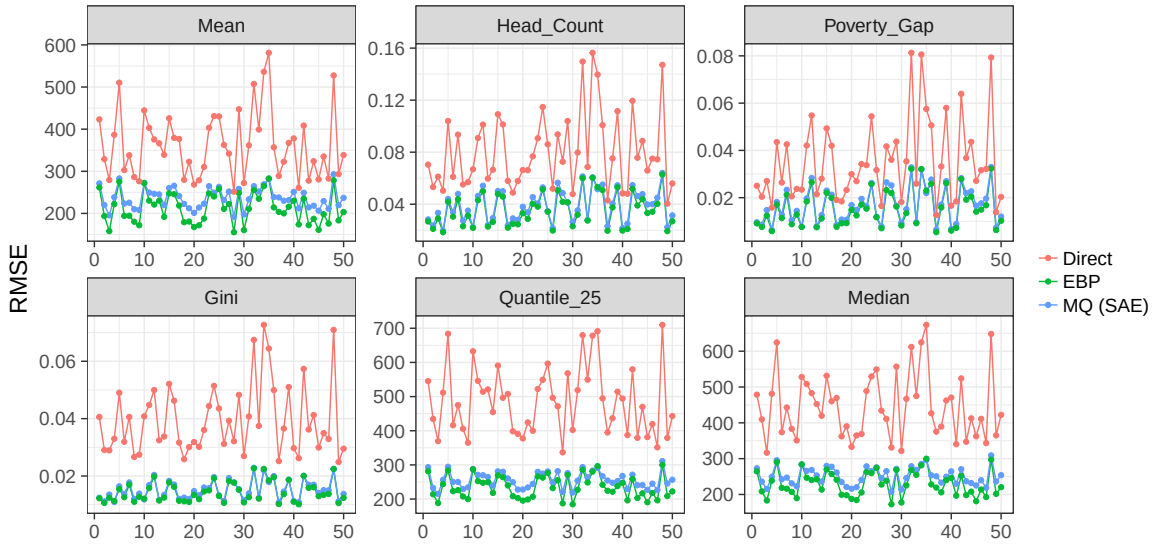

Figure 13: RMSE per area of direct, EBP and MQ (SAE) point estimation results in the normal errors scenario

5.2.2 The Log-scale Outcomes Scenario

Scenario 4. *Log-scale outcomes*

$$
\begin{aligned}
y_{ij} &= exp\{10 - x_{ij} + \vartheta_j + \epsilon_{ij}\} \\
x_{ij} &\sim N(\mu_j, 0.5^2) \\
\mu_j &\sim U[2, 3] \\
\vartheta_j &\sim N(0, 0.4^2) \\
\epsilon_{ij} &\sim N(0, 0.8^2) \\
\text{Note :} & \\
log(y_{ij}) &= 10 - x_{ij} + \vartheta_j + \epsilon_{ij}
\end{aligned}
$$

The box plots in figure 14 display the distribution of bias/relative bias over the areas. There are mixed results regarding the average bias. The EBP is unbiased for HCR, PG and Gini coefficient, but slightly biased (around 5%) for quantiles and mean. The direct estimator is unbiased for mean, HCR and PG, but shows a relative bias of around 4%-5.4% elsewhere. The MQ (SAE) approach is substantially biased for all indicators up to an average relative bias of circa 20% for median.

The line plots of the bias/relative bias per area displayed in figure 15 reveal, that the direct method is rather biased with regard to the $Q_{0.25}$ in some areas. The MQ (SAE) results look like a shifted version of the EBP results. Hence, for both methods the estimation is more or less biased in the same areas, while it is generally more biased for the MQ (SAE).

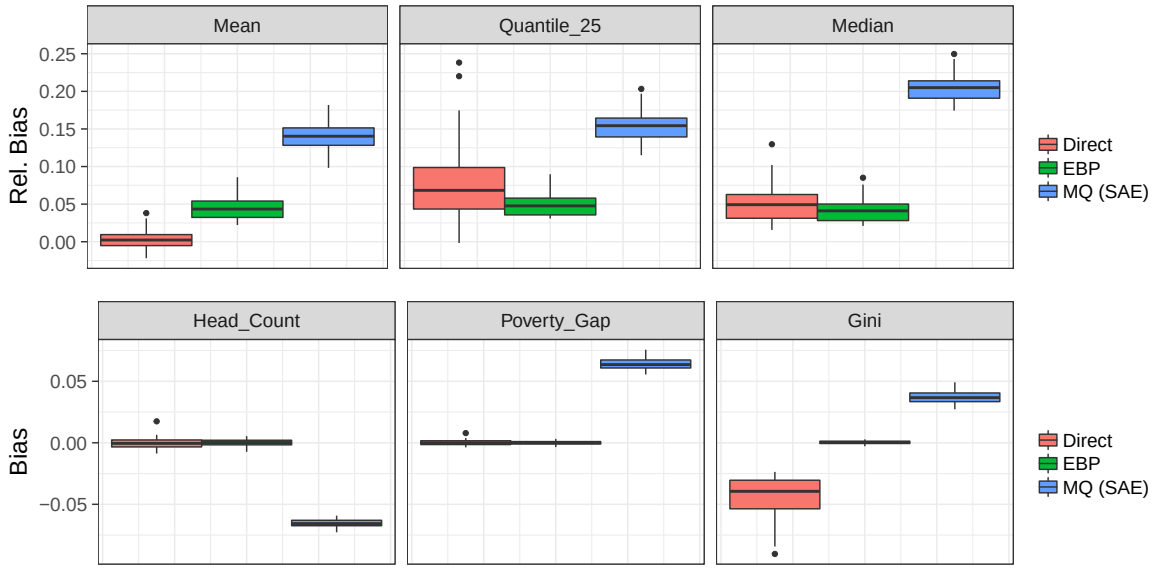

Figure 14: Distribution of bias/rel. bias of direct, EBP and MQ (SAE) point estimation results over the areas in the log-scale outcomes scenario

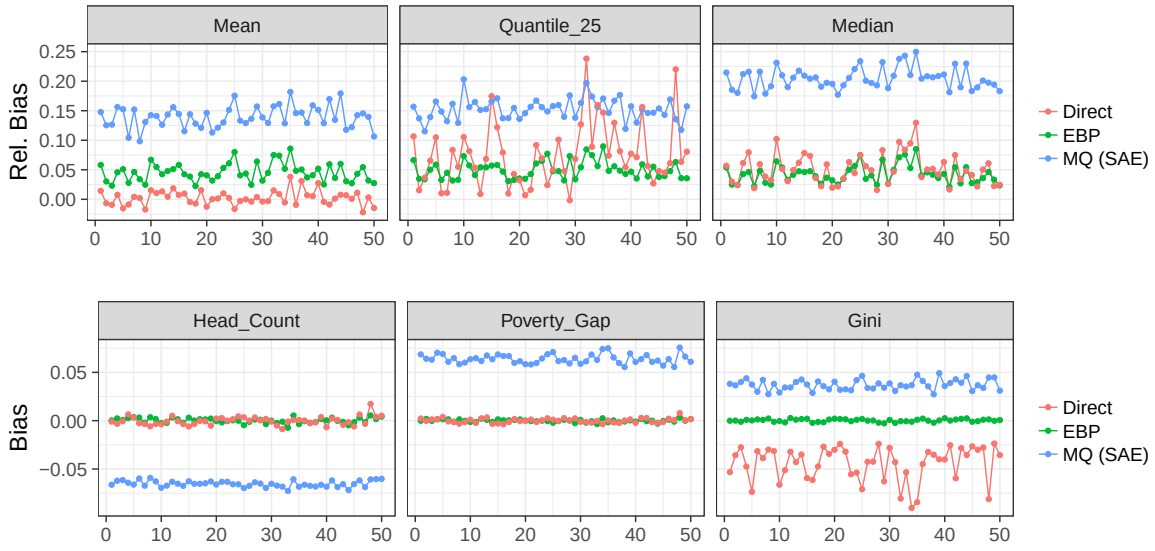

Figure 15: Bias/rel. bias per area of direct, EBP and MQ (SAE) point estimation results in the log-scale outcomes scenario

In figure 16 the distribution of the RMSE over the areas is laid out. The direct method and the MQ (SAE) have a considerable amount of variation, while the EBP is more efficient for all indicators.

The line plots displayed in figure 17 additionally disclose, that all methods show a similar behavior over the areas, that is low and high deflections in the same areas, while the EBP has a much smaller RMSE. Likely, this results from the different samples sizes in the domains,

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Mean	MQ (SAE)	0.098	0.128	0.140	0.139	0.151	0.182
Mean	EBP	0.022	0.032	0.043	0.045	0.054	0.086
Mean	Direct	-0.022	-0.005	0.002	0.003	0.009	0.038
Median	MQ (SAE)	0.174	0.191	0.205	0.205	0.214	0.250
Median	EBP	0.021	0.028	0.041	0.042	0.050	0.085
Median	Direct	0.016	0.031	0.049	0.051	0.063	0.130
Quantile_25	MQ (SAE)	0.115	0.139	0.154	0.153	0.164	0.203
Quantile_25	EBP	0.031	0.036	0.048	0.050	0.058	0.090
Quantile_25	Direct	-0.002	0.043	0.068	0.075	0.099	0.238
Gini	MQ (SAE)	0.027	0.033	0.037	0.037	0.041	0.049
Gini	EBP	-0.003	-0.001	0.000	0.000	0.001	0.003
Gini	Direct	-0.090	-0.054	-0.039	-0.044	-0.030	-0.024
Head_Count	MQ (SAE)	-0.073	-0.067	-0.066	-0.065	-0.063	-0.059
Head_Count	EBP	-0.007	-0.002	0.000	0.000	0.002	0.005
Head_Count	Direct	-0.009	-0.003	-0.000	-0.000	0.002	0.017
Poverty_Gap	MQ (SAE)	0.056	0.061	0.064	0.064	0.067	0.076
Poverty_Gap	EBP	-0.003	-0.001	-0.000	-0.000	0.001	0.003
Poverty_Gap	Direct	-0.004	-0.001	0.000	0.000	0.002	0.008

Table 7: Summary statistics over the areas for bias (upper half)/rel. bias (lower half) of direct, EBP and MQ (SAE) point estimation results in the log-scale outcomes scenario

that appear to affect all methods, albeit to a different extent.

Overall, the EBP method performs best regarding bias and efficiency, while neither the MQ (SAE) nor direct estimation provide desirable results at area level.

These results provide evidence that the Box Cox transformation implemented in the `emdi` package provides an advantage for the EBP method, when the true error distribution is only normal for a transformation of the dependent variable. The MQ (SAE) approach appears to be unable to handle the nonlinear relationship between the independent and untransformed dependent variables. Likely, this leads to a low explanatory power in the underlying regression models, which impairs its capability to "borrow strength".

Remark 12. *To check if the performance of the MQ (SAE) can be increased with a larger number of Monte-Carlo iterations, the log-scale scenario was additionally run with L=100 and L=150 iterations. However, the results are virtually identical regarding bias/rel. bias and RMSE.*

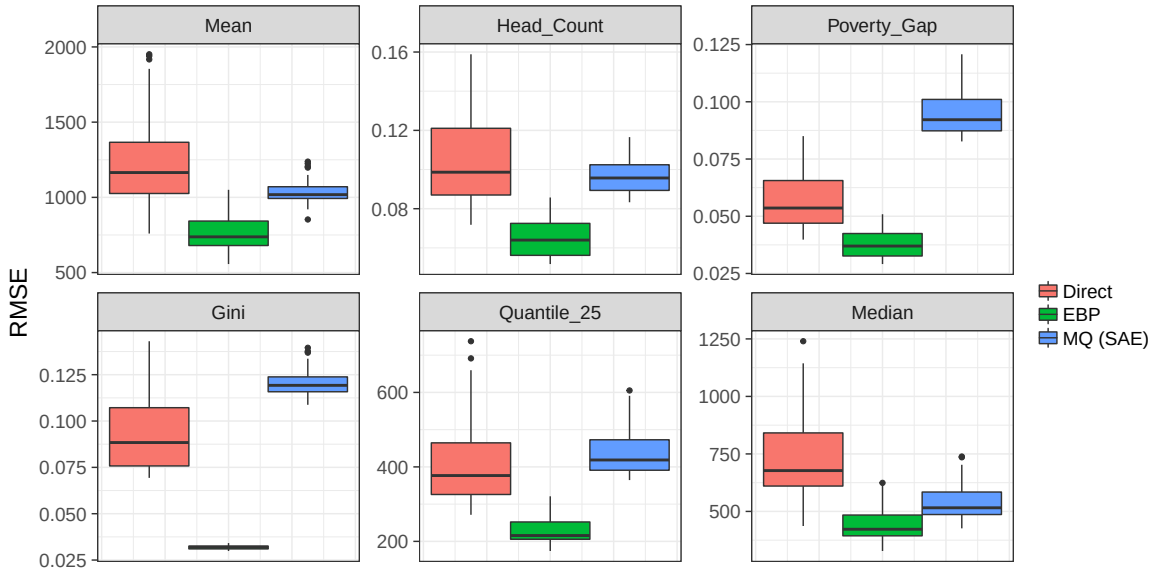

Figure 16: Distribution of RMSE of direct, EBP and MQ (SAE) point estimation results over the areas in the log-scale outcomes scenario

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Mean	MQ (SAE)	852.98	992.74	1018.16	1032.92	1070.70	1237.68
Mean	EBP	556.74	679.83	737.55	770.28	842.49	1050.20
Mean	Direct	759.49	1025.75	1164.89	1227.96	1366.16	1951.09
Median	MQ (SAE)	426.41	486.53	515.75	543.04	584.30	738.60
Median	EBP	327.90	393.89	422.50	447.22	484.25	624.17
Median	Direct	436.61	610.32	677.46	727.72	841.35	1239.88
Quantile_25	MQ (SAE)	364.60	391.02	418.56	444.79	472.81	605.24
Quantile_25	EBP	174.13	205.67	215.71	231.12	252.14	321.05
Quantile_25	Direct	271.36	325.91	376.75	411.08	464.56	737.41
Gini	MQ (SAE)	0.109	0.116	0.119	0.121	0.124	0.140
Gini	EBP	0.030	0.031	0.032	0.032	0.033	0.034
Gini	Direct	0.069	0.076	0.088	0.093	0.107	0.143
Head_Count	MQ (SAE)	0.083	0.089	0.096	0.096	0.102	0.117
Head_Count	EBP	0.052	0.056	0.064	0.065	0.073	0.086
Head_Count	Direct	0.072	0.087	0.099	0.104	0.121	0.159
Poverty_Gap	MQ (SAE)	0.083	0.087	0.092	0.095	0.101	0.121
Poverty_Gap	EBP	0.029	0.033	0.037	0.038	0.042	0.051
Poverty_Gap	Direct	0.040	0.047	0.054	0.056	0.066	0.085

