The R Package smicd: Statistical Methods for Interval-Censored Data

by Paul Walter

Abstract The package allows the use of two new statistical methods for the analysis of interval-censored data: 1) direct estimation/prediction of statistical indicators and 2) linear (mixed) regression analysis. Direct estimation of statistical indicators, for instance, poverty and inequality indicators, is facilitated by a non parametric kernel density algorithm. The algorithm is able to account for weights in the estimation of statistical indicators. The standard errors of the statistical indicators are estimated with a non parametric bootstrap. Furthermore, the package offers statistical methods for the estimation of linear and linear mixed regression models with an interval-censored dependent variable, particularly random slope and random intercept models. Parameter estimates are obtained through a stochastic expectation-maximization algorithm. Standard errors are estimated using a non parametric bootstrap in the linear regression model and by a parametric bootstrap in the linear mixed regression model. To handle departures from the model assumptions, fixed (logarithmic) and data-driven (Box-Cox) transformations are incorporated into the algorithm.

Introduction

Interval-censored or grouped data occurs when only the lower A_{k-1} and upper A_k interval bounds (A_{k-1}, A_k) of a variable are observed, and its true value remains unknown. Instead of measuring the variable of interest on a continuous scale, for instance, income data, the scale is divided into n_k intervals. The variable k $(1 \leq k \leq n_k)$ indicates in which of the n_k intervals an observation falls into. This leads to a loss of information since the shape of the distribution within the intervals remains unknown. In the field of survey statistics, asking for interval-censored data is often done in order to avoid item non-response and thus increase data quality. Item non-response is avoided because interval-censored data offers a higher level of data privacy protection (Hagenaars and Vos, 1988; Moore and Welniak, 2000). Among others, popular surveys and censuses that collect interval-censored data are the German Microcensus (Statistisches Bundesamt, 2017), the Colombian census (Departamento Administrativo Nacional De Estadística, 2005), and the Australian census (Australian Bureau of Statistics, 2011). While item non-response is reduced or avoided, the statistical analysis of the data requires more elaborate mathematical methods. Even statistical indicators that are easily calculated for metric data, e.g., the mean, cannot be estimated using standard formulas (Fahrmeir et al., 2016). Also, estimating linear and linear mixed regression models, which are applied in many fields of statistics, requires advanced statistical methods when the dependent variable is interval censored. Therefore, the presented R package implements three major functions: `kdeAlgo()` to estimate statistical indicators (e.g., the mean) from interval-censored data, `semLm()`, and `semLme()` to estimate linear and linear mixed regression models with an interval-censored dependent variable. The package code and the open-source contribution guidelines for the package are available on GitHub. Potential code contributions, feature requests, and bugs can be reported there by creating issues.

For the estimation of statistical indicators from interval-censored data, different approaches are described in the literature. These approaches can be broadly categorized into four groups: Estimation on the midpoints (Fahrmeir et al., 2016), linear interpolation of the distribution function, non parametric modeling via splines (Berger and Escobar, 2016), and fitting a parametric distribution function to the censored data (Dagum, 2008; McDonald, 1984; Bandourian et al., 2002). Some of these methods are implemented in R packages available on the Comprehensive R Archive Network (CRAN). The method of linear interpolation is implemented for the estimation of quantiles in the R package **actuar** (Goulet et al., 2020; Dutang et al., 2008). The package also enables the estimation of the mean on the interval midpoints. Fitting a parametric distribution to interval-censored data can be done by using the R package **fitdistrplus** (Delignette-Muller et al., 2020; Delignette-Muller and Dutang, 2015).

In survey statistics, interval-censored data is often collected for income or wealth variables. Thus, the performance of the above-mentioned methods is commonly evaluated by simulation studies that rely on data that follows some kind of income distribution. The German statistical office (DESTATIS) uses the method of linear interpolation for the estimation of statistical indicators from interval-censored income data collected by the German Microcensus. This approach gives the same results as assuming a uniform distribution within the income intervals. Estimation results are reasonably accurate if the estimated indicators do not depend on the whole shape of the distribution, e.g., the median (Lenau and Münnich, 2016). Fitting a parametric distribution to the data enables the estimation of indicators that rely on the whole shape of the distribution. This method works well when the data is censored to