Table 8: Summary statistics over the areas for the RMSE of direct, EBP and MQ (SAE) point estimation results in the log-scale outcomes scenario

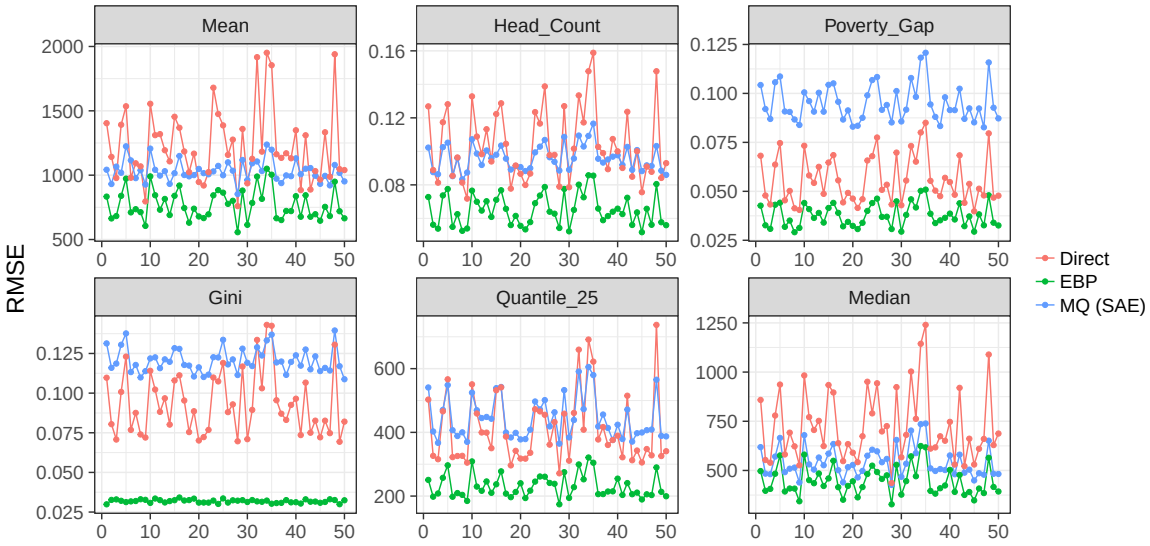

Figure 17: RMSE per area of direct, EBP and MQ (SAE) point estimation results in the log-scale outcomes scenario

5.2.3 The Pareto Errors Scenario

Scenario 5. *Pareto errors*

$$y_{ij} = 12000 - 400 * x_{ij} + \vartheta_j + \epsilon_{ij}$$
$$x_{ij} \sim N(\mu_j, 7.5^2)$$
$$\mu_j \sim U[-3, 3]$$
$$\vartheta_j \sim N(0, 500^2)$$
$$\epsilon_{ij} \sim \sqrt{2} * Pareto(scale = 2000, shape = 3)$$

$$\epsilon \textit{ is centered to expectation } 0$$

Figure 18 presents the distribution of bias/relative bias over the areas with some box plots. Overall, the estimation is on average relatively unbiased for most indicators with a relative bias of less than 1% respectively a bias near 0, with exception for $Q_{0.25}$ and median, where the EBP shows negative and positive bias of up to 2.7%. Additionally, the direct estimator is slightly biased for Gini coefficient with a value of circa -0.01. For direct estimation, there are also a few outlying values.

The line plots of the bias/relative bias per area displayed in figure 19 additionally show some outlying areas for EBP and direct estimation for the $Q_{0.25}$. The results for the MQ (SAE) are overall very good, and almost identical to the results of the normal scenario.

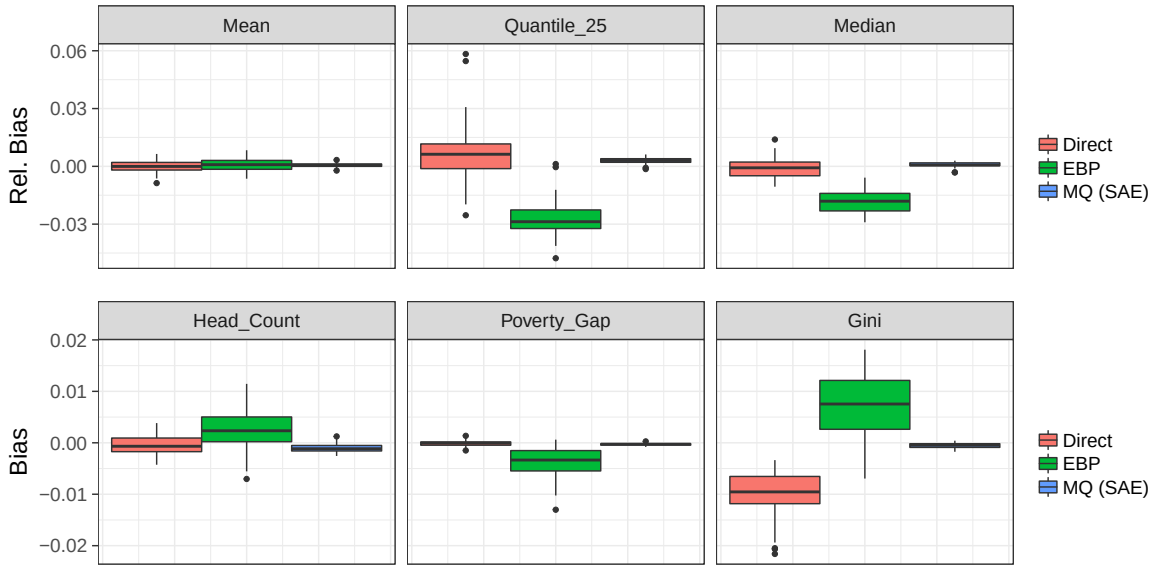

Figure 18: Distribution of bias/rel. bias of direct, EBP and MQ (SAE) point estimation results over the areas in the Pareto errors scenario

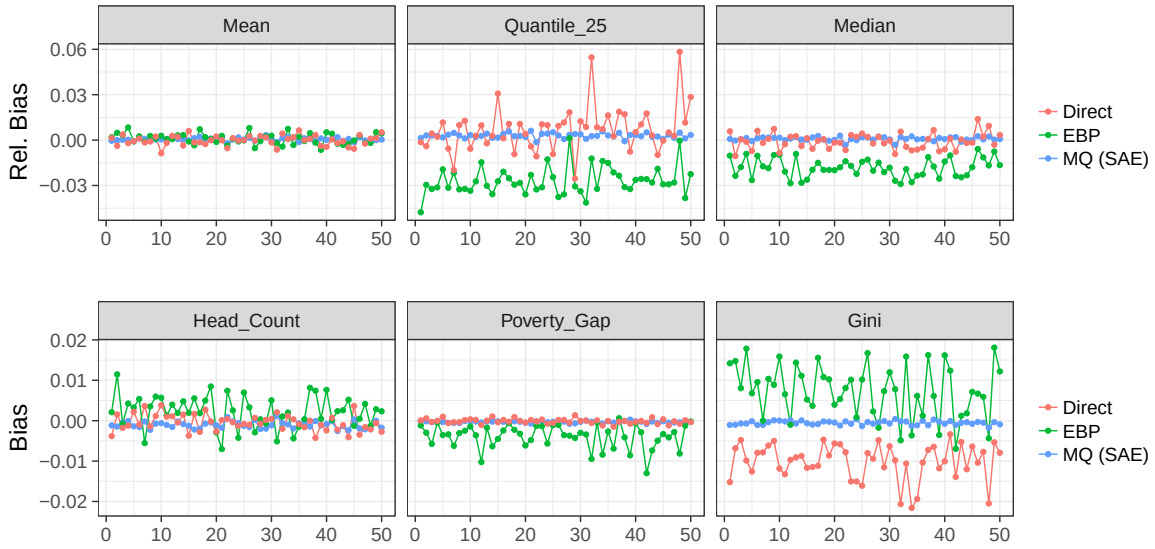

Figure 19: Bias/rel. bias per area of direct, EBP and MQ (SAE) point estimation results in the Pareto errors scenario

In figure 20 the distribution of the RMSE over the areas is displayed. Evidently, the direct method is very inefficient compared to the EBP and MQ (SAE) as it has a considerable amount of variation. Contrary to the previous scenarios, the MQ (SAE) approach now has a slightly better performance than the EBP. The results for MQ (SAE) are in fact comparable to those in the normal scenario.

The line plots displayed in figure 21 additionally reveal, that the MQ (SAE) and EBP

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Mean	MQ (SAE)	-0.002	-0.000	0.001	0.001	0.001	0.003
Mean	EBP	-0.006	-0.002	0.001	0.001	0.003	0.008
Mean	Direct	-0.009	-0.002	-0.000	-0.000	0.002	0.006
Median	MQ (SAE)	-0.003	0.000	0.001	0.001	0.002	0.003
Median	EBP	-0.029	-0.023	-0.018	-0.018	-0.014	-0.006
Median	Direct	-0.011	-0.005	-0.001	-0.001	0.002	0.014
Quantile_25	MQ (SAE)	-0.001	0.002	0.003	0.003	0.004	0.006
Quantile_25	EBP	-0.048	-0.032	-0.029	-0.027	-0.023	0.001
Quantile_25	Direct	-0.025	-0.001	0.006	0.007	0.012	0.058
Gini	MQ (SAE)	-0.002	-0.001	-0.000	-0.001	-0.000	0.000
Gini	EBP	-0.007	0.003	0.008	0.007	0.012	0.018
Gini	Direct	-0.022	-0.012	-0.010	-0.010	-0.007	-0.003
Head_Count	MQ (SAE)	-0.003	-0.002	-0.001	-0.001	-0.001	0.001
Head_Count	EBP	-0.007	0.000	0.002	0.002	0.005	0.011
Head_Count	Direct	-0.004	-0.002	-0.001	-0.000	0.001	0.004
Poverty_Gap	MQ (SAE)	-0.001	-0.000	-0.000	-0.000	-0.000	0.000
Poverty_Gap	EBP	-0.013	-0.005	-0.003	-0.004	-0.002	0.001
Poverty_Gap	Direct	-0.001	-0.000	-0.000	-0.000	0.000	0.001

Table 9: Summary statistics over the areas for bias (upper half)/rel. bias (lower half) of direct, EBP and MQ (SAE) point estimation results in the Pareto errors scenario

method follow a similar behavior over the areas, that is low and high deflections in the same areas, while the EBP generally shows a higher RMSE. For direct estimation, there are some rather extreme outliers.

Overall, the MQ (SAE) method performs best in terms of bias and efficiency, followed closely by the EBP. The direct estimator does not provide desirable results at area level. These results can be interpreted such that the EBP is not able to handle long tailed distribution like the Pareto distribution as well as the MQ approach. Contrary to the log-scale scenario, the Pareto distribution can not be as easily transformed to a normal distribution. In addition, for the MQ (SAE) approach the large values that result from the long tail of the Pareto distribution do not affect its estimation as much, because the tuning constant in the Huber loss function limits their influence. Still, both model based approaches successfully borrow strength in the smaller samples and should be preferred over the direct estimator.

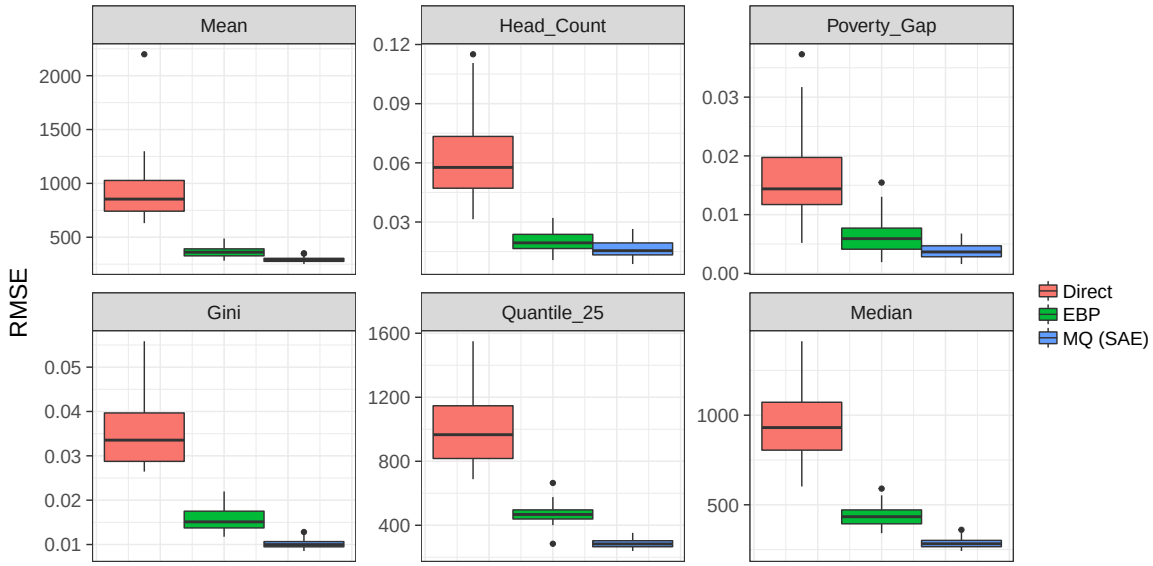

Figure 20: Distribution of RMSE of direct, EBP and MQ (SAE) point estimation results over the areas in the Pareto errors scenario

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Mean	MQ (SAE)	250.96	275.86	289.60	292.76	304.01	349.90
Mean	EBP	281.68	327.09	360.83	362.97	392.09	488.09
Mean	Direct	630.99	741.87	854.12	912.07	1027.18	2198.69
Median	MQ (SAE)	242.01	265.96	282.50	285.78	301.28	360.48
Median	EBP	341.58	393.81	433.46	435.97	471.32	590.41
Median	Direct	602.28	804.95	930.98	955.08	1072.30	1412.85
Quantile_25	MQ (SAE)	239.15	265.74	283.28	285.90	303.45	352.54
Quantile_25	EBP	284.43	439.17	467.43	472.79	496.23	664.41
Quantile_25	Direct	687.97	817.30	965.94	1003.04	1146.83	1550.05
Gini	MQ (SAE)	0.009	0.009	0.010	0.010	0.011	0.013
Gini	EBP	0.012	0.014	0.015	0.016	0.018	0.022
Gini	Direct	0.026	0.029	0.034	0.035	0.040	0.056
Head_Count	MQ (SAE)	0.009	0.013	0.015	0.016	0.019	0.026
Head_Count	EBP	0.011	0.017	0.019	0.020	0.024	0.032
Head_Count	Direct	0.031	0.047	0.058	0.062	0.073	0.115
Poverty_Gap	MQ (SAE)	0.002	0.003	0.004	0.004	0.005	0.007
Poverty_Gap	EBP	0.002	0.004	0.006	0.006	0.008	0.015
Poverty_Gap	Direct	0.005	0.012	0.014	0.016	0.020	0.037

Table 10: Summary statistics over the areas for the RMSE of direct, EBP and MQ (SAE) point estimation results in the Pareto errors scenario

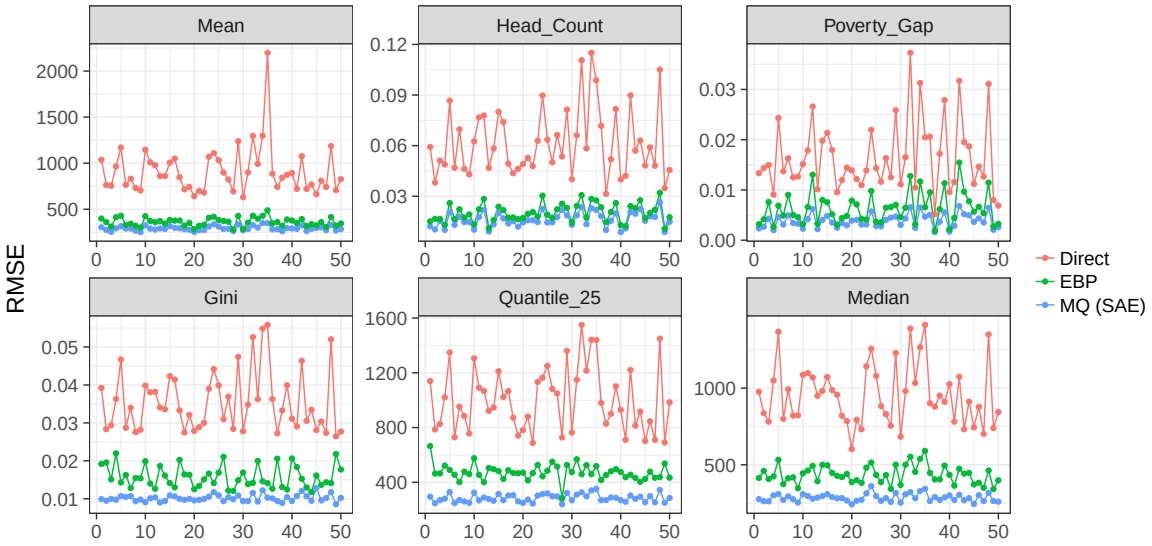

Figure 21: RMSE per area of direct, EBP and MQ (SAE) point estimation results in the Pareto errors scenario

5.2.4 The Contaminated Normal Errors Scenario

Scenario 6. *Contaminated normal errors*

$$
\begin{aligned}
y_{ij} &= max(4500 - 400 * x_{ij} + \vartheta_j + \epsilon_{ij}, 0) \\
x_{ij} &\sim N(\mu_j, 3^2) \\
\mu_j &\sim U[-3, 3] \\
\vartheta &\sim N(0, 500^2) \\
\epsilon_{ij} &\sim 0.95 * N(0, 1000^2) + 0.05 * N(0, 6000^2)
\end{aligned}
$$

The box plots displayed in figure 22 visualize the distribution of bias/relative bias over the areas. Overall, the estimation is on average relatively unbiased for the direct estimator and the MQ (SAE) with a relative bias of less than 1% respectively a bias near 0. Exceptions are Gini coefficient, where the bias of the direct estimator is absolutly seen with -0.013 slightly higher and the $Q_{0.25}$, where the relative bias of MQ (SAE) is around 2%. The EBP shows in particular a stronger negative relative bias of around 0.03-0.045 for $Q_{0.25}$, HCR and Gini coefficient. For direct estimation, there are also a few outlying values.

Based on the line plots of the bias/relative bias per area displayed in figure 23 some outlying areas for EBP and direct estimation for the $Q_{0.25}$ can be identified. The results for the MQ (SAE) are overall very good, and similar to the results of the normal scenario or

Pareto scenario.

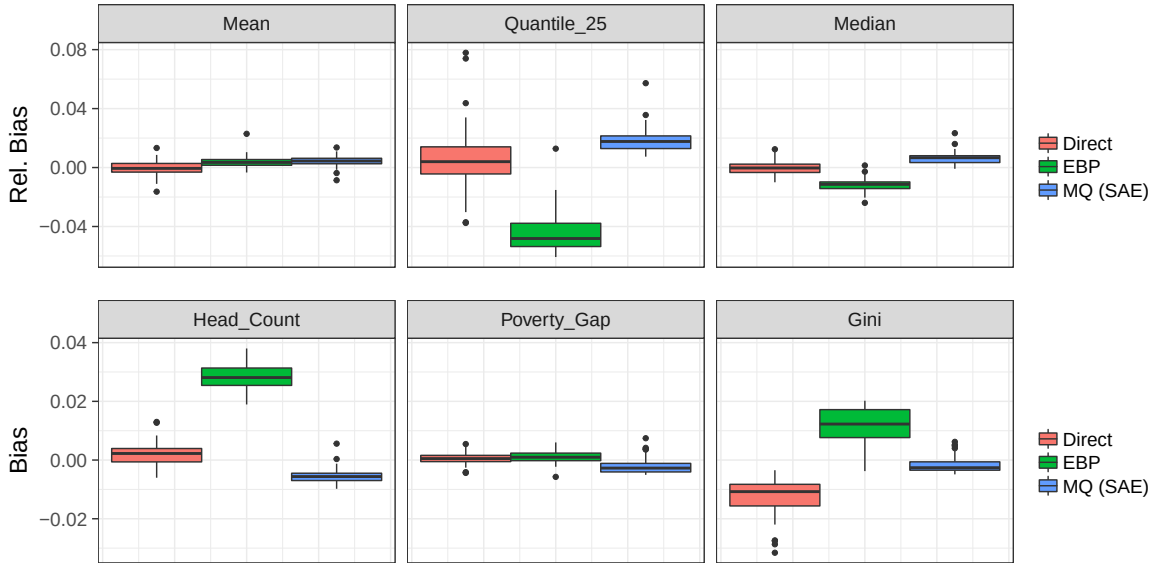

Figure 22: Distribution of bias/rel. bias of direct, EBP and MQ (SAE) point estimation results over the areas in the contaminated normal errors scenario

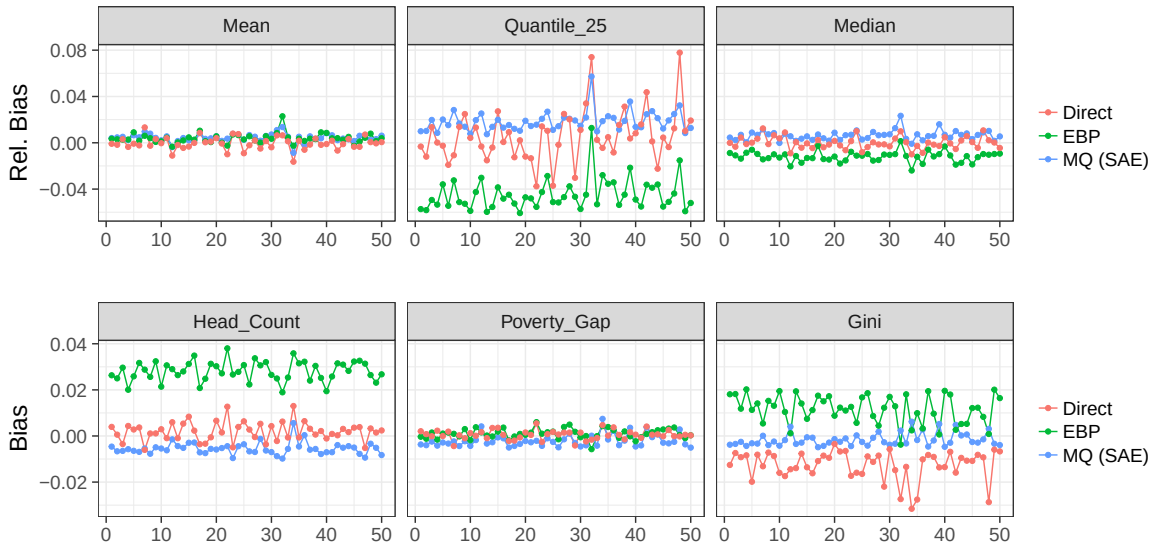

Figure 23: Bias/rel. bias per area of direct, EBP and MQ (SAE) point estimation results in the contaminated normal errors scenario

In figure 24 the distribution of the RMSE over the areas is exposed. Evidently, the direct method performs worse than EBP and MQ (SAE) as it has a considerable amount of variation. Now the MQ (SAE) approach has a slightly better performance than the EBP. Again, the results for MQ (SAE) are comparable to those in the normal scenario.

The line plots displayed in figure 25 additionally disclose, that the MQ (SAE) and EBP

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Mean	MQ (SAE)	-0.009	0.003	0.004	0.004	0.006	0.014
Mean	EBP	-0.003	0.002	0.003	0.004	0.005	0.023
Mean	Direct	-0.016	-0.003	-0.001	-0.000	0.003	0.013
Median	MQ (SAE)	-0.001	0.003	0.007	0.006	0.008	0.023
Median	EBP	-0.024	-0.014	-0.011	-0.012	-0.010	0.001
Median	Direct	-0.010	-0.003	-0.000	-0.000	0.002	0.012
Quantile_25	MQ (SAE)	0.007	0.013	0.018	0.018	0.021	0.057
Quantile_25	EBP	-0.061	-0.054	-0.048	-0.045	-0.038	0.013
Quantile_25	Direct	-0.038	-0.004	0.004	0.006	0.014	0.078
Gini	MQ (SAE)	-0.005	-0.003	-0.003	-0.002	-0.001	0.006
Gini	EBP	-0.004	0.008	0.012	0.012	0.017	0.020
Gini	Direct	-0.032	-0.016	-0.011	-0.013	-0.008	-0.003
Head_Count	MQ (SAE)	-0.010	-0.007	-0.006	-0.005	-0.004	0.006
Head_Count	EBP	0.019	0.025	0.028	0.028	0.031	0.038
Head_Count	Direct	-0.006	-0.001	0.002	0.002	0.004	0.013
Poverty_Gap	MQ (SAE)	-0.005	-0.004	-0.003	-0.002	-0.001	0.007
Poverty_Gap	EBP	-0.006	-0.000	0.001	0.001	0.002	0.006
Poverty_Gap	Direct	-0.004	-0.001	0.000	0.001	0.002	0.005

Table 11: Summary statistics over the areas for bias (upper half)/rel. bias (lower half) of direct, EBP and MQ (SAE) point estimation results in the contaminated normal errors scenario

method follow a similar behavior over the areas, that is low and high deflections in the same areas, however, the EBP generally shows a higher RMSE.

Overall, the MQ method performs best in terms of efficiency and bias, followed closely by the EBP. The direct estimator again does not provide desirable results at area level, as it is very inefficient. These results can be interpreted such that the EBP is affected by the outlying values, while in the MQ (SAE) approach the tuning constant of Huber loss function limits their influence. Still, both model based approaches successfully borrow strength particularly in the smaller samples.

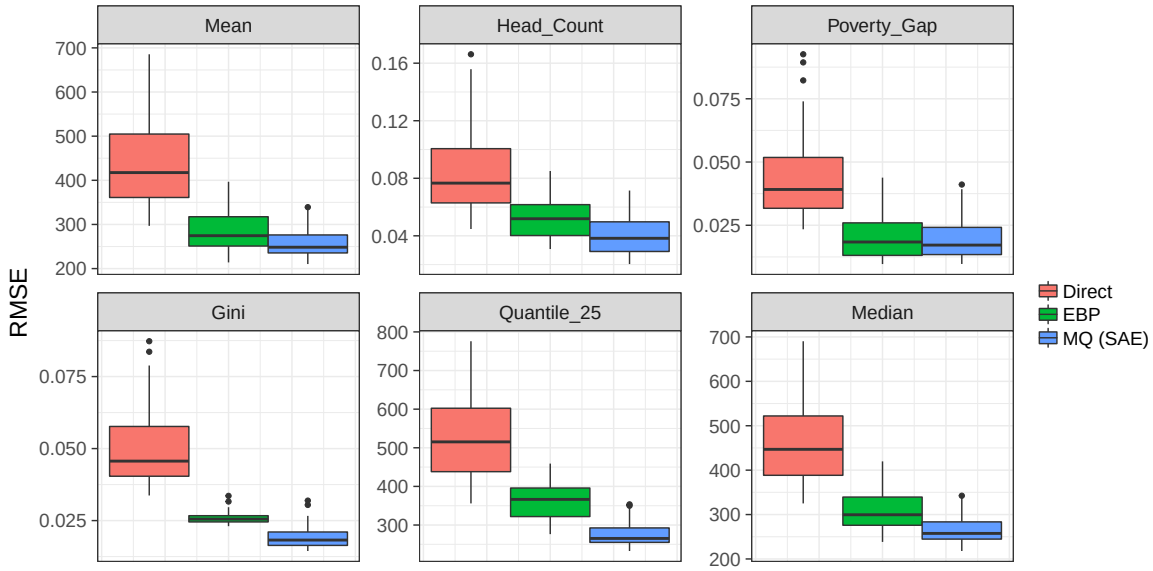

Figure 24: Distribution of RMSE of direct, EBP and MQ (SAE) point estimation results over the areas in the contaminated normal errors scenario