only a few equidistant intervals (Lenau and Münnich, 2016). Non parametric modeling via splines shows especially good results for a high number of intervals in ascending order (Lenau and Münnich, 2016). However, according to Lenau and Münnich (2016), all of the above-mentioned methods show large biases and variances when the estimation is based on a small number of intervals. Therefore, a novel kernel density estimation (KDE) algorithm is implemented in the **smicd** package that overcomes the drawbacks of the previously mentioned methods (Walter, 2019, 2020). The algorithm bases the estimation of statistical indicators on pseudo samples that are drawn from a fitted non parametric distribution. The method automatically adapts to the shape of the true unknown distribution and provides reliable estimates for different interval-censoring scenarios. It can be applied via the function `kdeAlgo()`.

Similar to the direct estimation of statistical indicators from interval-censored data, there exists a variety of ad-hoc approaches and explicitly formulated mathematical methods for the estimation of linear regression models with an interval-censored dependent variable. The following methods and approaches are used for handling interval-censored dependent variables within linear regression models: Ordinary least squares (OLS) regression on the midpoints (Thompson and Nelson, 2003), ordered logit- or probit-regression (McCullagh, 1980), and regression methodology formulated for left-, right-, and interval-censored data (Tobin, 1958; Rosett and Nelson, 1975; Stewart, 1983). All of these methods are implemented in different R packages available on CRAN. OLS regression on the midpoints is applicable by using the `lm()` function from the **stats** package (R Core Team, 2020), ordered logit regression is implemented in the **MASS** package (Ripley, 2019; Venables and Ripley, 2002), and interval regression is implemented in the **survival** (Therneau, 2020; Therneau and Grambsch, 2000) package.

While OLS regression on the midpoints of the intervals is easily applied, it comes with the disadvantage of giving biased estimation results (Cameron, 1987). This approach disregards the uncertainty stemming from the unknown true distribution of the data within the intervals, and therefore, leads to biased parameter estimates. Its performance relies on the number of intervals, and estimation results are only comparable to more advanced methods when the number of intervals is very large (Fryer and Pethybridge, 1972). Conceptualizing the model as an ordered logit or probit regression is feasible by treating the dependent variable as an ordered factor variable (McCullagh, 1980). However, this approach also neglects the unknown distribution of the data within the intervals. Furthermore, the predicted values are not on a continuous scale but are in terms of the probability of belonging to a certain group. To overcome these disadvantages and obtain unbiased estimation results Stewart (1983) introduces regression methodology for models with an interval-censored dependent variable. Walter (2019) further develops his approach and introduces a novel stochastic expectation-maximization (SEM) algorithm for the estimation of linear regression models with an interval-censored dependent variable that is implemented in the **smicd** package. The model parameters are unbiasedly estimated as long as the model assumptions are fulfilled. The function `semLm()` provides the SEM algorithm and enables the use of fixed (logarithmic) and data-driven (Box-Cox) transformations (Box and Cox, 1964). The Box-Cox transformation automatically adapts to the shape of the data and transforms the dependent variable in order to meet the model assumption.

In order to analyze longitudinal or clustered data (e.g., students within schools), linear mixed regression models are applicable. These kinds of models control for the correlated structure of the data by including random effects in addition to the usual fixed effects. In order to deal with an interval-censored dependent variable in linear mixed regression models, there are several approaches described in the literature. Linear mixed regression models, just like linear regression models, can be estimated on the interval midpoints of the censored-dependent variable. Furthermore, conceptualizing the model as an ordered logit or probit regression model is feasible (Agresti, 2010). These approaches inherit the same advantages and disadvantages as previously discussed. Linear mixed regression on the midpoints can be applied by the **lme4** (Bates et al., 2020b, 2015) or **nlme** (Pinheiro et al., 2020) package and the ordered logit regression is implemented in the **ordinal** package (Christensen, 2019). To my knowledge, there are no R packages for the estimation of linear mixed regression models with an interval-censored dependent variable. Therefore, the package **smicd** contains the SEM algorithm proposed by Walter (2019) for the estimation of linear mixed regression models with an interval-censored dependent variable. If the model assumptions are fulfilled, the method gives unbiased estimation results. The function `semLme()` enables the estimation of the regression parameters, and it also allows for the usage of the logarithmic and Box-Cox transformation in order to fulfill the model assumptions (Gurka et al., 2006).