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Mean	MQ (SAE)	210.28	235.46	248.39	256.69	276.23	339.15
Mean	EBP	214.00	251.07	274.52	283.20	317.50	396.68
Mean	Direct	296.80	361.08	417.47	440.25	504.79	685.50
Median	MQ (SAE)	217.90	244.69	257.54	265.48	283.50	342.39
Median	EBP	238.36	275.87	299.53	308.54	339.36	419.71
Median	Direct	325.07	388.38	446.88	467.95	522.05	690.29
Quantile_25	MQ (SAE)	232.72	255.02	265.27	274.44	292.50	353.47
Quantile_25	EBP	276.42	321.88	366.37	363.91	395.82	459.05
Quantile_25	Direct	355.95	438.15	515.29	527.02	602.39	775.85
Gini	MQ (SAE)	0.014	0.016	0.018	0.019	0.021	0.032
Gini	EBP	0.023	0.025	0.026	0.026	0.027	0.034
Gini	Direct	0.034	0.040	0.046	0.051	0.058	0.087
Head_Count	MQ (SAE)	0.020	0.029	0.038	0.040	0.050	0.071
Head_Count	EBP	0.031	0.040	0.052	0.053	0.062	0.085
Head_Count	Direct	0.045	0.063	0.077	0.084	0.101	0.166
Poverty_Gap	MQ (SAE)	0.010	0.013	0.017	0.019	0.024	0.041
Poverty_Gap	EBP	0.010	0.013	0.018	0.020	0.026	0.044
Poverty_Gap	Direct	0.023	0.032	0.039	0.044	0.052	0.093

Table 12: Summary statistics over the areas for the RMSE of direct, EBP and MQ (SAE) point estimation results in the contaminated normal errors scenario

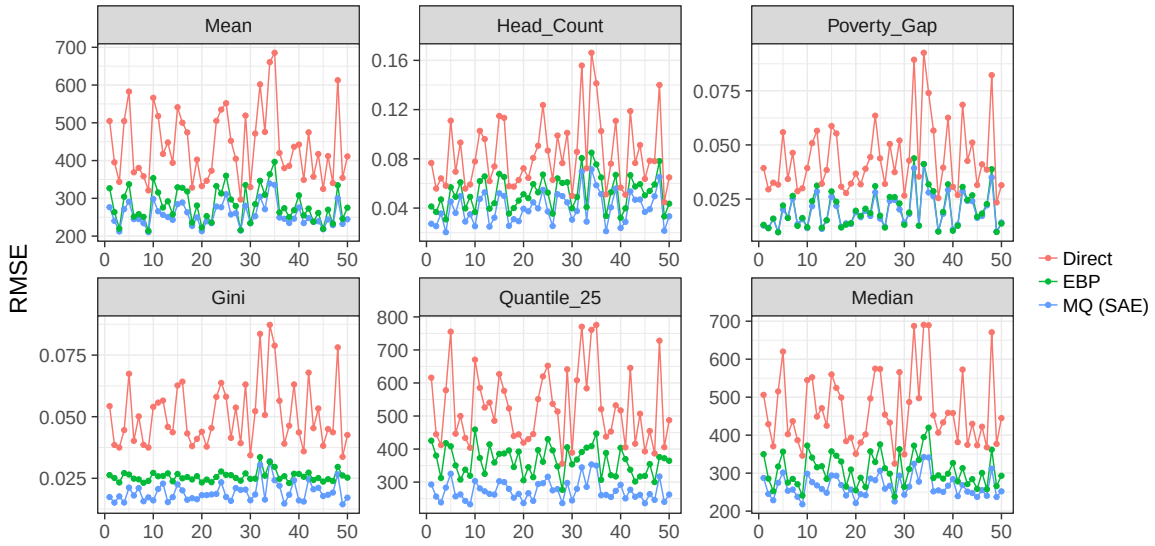

Figure 25: RMSE per area of direct, EBP and MQ (SAE) point estimation results in the contaminated normal errors scenario

5.3 Comparison of EBP and MQ (SAE) Bootstrap MSE Estimation

The MSE estimation is evaluated only for the normal errors and Pareto errors scenario, with an identical simulation setup to the point estimation simulations and also $H = 500$ runs. Bootstrapping is done with $B = 50$ bootstrap populations for the EBP. For the MQ (SAE) approach $B = 50$ populations and $S = 50$ samples are chosen. Because variance estimation for the direct estimator via bootstrapping failed in too many areas, the results are not reported here. The relative bias and the relative RMSE (RRMSE) serve as quality measures. Over H Monte-Carlo simulations, the RRMSE is estimated with

$$RRMSE(\hat{\nu}_j) = \left[\frac{1}{H}\sum_{h=1}^{H}\left(\frac{\hat{\nu}_{j,h} - \nu_j}{\nu_j}\right)^2\right]^{0.5}. \tag{45}$$

To make these results comparable with previous studies regarding the EBP, instead of the MSE the RMSE is used for the calculations of the quality measures.

5.3.1 The Normal Errors Scenario

The box plots displayed in figure 26 visualize the distribution of the relative bias of the MSE estimation. Overall, the EBP estimation is on average relatively unbiased for Gini coefficient (ca. -4.3%), PG (ca. -1.2%) and $Q_{0.25}$ (ca. 4.3%). For mean (8.6%) and median (7.2%) the relative bias is slightly larger. The MQ (SAE) approach is biased for all indicators, with

an absolute relative bias of not less then 16%. Note that the EBP rather overestimates the MSE, while the MQ (SAE) tends toward underestimation, as indicated by the positive and negative bias.

The line plots of the bias/relative bias per area displayed in figure 27 additionally show quite some variation in the areas for all indicators, but only slightly outlying values for both methods.

Regarding the RRMSE, it is however clear that the EBP is superior in this scenario, as the RRMSE is much smaller for all indicators, which is consistent with the large bias of the MQ (SAE) approach.

Hence, the EBP bootstrap shows less bias and more efficiency than the MQ (SAE) bootstrap. This is in line with the expectations, because the EBP assumptions are met. Possibly the MQ approach could perform better with more bootstrap samples. Because the bootstrapping simulations are computationally very expensive, this could not be investigated further.

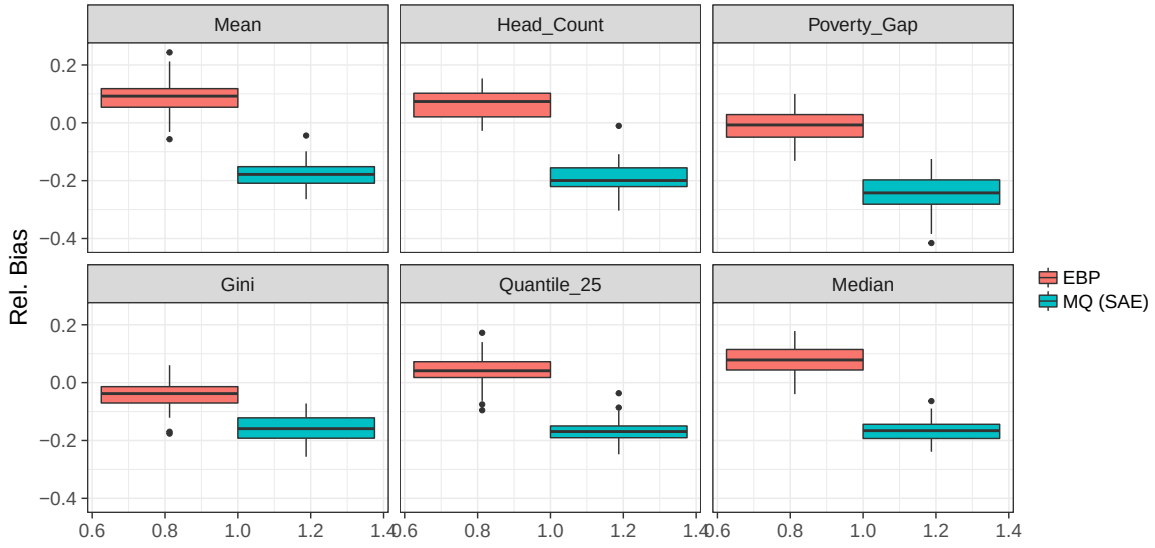

Figure 26: Distribution of rel. bias of EBP and MQ (SAE) RMSE estimation results over the areas in the normal errors scenario

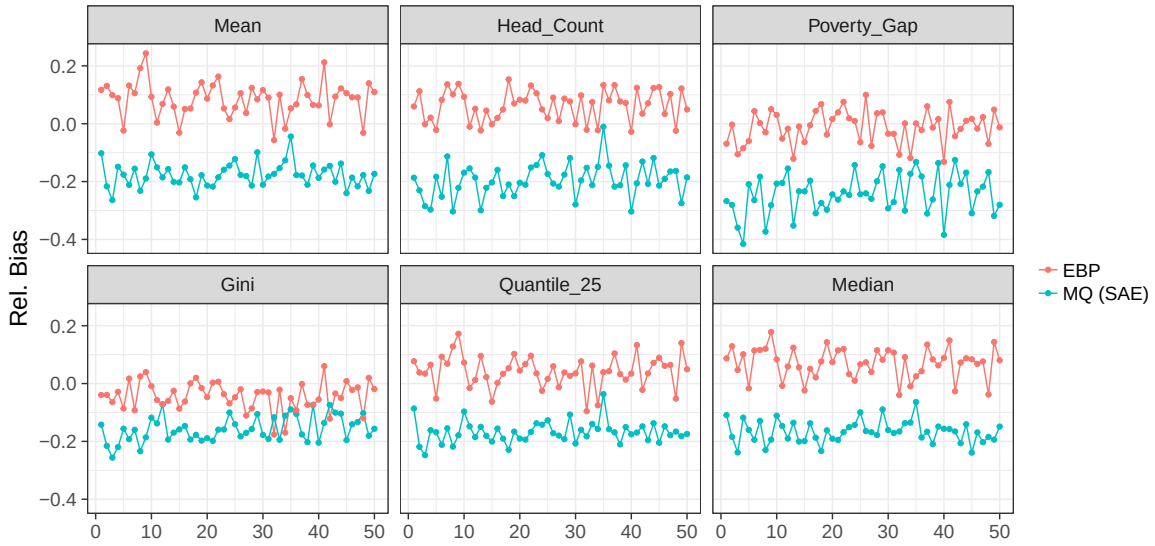

Figure 27: Rel. bias of EBP and MQ (SAE) RMSE estimation results per area in the normal errors scenario

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Gini	MQ (SAE)	-0.256	-0.192	-0.159	-0.157	-0.122	-0.072
Gini	EBP	-0.176	-0.070	-0.038	-0.043	-0.014	0.060
Head_Count	MQ (SAE)	-0.304	-0.220	-0.200	-0.195	-0.156	-0.010
Head_Count	EBP	-0.028	0.021	0.074	0.063	0.102	0.153
Mean	MQ (SAE)	-0.264	-0.209	-0.178	-0.177	-0.152	-0.044
Mean	EBP	-0.057	0.054	0.092	0.086	0.118	0.243
Median	MQ (SAE)	-0.239	-0.193	-0.166	-0.166	-0.144	-0.064
Median	EBP	-0.040	0.044	0.078	0.072	0.115	0.178
Poverty_Gap	MQ (SAE)	-0.416	-0.281	-0.242	-0.243	-0.197	-0.125
Poverty_Gap	EBP	-0.132	-0.050	-0.007	-0.012	0.028	0.100
Quantile_25	MQ (SAE)	-0.248	-0.190	-0.169	-0.168	-0.150	-0.037
Quantile_25	EBP	-0.095	0.018	0.041	0.043	0.072	0.172

Table 13: Summary statistics over the areas for the relative bias of the RMSE estimation in the normal errors scenario

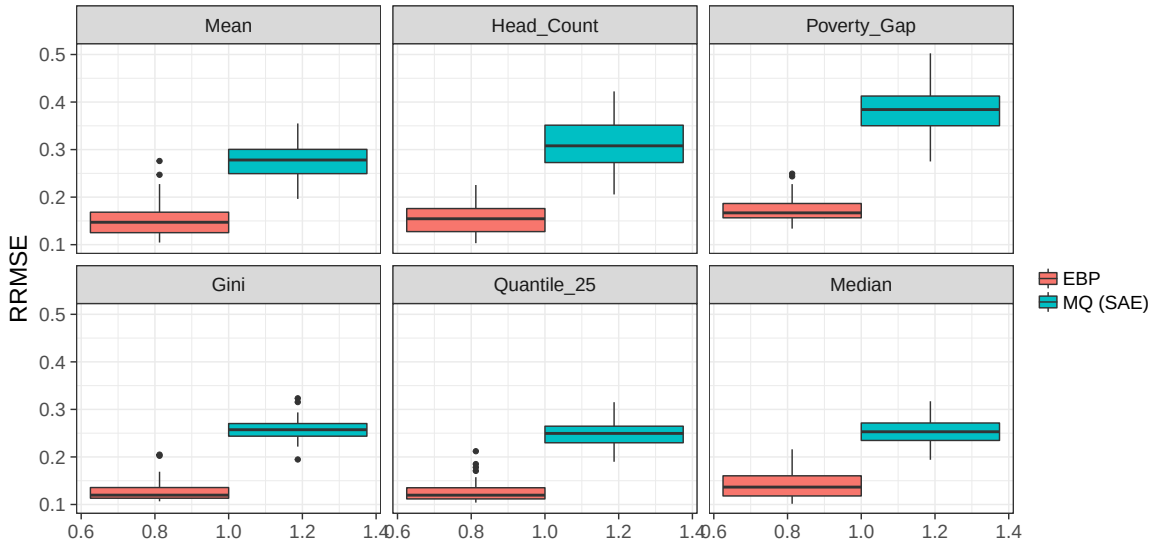

Figure 28: Distribution of RRMSE of EBP and MQ (SAE) RMSE estimation results over the areas in the normal errors scenario

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Gini	MQ (SAE)	0.195	0.244	0.257	0.257	0.270	0.323
Gini	EBP	0.107	0.113	0.120	0.127	0.136	0.205
Head_Count	MQ (SAE)	0.206	0.273	0.308	0.309	0.351	0.422
Head_Count	EBP	0.103	0.127	0.154	0.157	0.176	0.225
Mean	MQ (SAE)	0.196	0.249	0.278	0.275	0.300	0.355
Mean	EBP	0.104	0.125	0.147	0.151	0.168	0.276
Median	MQ (SAE)	0.194	0.235	0.253	0.253	0.271	0.317
Median	EBP	0.102	0.118	0.137	0.140	0.160	0.216
Poverty_Gap	MQ (SAE)	0.275	0.350	0.384	0.382	0.413	0.502
Poverty_Gap	EBP	0.134	0.156	0.167	0.174	0.187	0.249
Quantile_25	MQ (SAE)	0.190	0.230	0.250	0.247	0.265	0.315
Quantile_25	EBP	0.104	0.112	0.120	0.128	0.135	0.212

Table 14: Summary statistics over the areas for the RRMSE of the RMSE estimation results in the normal errors scenario

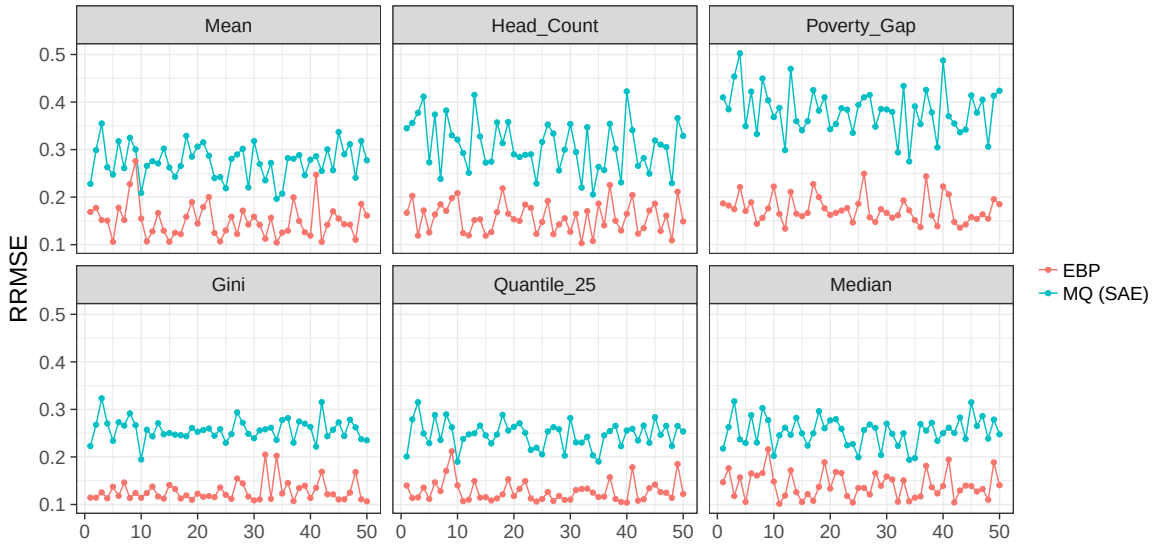

Figure 29: RRMSE of EBP and MQ (SAE) RMSE estimation results per area in the normal errors scenario

5.3.2 The Pareto Errors Scenario

Note that for the MSE estimation of the Pareto errors only 400 simulations are run instead of 500, because some convergence issues occurred for the EBP.

The box plots displayed in figure 30 show the distribution of the relative bias of the RMSE estimation over the areas. Overall, the estimation is on average relatively unbiased (5.7% absolute relative bias or less) for the EBP for mean, HCR and median. For PG and $Q_{0.25}$ the absolute relative bias is larger with circa 22% and 33%, and very large for Gini coefficient with more than 50%. The MQ (SAE) approach is biased for all indicators, but does not show such extreme values, since the absolute relative bias is for no indicator on average larger than 16%. Note that both estimators tend to underestimate the RMSE in this scenario.

The line plots of the bias/relative bias per area displayed in figure 31 additionally show that apart from PG, the differences in bias over the areas are moderate.

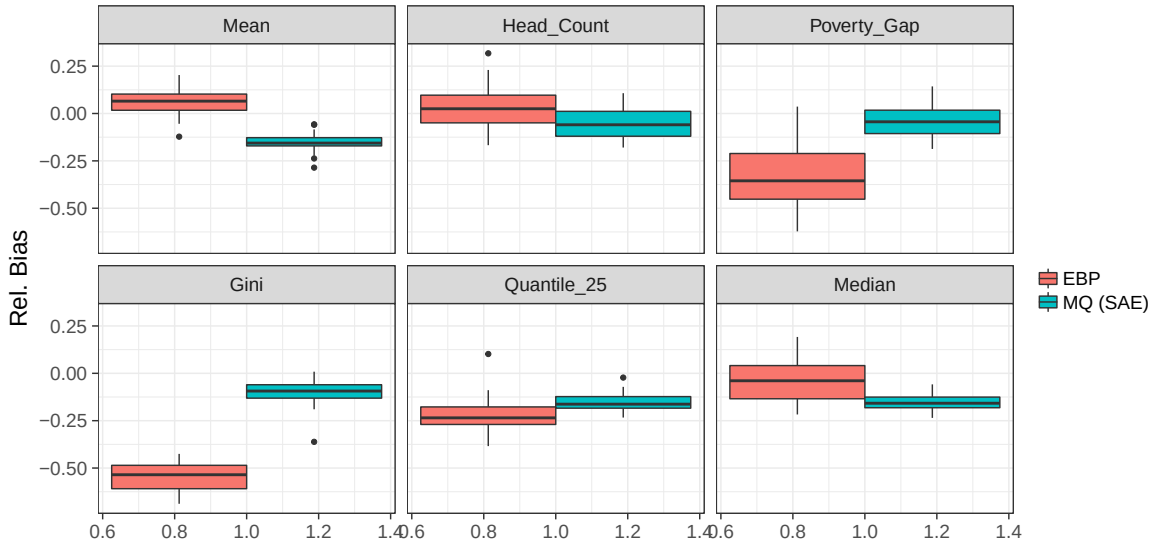

Figure 30: Distribution of rel. bias of EBP and MQ (SAE) RMSE estimation results over the areas in the Pareto errors scenario

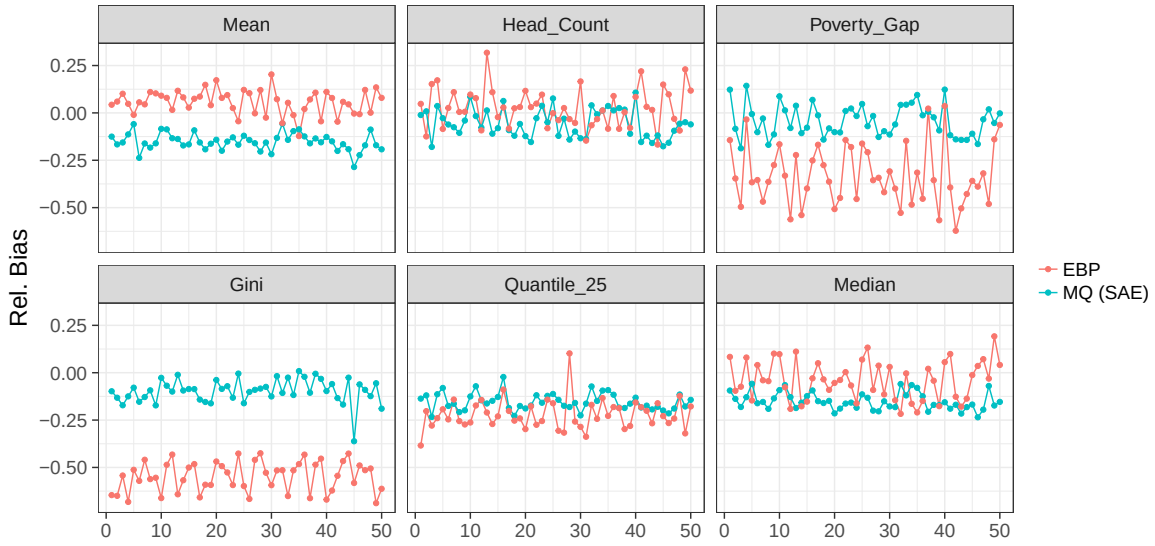

Figure 31: Rel. bias of EBP and MQ (SAE) RMSE estimation results per area in the Pareto errors scenario

For the RRMSE, the results are also mixed. Consistent with those indicators, where the EBP RMSE estimation is biased, it also shows a higher RRMSE for PG, Gini coefficient and $Q_{0.25}$ than the MQ (SAE) approach, while the opposite is true for the remaining indicators.

Regarding the line plots depicted in 33 the EBP shows considerable differences in the RMSE estimation for PG and Gini coefficient.

Because of the violations of the normality assumptions, the results are somewhat in line

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Gini	MQ (SAE)	-0.362	-0.131	-0.094	-0.097	-0.060	0.009
Gini	EBP	-0.689	-0.609	-0.535	-0.548	-0.486	-0.425
Head_Count	MQ (SAE)	-0.180	-0.120	-0.059	-0.055	0.011	0.107
Head_Count	EBP	-0.167	-0.049	0.025	0.027	0.097	0.318
Mean	MQ (SAE)	-0.286	-0.171	-0.156	-0.152	-0.128	-0.057
Mean	EBP	-0.122	0.017	0.065	0.057	0.103	0.203
Median	MQ (SAE)	-0.235	-0.182	-0.158	-0.150	-0.126	-0.058
Median	EBP	-0.217	-0.134	-0.039	-0.038	0.041	0.192
Poverty_Gap	MQ (SAE)	-0.187	-0.106	-0.044	-0.040	0.018	0.143
Poverty_Gap	EBP	-0.622	-0.452	-0.355	-0.331	-0.211	0.036
Quantile_25	MQ (SAE)	-0.233	-0.184	-0.163	-0.154	-0.123	-0.022
Quantile_25	EBP	-0.384	-0.270	-0.235	-0.221	-0.177	0.102

Table 15: Summary statistics over the areas for the relative bias of the EBP and MQ (SAE) RMSE estimation in the Pareto errors scenario

with the expectations, since the MSE estimation of the EBP is overall more biased and less efficient compared to the normal errors scenario. It is however surprising, that there is that much variation in the results depending on the indicators. Particularly in the case of Gini coefficient, the RMSE estimation gets very imprecise for both methods and is severely biased for the EBP.

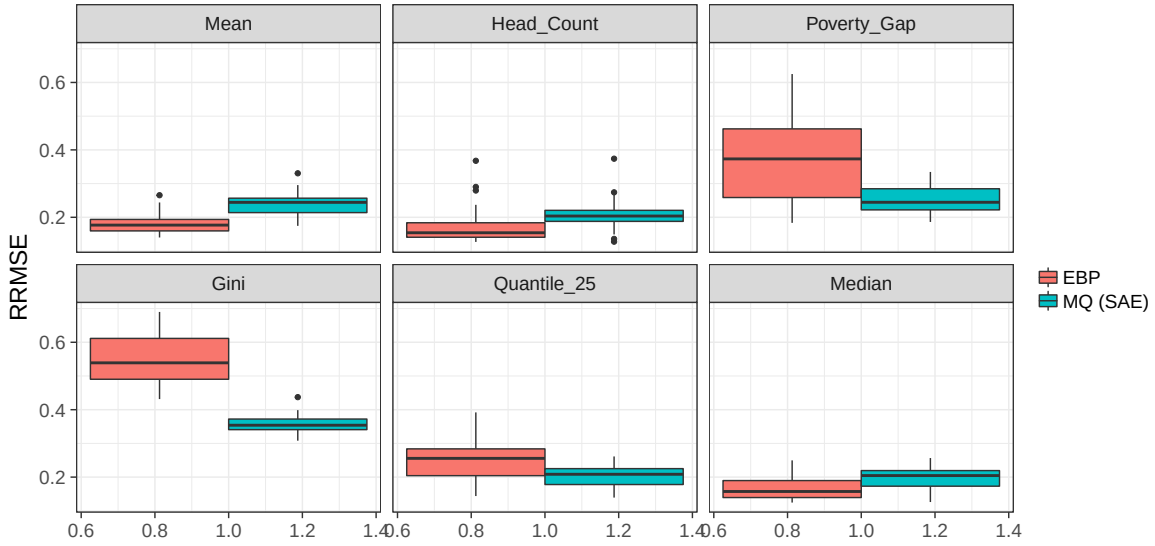

Figure 32: Distribution of RRMSE of EBP and MQ (SAE) RMSE estimation results over the areas in the Pareto errors scenario

Indicator	Estimator	Min	Q.25	Median	Mean	Q.75	Max
Gini	MQ (SAE)	0.308	0.341	0.354	0.356	0.372	0.437
Gini	EBP	0.432	0.490	0.539	0.552	0.611	0.690
Head_Count	MQ (SAE)	0.128	0.188	0.204	0.208	0.221	0.374
Head_Count	EBP	0.127	0.141	0.154	0.169	0.184	0.368
Mean	MQ (SAE)	0.175	0.214	0.244	0.238	0.257	0.331
Mean	EBP	0.140	0.160	0.177	0.180	0.194	0.266
Median	MQ (SAE)	0.126	0.173	0.205	0.198	0.219	0.257
Median	EBP	0.125	0.139	0.157	0.168	0.190	0.250
Poverty_Gap	MQ (SAE)	0.186	0.222	0.245	0.253	0.285	0.335
Poverty_Gap	EBP	0.184	0.259	0.373	0.368	0.462	0.625
Quantile_25	MQ (SAE)	0.139	0.178	0.208	0.203	0.225	0.261
Quantile_25	EBP	0.144	0.204	0.256	0.249	0.284	0.392

Table 16: Summary statistics over the areas for the RRMSE of the EBP and MQ (SAE) RMSE estimation results in the Pareto errors scenario

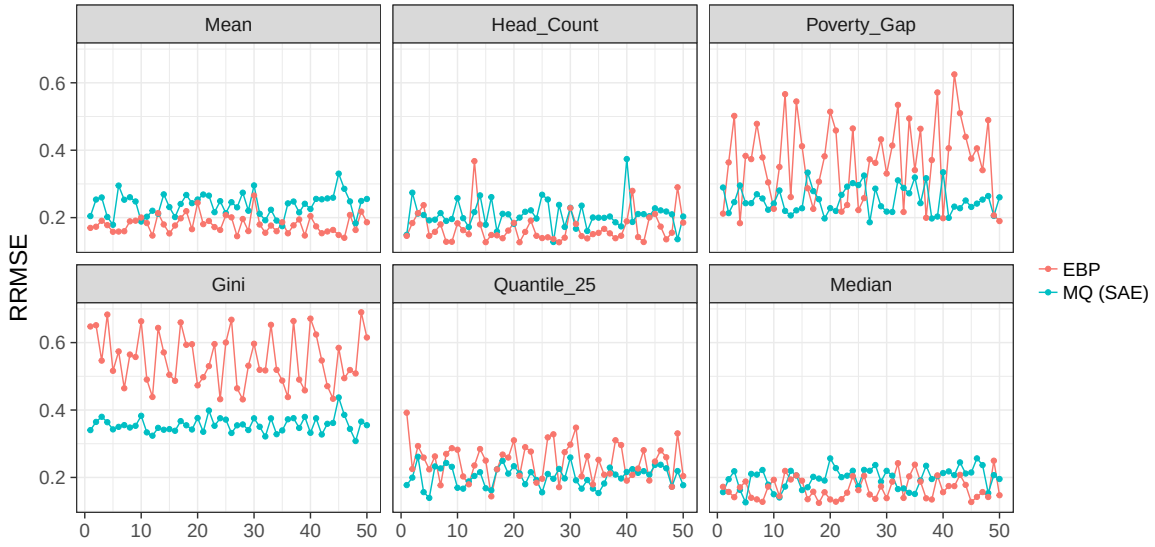

Figure 33: RRMSE of EBP and MQ (SAE) RMSE estimation results per area in the Pareto errors scenario

5.4 Summary of Simulation Results

The simulations for the point estimation show differences regarding bias and efficiency for the three considered estimators. Generally, the model based estimation provides much better results than direct estimation. Between EBP and MQ (SAE) there are also differences: the EBP is more efficient and less biased than the MQ (SAE) approach when the individual error terms are normally distributed (normal errors scenario) or a normal distribution of the individual errors can be achieved with a Box Cox transformation (log-scale outcomes scenario). In the log-scale scenario the MQ (SAE) approach performs even worse than direct

estimation. Likely, the non-linear relationship between dependent and independent variable hinders the MQ (SAE) to effectively borrow strength.

In contrast, the MQ (SAE) approach is more efficient and less biased than EBP estimation for long-tailed error distributions, since the results in the Pareto and contaminated normal errors scenario are quite similar in this manner. In the presence of these distributions, the EBP is either affected by outlying values or the inability to transform the distribution to meet the Gaussian assumptions.

It should also be highlighted, that the estimation of the considered indicators is not of equal quality. Particularly for Gini coefficient,$Q_{0.25}$ and sometimes the FGT poverty measures a larger bias and more variation compared to median and mean is found.

In the MSE estimation simulations, the results indicate some bias for both model based methods for most indicators. Nonetheless, in the normal scenario the results of the EBP are still trustworthy, while the MQ (SAE) approach shows about twice as much absolute relative bias. Further, MQ (SAE) tends here to underestimation of the MSE. For the Pareto scenario, the results are mixed, and particularly for Gini coefficient both methods do not produce reliable estimates for the MSE.

A clearer picture regarding the performance of both bootstrap MSE estimators might be obtainable, when larger values for bootstrap populations and bootstrap samples are chosen. However, despite parallelization in the current setting the computational time for one MSE scenario on a computational server with 24 processing cores is around five to six days, and increasing these parameters contributes to the computational time in a non-linear way. Hence, improving this aspect has to be left for future research, possibly after translating some of the new functions to `c++` as suggested in section 4.2.3.

6 Application: Estimation of Poverty and Inequality in Austria

In the final part of this thesis the most important capabilities of the `mquantreg` package are demonstrated in the context of *poverty mapping*. Firstly, the concept of poverty mapping is outlined. Afterwards an applied on Austrian income data based on the M-quantile small area approach is presented. What follows, can serve together with the actual R-output displayed in the appendix as a basis for a vignette for the `mquantreg` package.

6.1 The Poverty Mapping Framework

Poverty mapping is an approach introduced by Henninger and Snel (2002). The aim of poverty mapping is to obtain high resolution maps of poverty for smaller regional areas of a country. These maps help to understand the spatial distribution of poverty, and in combination with small area methods that "borrow strength" they can help to "uncover poor areas that might otherwise go undetected" (Henninger and Snel, 2002, p.1). Because poverty mapping provides an intuitive and easily interpretable way of presenting the findings of poverty estimation, it can be of especial importance when research is used in the context of policy and decision making.

A detailed description of the poverty mapping approach is presented in Henninger and Snel (2002). Generally, the approach can be divided in eight[6] steps which include defining the poverty concept and its measurement, model building and presentation of results. For the economy of space the different steps are presented directly together with the application on the EU-SILC (European Union Statistics on Income and Living Conditions) data in the now following application.

6.2 Poverty Mapping for Austria with the MQ (SAE) Approach

Step 1: Define purpose and expected use of mapping: The aim of this application is to present the spatial distribution of inequality and poverty in Austria on district level.

Step 2: Select measure(s) of poverty and human well-being: To measure poverty and inequality, only income-based measures are used. Results are presented for the head count ratio, the poverty gap (see equation (25)) and Gini coefficient (see equation (26)). For the first and second indicator, the poverty line is set to 60% of the median income.

[6] Step 8 (monitoring of use and feedback) is not relevant in this application.

Step 3: Select input data: The data set is the Austrian EU-SILC as provided by the `emdi` package. This data is a synthetic dataset which is generated based on the real Austrian EU-SILC.

Data is available on population level (`eusilcA_pop`) and sample level (`eusilcA_smp`) for 17 variables including three regional variables for the states, districts and counties. From these variables, the equalized household income (eqIncome) is used as the dependent variable, and it is assumed to be measured only in the sample.

For presentation purposes, only five explanatory variables are used: the respondents gender (gender), net employee cash or near cash income (cash), net cash benefits or losses from self-employment (self_empl), net unemployment cash benefits (unempl_empl) and the equivalized household size according to the modified OECD scale (eqsize). Gender is a factor variable and the other variables are numeric. Descriptive statistics for the variables are displayed in table 17.

	Min	Q.25	Median	Mean	Q.75	Max
eqIncome	313.04	13584.63	18315.47	19864.73	24501.74	89555.46
eqsize	1.00	1.00	1.50	1.61	2.00	4.50
cash	0.00	0.00	12186.01	12546.22	21674.42	102531.37
self_empl	0.00	0.00	0.00	1884.26	0.00	68278.18
unempl_ben	0.00	0.00	0.00	429.01	0.00	21937.70
	Count	Proportion				
male	606	60.6 %				
female	394	39.4 %				

Table 17: Summary statistics for selected variables of the eusilcA_smp dataset. N=1000.

Step 4: Select method of estimating or calculating poverty indicator: The measurement of poverty is based on a single variable, which is the equivalized household income. Hence, only income type concepts of poverty are considered.

Step 5: Select a method to calculate, estimate, or display poverty indicator for geographic area: The method to estimate the respective indicators is the MQ (SAE) approach, as presented in this thesis. To be able to apply the MQ (SAE) method a model has to be build that links the available auxiliary data to the target variable, usually based on theoretical assumptions or previous research results. For the here selected independent variables, a first set of M-quantile regressions can be run to investigate, if there is a sufficient relationship between the dependent and independent variables (as defined in the previous step).

M-quantile	0.25	0.5	0.75
(Intercept)	13870.92	17272.19	21453.23
genderfemale	-507.56	-437.60	-669.10
eqsize	-2639.21	-2657.00	-2486.04
cash	0.45	0.42	0.38
self_empl	0.49	0.48	0.47
unempl_ben	-0.03	-0.05	-0.09
Pseudo R^2	0.16	0.31	0.23
Iterations	7	6	7

Table 18: Results of M-quantile regression for $\tau \in \{0.25, 0.5, 0.75\}$ in the sample data. N=1000. Based on the `summary.mq` function

Table 18 shows the results of three M-quantile regressions that are estimated with the `mq` function. As an approximate of the explanatory strength of the model in the data, the Pseudo-R^2 can be considered.[7] In this case, up to 31% of the variance can be explained by the $\tau = 0.5$ M-quantile regression. This indicates that at least in the sample there is a sufficient relationship between the variables. The M-quantile regression results further show, that the independent variables have effects pointing in the same direction on the different conditional M-quantiles. In particular, being female, having a large family and the reception of unemployment benefit have a negative impact on the equivalized household income in the sampled data for all estimated M-quantile regressions

As explained in section 3.3.2, the MQ (SAE) approach is based on pseudo random effects. It will however be only sensible to use this approach, if there is indeed variation of intercepts and/or slopes in between the areas. To check this, the `mmqm` function can serve as a diagnostic tool. Note that the `mmqm` model is always stored when the `mq_sae` function is run, such that it is also possible to run these diagnostics at a later point.

Figure 34 shows a plot of the pseudo random effects per variable based on the `mmqm.plot` function. In each plot, there is one regression line per area. Indeed, there are considerable differences in the intercept between the areas, and especially for the variable "cash" different slopes can be found.

[7]The Pseudo-R^2 is calculated as $1 - SSR/SST$. Because the residuals of the M-quantile regression do not generally sum up to 1, it can also produce negative values, especially for M-quantiles distant from 0.5. Therefore, it should only be used for M-quantile regressions near $\tau = 0.5$ and regarded as an approximate figure.

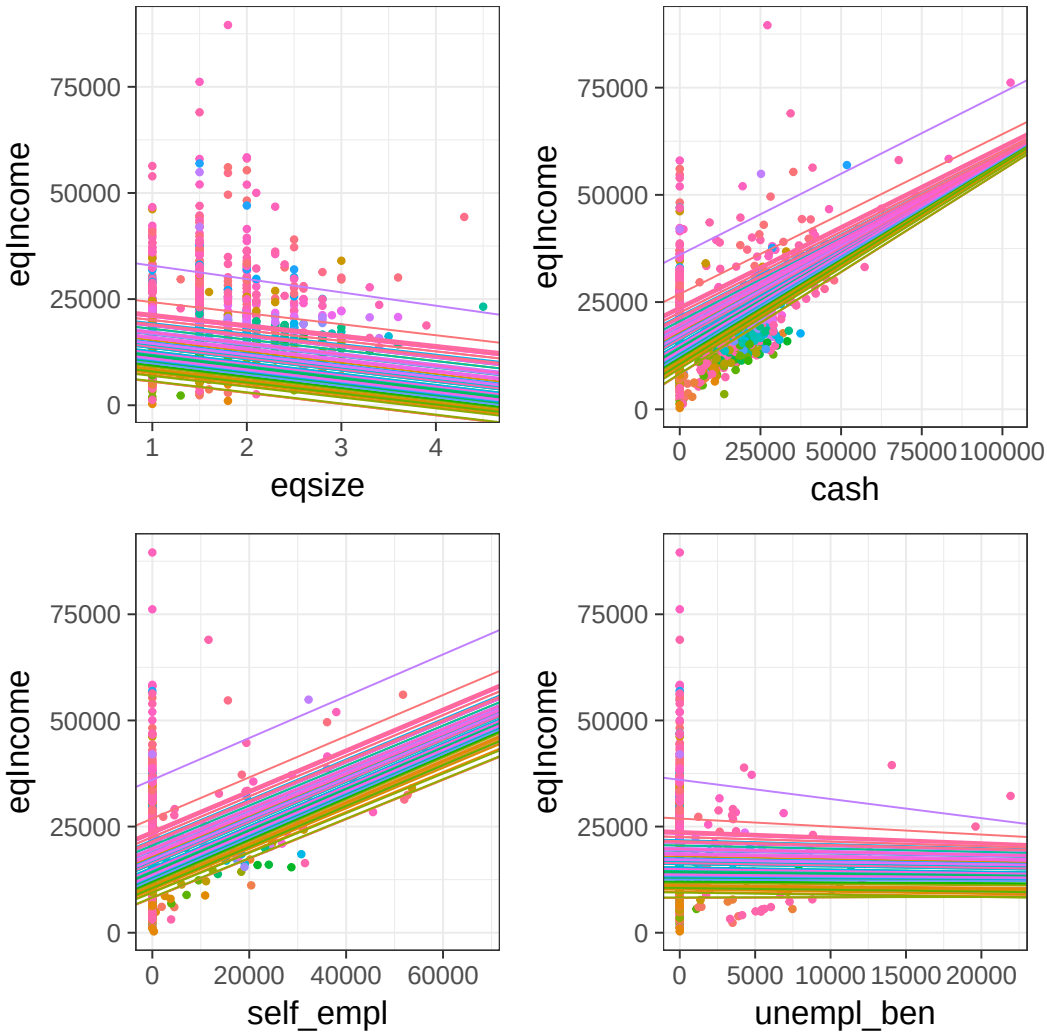

Figure 34: Pseudo random effects per independent variable based on the `mmqm.plot` function. One line per district

One can also check whether there are differences in the resulting $\bar{\tau}_j$ values per area. In table 6.2 the values for the first 20 districts are shown, and there are evidently positive ($\bar{\tau}_j > 0.5$) and negative ($\bar{\tau}_j < 0.5$) pseudo random effects.

Based on the now available knowledge from the Monte-Carlo simulations from section 5.2 the MQ (SAE) model produces good results for normal and longtailed near-normal error distributions. Figure 35 indicates, that at least the dependent variable shows such a distribution. If this is taken as a first guess on the error distribution, the MQ (SAE) approach can make sense.

Domain	$\bar{\tau}$
Amstetten	0.01
Baden	0.05
Bludenz	0.01
Bodensee	0.29
Braunau am Inn	0.07
Bregenz	0.09
Bruck-Mrzzuschlag	0.03
Bruck an der Leitha	0.08
Deutschlandsberg	0.05
Dornbirn	0.54
Eferding	0.25
Eisenstadt	0.18
Feldkirch	0.71
Feldkirchen	0.06
Freistadt	0.19
Gnserndorf	0.20
Gmnd	0.22
Gmunden	0.18
Graz	0.11
Graz Umgebung	0.11

Table 19: Average M-quantile per district in the eusilcA_smp dataset (20 districts are shown)

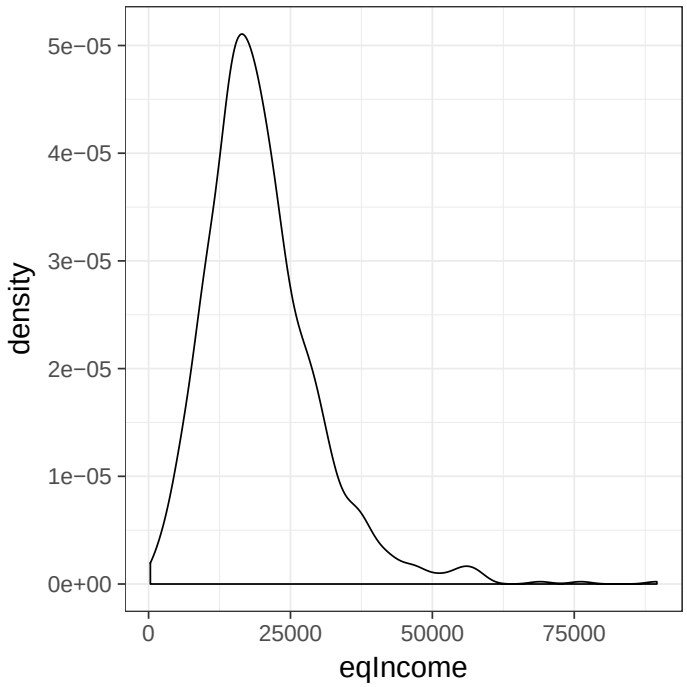

Figure 35: Density plot of the dependent variable eqIncome

Overall, the data appears to be suitable for the MQ (SAE) approach. Hence, the `mq_sae`

function is used to calculate the model. The model is run using the standard settings for point estimation (i.e. $L = 50$ approximations). The MSE calculation is based on $B = 30$ bootstrap populations and $S = 30$ bootstrap samples. The poverty maps can then be created with the `map_plot` function from the `emdi` package (here a modified version of this function is used to obtain a different color scale). As a comparison, the results for the EBP with Box Cox transformation and $L = 50$ approximations and $B = 30$ bootstrap populations are also reported.

Step 6: Decide on number of units for final map (resolution) to present poverty data: This decision on the resolution should be based on the purpose of the map, but one should also take measures of uncertainty into account. Therefore, maps showing the MSE are also created. The results are presented on the district level, which means 96 Austrian districts.

Step 7: Produce and distribute maps: The resulting maps now show point estimates in figure (36) as well as the MSE as a measure of their precision in figure (37).

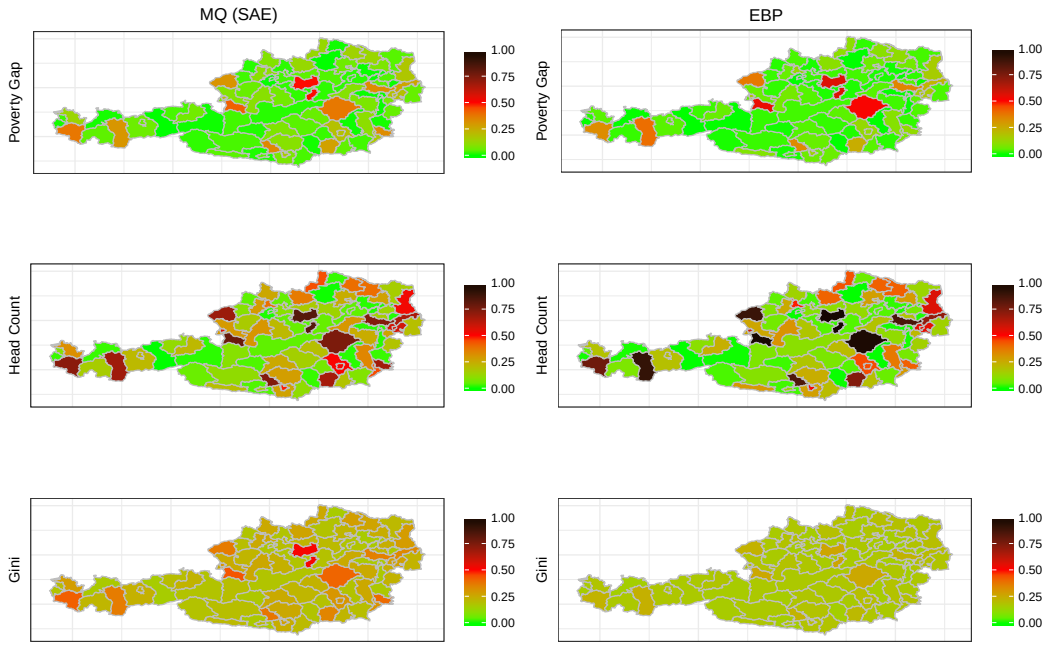

Figure 36: Point estimation results for poverty mapping in Austria based on the `mq_sae` and `ebp` functions. Note that these maps are created with a modified version of the `emdi::map_plot` function to allow for a different color scale.

The point estimation results show indeed an uneven spatial distribution of poverty. In

general, the poverty estimation is relatively consistent regarding both indicators. Those districts, that have a high HCR value, also have a rather high PG and vice versa. To some extent this also applies to the relation of inequality and poverty. Particularly the MQ (SAE) approach estimates high values for the Gini in districts, that also have a high level of poverty.

Concerning the comparison of both methods, the results are pretty similar. Most importantly, the same districts are identified by both methods as those with high or low levels of poverty and inequality. A slight difference can be found regarding the magnitude, because the EBP estimates somewhat higher values of poverty, while the MQ (SAE) approach estimates slightly higher values for the Gini coefficient. However, the differences are relatively small.

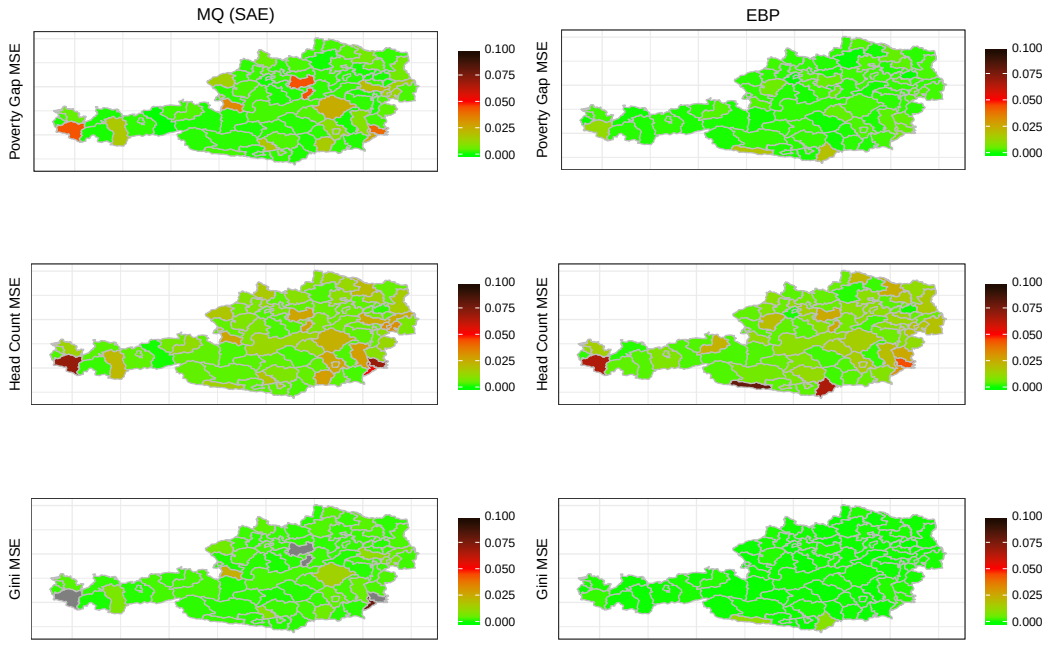

Figure 37: MSE estimation results for poverty mapping in Austria based on the `mq_sae` and `ebp` functions. Note that these maps are created with a modified version of the `emdi::map_plot` function to allow for a different color scale.

The MSE estimation results generally indicate a higher uncertainty for the MQ (SAE) estimates. The MQ (SAE) approach shows consistent results over the areas, while the MSE is higher for the HCR. The EBP also finds a higher MSE for the HCR than for the other indicators. For the HCR, both methods also show relative similar results over the areas. However, for Gini and PG, the EBP estimates much lower levels of uncertainty. For the MQ (SAE) approach three areas (marked as grey) are even excluded, because a MSE of more than one is found. Because these are areas only have a sample size of one and four, it is however relative unintuitive that the EBP estimates indicators for these areas with very

high precision. Nonetheless, because the number of bootstrap samples and populations is set rather low, the results should not be overinterpreted, but rather serve as an example for visualizing MSE estimation results in the course of poverty mapping.

In practice, the important question is now, which results are closer to the true poverty and inequality indicators. This will depend mostly on the question, if the assumptions of the EBP are fulfilled, in which case it should be given the priority. The QQ-plot shown in figure 38 in the appendix indicates however, that the distributional assumptions are violated for the individual errors. Hence, it could be better to choose the more robust results of the MQ-approach. On the other hand, it should be highlighted that both methods produce in relative terms similar results in the point estimation. Therefore, the question in which districts poverty is higher or lower would be answered similarly by both methods, which might be of importance in applied work.

7 Summary and Conclusions

The primary aim of this thesis was to provide the means for a wider scientific audience to be able to use the M-quantile small area approach for the estimation of non-linear indicators.

With the `mquantreg` package that includes not only functions for M-quantile based estimation of regressions, pseudo random effects, linear and non-linear indicators in small areas as well as their mean squared error but also allows to summarize, print and visualize the results, important steps to user friendly implementation according to R standards are now taken.

The second aim was to compare the performance of the MQ (SAE) approach to the EBP based on different Monte-Carlo simulation studies. The results show that both methods have their ideal habitats: the EBP is superior when its Gaussian assumptions are met by the error distributions, or when these assumptions can be established by an appropriate transformation of the target variable. The MQ (SAE) approach is more efficient and less biased in the here considered long-tailed error distributions, i.e. the Pareto distribution and in the presence of outliers. These results are relatively consistent for the different linear and non-linear indicators that are considered.

These simulation results are mostly in line with theoretical expectations, since both methods have certain strengths resulting from the different underlying approach: The EBP aims to establish a normal distribution based on transformations, while the MQ (SAE) limits the influence of data points based on the Huber proposal 2 loss function. Overall, it is therefore probably best to consider MQ (SAE) and EBP as complementing methods. Particularly in income-related estimation outliers in the data are common, but also log-normal relationships between the dependent and independent variable, which can make either method advantageous depending on the circumstances.

Limitations of the conducted research include that the MSE estimation is only tested on two error distributions. It would also be worth to investigate if possibly larger values for the number of bootstrap samples could increase the bootstrapping performance, especially for the MQ (SAE). Regarding the implementation in R, the focus in this thesis laid on the MQ (SAE) function. For the additionally implemented functions, many extensions are possible, for instance, the implementation of variance estimation for the M-quantile regression function. Some mostly performance related suggestions for the improvement of the functions in general are already mentioned in section 4.2.3 and not repeated here.

Further possible research directions for the M-quantile approach in general concern the

specification of the grid, on which M-quantile pseudo random effects are calculated, as well procedures to determine the optimal tuning parameter k in the Huber loss function. In the small area context, the role of weights in the MQ (SAE) approach can be further investigated. Since its underlying model is a weighted regression, possibly informative sampling designs could be accounted for. This would be an advantage to the EBP, which does not support weights. Also, the implementation of at least the log-transformation in the MQ approach could be important to linearize the relationship between the dependent and independent variables. That is, while the error distribution does not need to be of a certain type, the explanatory power of the underlying M-quantile model that "borrows strength" can be low, when the true relationship between the independent and dependent variable is not linear. This is indicated by the simulation results of the log-scale scenario. Regarding the comparision of MQ (SAE) and EBP, so far only scenarios with normally distributed area effects are simulated. It can however be expected that the MQ approach also shows a good performance in the presence of non-normal area effects. Finally, a comprehensive comparison to the ELL method is still pending as already noticed by Tzavidis et al. (2008).

The application in section 6 shows, that the implemented functions can already be used for poverty mapping in synergy with the `emdi` package. Naturally, the use of the implemented small area approach is not limited to poverty estimation, but it can also be used for robust estimation of linear and non-linear small area indicators in other research areas like the analysis of rental, agricultural or business data. Outside of small area estimation the ability to conduct robust M-quantile regression can also be interesting, wherever the aim does not lie in the testing of hypothesis, but rather in the robust prediction of non-sampled values. An instance where this is the case is machine learning.

References

Battese, G. E., R. M. Harter, and W. A. Fuller (1988): "An Error-Components Model for Prediction of County Crop Areas Using Survey and Satellite Data," *Journal of the American Statistical Association*, 83, 28.

Betti, G. and A. Lemmi (2013): *Measuring multidimensional deprivation with dichotomized and ordinal variables*, Poverty and Social Exclusion: New Methods of Analysis, Routledge: London.

Bianchi, A., E. Fabrizi, U. C. d. S. Cuore, N. Salvati, and N. Tzavidis (2015): *Estimation and Testing in M-quantile Regression with application to small area estimation*, Working paper.

Bianchi, A. and N. Salvati (2015): "Asymptotic properties and variance estimators of the M-quantile regression coefficients estimators," *Communications in Statistics - Theory and Methods*, 44, 2416–2429.

Breckling, J. and R. Chambers (1988): "M-quantiles," *Biometrika*, 75, 761–771.

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

Chambers, R. L. and R. Dunstan (1986): "Estimating distribution functions from survey data," *Biometrika*, 73, 597–604.

Dowle, M. and A. Srinivasan (2017): *data.table: Extension of 'data.frame'*, R package version 1.10.4.

Elbers, C., J. O. Lanjouw, and P. Lanjouw (2003): "Micro-Level Estimation of Poverty and Inequality," *Econometrica*, 71, 355–364.

Eurostat (2013): *Guide to Statistics in European Commission Development Co-operation*, Luxembourg: European Union.

Fay, R. E. and R. A. Herriot (1979): "Estimates of Income for Small Places: An Application of James-Stein Procedures to Census Data," *Journal of the American Statistical Association*, 74, 269–277.

Foster, J., J. Greer, and E. Thorbecke (1984): "A class of decomposable poverty measures," *Econometrica*, 52, 761–766.

Fox, J. and H. S. Weisberg (2010): *An R Companion to Applied Regression*, Thousand Oaks, California: SAGE, 2 ed.

Gastwirth, J. L. (1972): "The Estimation of the Lorenz Curve and Gini Index," *The Review of Economics and Statistics*, 54, 306.

Gonzalez-Manteiga, W., M. J. Lombardia, I. Molina, D. Morales, and L. Santamaria (2008): "Bootstrap mean squared error of a small-area EBLUP," *Journal of Statistical Computation and Simulation*, 78, 443–462.

Haslett, S. J. (2016): "Small Area Estimation Using Both Survey and Census Unit Record Data: Links, Alternatives, and the Central Roles of Regression and Contextual Variables," in *Analysis of Poverty Data by Small Area Estimation*, West Sussex, United Kingdom: John Wiley and Sons Ltd, 327–348.

Henninger, N. and M. Snel (2002): *Where are the poor? Experiences with the development and use of poverty maps*, Washington, D.C.: World Resources Institute.

Holland, P. W. and R. E. Welsch (1977): "Robust regression using iteratively reweighted least-squares," *Communications in Statistics - Theory and Methods*, 6, 813–827.

Horvitz, D. G. and D. J. Thompson (1952): "A Generalization of Sampling Without Replacement From a Finite Universe," *Journal of the American Statistical Association*, 47, 663–685.

Huber, P. J. (1964): "Robust Estimation of a Location Parameter," *The Annals of Mathematical Statistics*, 35, 73–101.

Koenker, R. (2000): *Quantile regression*, Working paper.

——— (2017): *quantreg: Quantile Regression*, R package version 5.33.

Koenker, R. and G. Bassett (1978): "Regression Quantiles," *Econometrica*, 46, 33.

Koenker, R. and K. Hallock (2001): "Quantile Regression," *Journal of Economic Perspectives*, 15, 143–156.

Kokic, P., R. Chambers, J. Breckling, and S. Beare (1997): "A Measure of Production Performance," *Journal of Business & Economic Statistics*, 15, 445.

Kreutzmann, A.-K., S. Pannier, N. Rojas-Perilla, T. Schmid, M. Templ, and N. Tzavidis (2017): *The R package emdi for estimating and mapping regionally disaggregated indicators*, Working paper.

Lombardia, M. J., W. Gonzalez-Manteiga, and J. M. Prada-Sanchez (2004): "Bootstrapping the Dorfman-Hall-Chambers-Dunstan estimator of a finite population distribution function," *Journal of Nonparametric Statistics*, 16, 63–90.

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.

Molina, I. and J. N. K. Rao (2010): "Small area estimation of poverty indicators," *The Canadian Journal of Statistics*, 38, 369–385.

Newey, W. K. and J. L. Powell (1987): "Asymmetric Least Squares Estimation and Testing," *Econometrica*, 55, 819.

Pratesi, M., M. G. Ranalli, and N. Salvati (2009): "Nonparametric M-quantile regression using penalised splines," *Journal of Nonparametric Statistics*, 21, 287–304.

R Core Team (2017): *R: A Language and Environment for Statistical Computing*, Vienna, Austria: R Foundation for Statistical Computing.

Rao, J. N. K. and I. Molina (2015): *Small area estimation*, Wiley series in survey methodology, Hoboken, New Jersey: John Wiley & Sons, Inc, second edition ed.

Schmid, T. (2011): "Spatial Robust Small Area Estimation applied on Business data," Ph.D. thesis, University of Trier, Trier.

Schmid, T., F. Bruckschen, N. Salvati, and T. Zbiranski (2017): "Constructing sociodemographic indicators for national statistical institutes by using mobile phone data: estimating literacy rates in Senegal," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 1–28.

Schoch, T. (2012): "Robust unit-level small area estimation: a fast algorithm for large data sets," *Austrian Journal of Statistics*, 41, 243–265.

——— (2014): *rsae: Robust Small Area Estimation*, R package version 0.1-5.

Sobotka, F., S. Schnabel, L. S. Waltrup, P. Eilers, T. Kneib, and G. Kauermann (2014): *expectreg: Expectile and Quantile Regression*, R package version 0.39.

Tzavidis, N., S. Marchetti, and R. Chambers (2010): "Robust estimation of Small Area Means and Quantiles," *Australian & New Zealand Journal of Statistics*, 52, 167–186.

Tzavidis, N., N. Salvati, M. Pratesi, and R. Chambers (2008): "M-quantile models with application to poverty mapping," *Statistical Methods and Applications*, 17, 393–411.

Tzavidis, N., N. Salvati, T. Schmid, E. Flouri, and E. Midouhas (2016): "Longitudinal analysis of the strengths and difficulties questionnaire scores of the Millennium Cohort Study children in England using M-quantile random-effects regression," *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 179, 427–452.

United Nations, ed. (2016): *The Sustainable Development Goals Report*, New York.

Venables, W. N. and B. D. Ripley (2002): *Modern Applied Statistics with S*, New York: Springer, fourth ed.

Welsh, A. H. and A. M. Richardson (1997): "13 Approaches to the robust estimation of mixed models," in *Handbook of Statistics*, Elsevier, vol. 15 of *Robust Inference*, 343–384.

Wickham, H. (2014): *Advanced R*, Boca Raton, Florida: Chapman and Hall, 1 ed.

Wickham, H., P. Danenberg, and M. Eugster (2017): *roxygen2: In-Line Documentation for R*, R package version 6.0.1.

A Figures

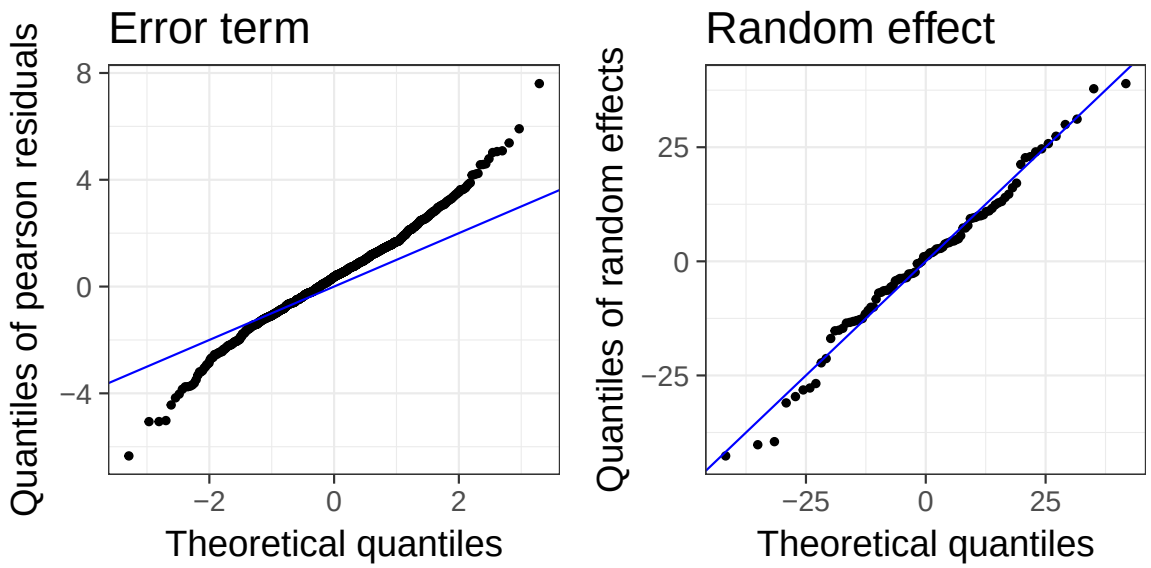

Figure 38: QQ-plot of EBP estimation residuals in the poverty mapping for Austria application

B R-Listings

Listing 1: Example R output from the function "mq" based on EU-SILC data

```
Call:
 eqIncome ~ gender + eqsize + cash + self_empl + unempl_ben

Number of Iterations until Convergence:
0.25  0.5 0.75
   7    6    7

Residuals:
      0.25              0.5                0.75
 Min.   :-10565   Min.   :-14014.6   Min.   :-18133.5
 1st Qu.: -1710   1st Qu.: -4622.7   1st Qu.: -8527.3
 Median :  2295   Median :  -604.3   Median : -4380.8
 Mean   :  3916   Mean   :   939.4   Mean   : -2957.1
 3rd Qu.:  7600   3rd Qu.:  4534.7   3rd Qu.:   545.2
 Max.   : 68248   Max.   : 65732.6   Max.   : 62209.2

Coefficients:
                    0.25        0.5        0.75
(Intercept)  13870.9179 17272.1853 21453.2348
genderfemale  -507.5561  -437.6011  -669.0984
eqsize       -2639.2130 -2657.0015 -2486.0363
```

```
cash             0.4485     0.4171     0.3815
self_empl        0.4900     0.4795     0.4742
unempl_ben      -0.0333    -0.0543    -0.0855

Estimators of Scale:
    0.25      0.5     0.75
6273.023 6777.362 9401.558

Pseudo R-squared:
0.25  0.5 0.75
0.16 0.31 0.23
```

Listing 2: Example R output from the function "mmqm" based on EU-SILC data (state-level)

```
Mixed M-Quantile Model

      Formula:  eqIncome ~ gender + eqsize + cash + self_empl + unempl_ben

Domains: "state"

Pseudo Random Effects per Domain
          state (Intercept) genderfemale    eqsize      cash self_empl  unempl_ben
1    Burgenland    16453.67    -426.2310 -2659.629 0.4249936 0.4833912 -0.05415672
2 Lower Austria    16977.50    -436.5673 -2657.319 0.4198912 0.4808733 -0.05298375
3        Vienna    16382.26    -425.4878 -2658.551 0.4256620 0.4836141 -0.05405575
4     Carinthia    16085.54    -425.5975 -2654.270 0.4284935 0.4844157 -0.05374083
5        Styria    16645.22    -431.1970 -2658.698 0.4231396 0.4824453 -0.05395697
6 Upper Austria    17298.48    -438.5769 -2656.357 0.4168088 0.4793236 -0.05453722
7      Salzburg    16992.76    -436.4535 -2657.397 0.4197502 0.4808103 -0.05304320
8         Tyrol    16383.45    -425.3881 -2658.611 0.4256532 0.4836137 -0.05405349
9    Vorarlberg    17277.92    -437.9555 -2656.776 0.4170109 0.4794587 -0.05434892


Number of Observations: 1000
Number of Domains: 9
```

Listing 3: Example R output from the function "mq_sae"

```
M-Quantile Small Area Model

Out-of-sample domains:  3
In-sample domains:  93

Model fit:
For model fit mqmm methods are applicable to emdiObject$model
where df equals smp_data
and where call equals eqIncome ~ gender + eqsize + cash + self_empl + unempl_ben
```

Listing 4: R code from the function "mq_sae" as an example for the coding philosophy (without roxygen block)

```
                    pop_domains,
                    smp_data,
                    smp_domains,
                    L = 50,
                    threshold = NULL,
                    MSE = FALSE,
                    #smoothed = 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,
                    ...
) {
  # smoothing currently disabled
  smoothed = F

  mq_sae_check1(fixed = fixed, pop_data = pop_data, pop_domains = pop_domains,
              smp_data = smp_data, smp_domains = smp_domains, L = L)

  mq_sae_check2(threshold = threshold, MSE = MSE,  B = B, S = S, smoothed = smoothed,
              custom_indicator = custom_indicator, cpus = cpus,  seed = seed,
              na.rm = na.rm)



  # Save function call -----------------------------------------------------
  call <- match.call()

  # Set Seed ---------------------------------------------------------------
  if(!is.null(seed)) {
    if (cpus > 1 && parallel_mode != "socket") {
      RNG_kind <- RNGkind()
      set.seed(seed, kind = "L'Ecuyer")
    }
    else
    {
      set.seed(seed)
    }
  }

  # The function framework_mq_sae can be found in script framework_mq_sae.R
```

```
framework <-  framework_mq_sae(pop_data         = pop_data,
                               pop_domains      = pop_domains,
                               smp_data         = smp_data,
                               smp_domains      = smp_domains,
                               custom_indicator = custom_indicator,
                               fixed            = fixed,
                               threshold        = threshold,
                               na.rm            = na.rm
                               )



# Point Estimation ------------------------------------------------------------

# The function point_estim can be found in script point_estimation.R
point_estimates <- point_estim_mq(framework  = framework,
                                  fixed      = fixed,
                                  L          = L,
                                  keep_data  = TRUE,
                                  ...
                                  )



# MSE Estimation --------------------------------------------------------------

if (MSE == TRUE) {

# The function bootstrap_mq_sae can be found in script mse_estimation_mq_sae.R
  mse_estimates <-  bootstrap_mq_sae(framework       = framework,
                                     point_estimates = point_estimates,
                                     fixed           = fixed,
                                     L               = L,
                                     B               = B,
                                     S               = S,
                                     parallel_mode   = parallel_mode,
                                     cpus            = cpus,
                                     smoothed        = smoothed,
                                     ...
                                     )



  mq_sae_out <- list(ind          = point_estimates$ind,
                     MSE          = mse_estimates,
                     model        = point_estimates$model,
                     framework    = framework[c("N_dom_unobs",
                                                "N_dom_smp",
```

```
                                                "N_smp",
                                                "N_pop",
                                                "smp_domains",
                                                "smp_data",
                                                "smp_domains_vec",
                                                "pop_domains_vec")],
                    method          = "M-quantile/IRWLS",
                    fixed           = fixed,
                    call            = call
                    )
  } else {

    mq_sae_out <- list(ind          = point_estimates$ind,
                       MSE          = NULL,
                       model        = point_estimates$model,
                       framework    = framework[c("N_dom_unobs",
                                                "N_dom_smp",
                                                "N_smp",
                                                "N_pop",
                                                "smp_domains",
                                                "smp_data",
                                                "smp_domains_vec",
                                                "pop_domains_vec")],
                       method       = "MQ",
                       fixed        = fixed,
                       call         = call,
                       ...
                    )
  }

  if (cpus > 1 && parallel_mode != "socket") {
    RNGkind(RNG_kind[1]) # restoring RNG type
  }
  class(mq_sae_out) <- c("emdi", "mqmodel")
  return(mq_sae_out)
}
```

C Digital Resources

Name	Description
mquantreg_0.1.0.tar.gz	R-Package, to be installed in `R`. **Note: `data.table, Hmisc and ggplot2` must be installed first.** See example.R
mquantreg.pdf	PDF manual of the package
master_thesis_tammena.pdf	PDF version of the thesis
presentation_tammena.pdf	PDF version of the presentation held in the seminar
examples.R	Several R-examples, including map-plotting in conjunction with the emdi package
application	Folder with R-files relevant for the application
replication	Folder with R-files relevant for the replication, including setup, results and analysis
simulation	Folder with R-files relevant for the simulations, including setup, results and analysis
theory	Folder with R-files relevant for the main examples in the theoretical part
tex_sources	Folder with LaTeXsource files of the application

Table 20: Digitally provided resources. Note that the supplementary R-Code in the folders application, replication, simulation and theory is unformatted and the working directory needs to be adjusted

Declaration of Authorship/Eidesstattliche Erklärung

I hereby confirm that I have authored this Master's thesis independently and without use of others than the indicated sources. All passages which are literally or in general matter taken out of publications or other sources are marked as such.

Berlin, December 20, 2017

Enno Tammena