The paper is structured into two main sections. Section 2.2 deals with the direct estimation of statistical indicators from interval-censored data, whereas Section 2.3 introduces linear and linear mixed regression models with an interval-censored dependent variable. Both sections have been divided into three subsections: first, the statistical methodology is introduced, then the core functions of the **smicd** package are presented, and finally, illustrative examples with two different data sets are provided. In Section 2.4, the main results are summarized, and an outlook is given.

Direct estimation of statistical indicators

In the following three subsections, the methodology for the direct estimation of statistical indicators from interval-censored data is introduced, the core functionality of the function `kdeAlgo()` is presented, and statistical indicators are estimated using the synthetic EU-SILC (European Union Statistics on Income and Living Conditions) data set from Austria.

Methodology: Direct estimation of statistical indicators

In order to estimate statistical indicators from interval-censored data, the proposed algorithm generates metric pseudo samples of an interval-censored variable. These pseudo samples can be used to estimate any statistical indicator. They are drawn from a non parametrically estimated kernel density. Kernel density estimation was first introduced by (Rosenblatt, 1956) and (Parzen, 1962). By its application, the density $f(x)$ of a continuous independently and identically distributed random variable is estimated without assuming any distributional shape of the data. The estimator is defined as:

$$\hat{f}_h(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right), \qquad i = 1, \ldots, n,$$

where $K(\cdot)$ is a kernel function, $h > 0$ the bandwidth, and $x = \{x_1, x_2, \ldots, x_n\}$ denotes a sample of size n. The performance of the estimator is determined by the optimal choice of h. The selection of an optimal h is widely discussed in the literature; see Jones et al. (1996); Loader (1999); Zambom and Dias (2012). When working with interval-censored data, a standard KDE cannot be applied since x is not observed on a continuous scale. Nevertheless, its unobserved true distribution is of continuous form. As an ad hoc solution, the density $\hat{f}_h(x)$ can be estimated based on the interval midpoints. The resulting density estimate will be spiky unless the bandwidth is sufficiently large. A large bandwidth, however, leads to a loss of information (Wang and Wertelecki, 2013). Therefore, Walter (2019) proposes an iterative KDE algorithm for density estimation from interval-censored data. The approach is based on Groß et al. (2017), who introduce a similar KDE algorithm in a two-dimensional setting with an equidistant interval width. Walter (2019) shows that the algorithm can be adjusted to one-dimensional data with an arbitrary class width. For the estimation of linear and non-linear statistical indicators, the unknown distribution of x has to be reconstructed by using the observed interval $k = \{k_1, k_2, \ldots, k_n\}$ that an observation falls into. From Bayes' theorem (Bayes, 1763), it follows that the conditional distribution of $(x|k)$ is:

$$\pi(x|k) \propto \pi(k|x)\, \pi(x),$$

with $\pi(k|x)$ is defined by a product of a Dirac distribution $\pi(k|x) = \prod_{i=1}^{n} \pi(k_i|x_i)$ with

$$\pi(k_i|x_i) = \begin{cases} 1 & \text{if } A_{k_i-1} \leq x_i \leq A_{k_i}, \\ 0 & \text{else,} \end{cases}$$

for $i = 1, \ldots, n$. Since $\pi(x)$ is unknown, it is replaced by a kernel density estimate $\hat{f}_h(x)$.

Estimation and computational details

To fit the model, pseudosamples of x_i are drawn from the conditional distribution

$$\pi(x_i|k_i) \propto \mathbf{I}\left(A_{k_i-1} \leq x_i \leq A_{k_i}\right) f(x_i),$$

where $\mathbf{I}(\cdot)$ denotes the indicator function. The conditional distribution of $\pi(x_i|k_i)$ is given by the product of a uniform distribution and density $f(x_i)$. As the density is unknown, it is replaced by an estimate $\hat{f}_h(x)$, which is obtained by the KDE. In particular, x_i is repeatedly drawn from the given interval $\left(A_{k_i-1}, A_{k_i}\right)$ by using the current density estimate $\hat{f}_h(x)$ as a sampling weight. The explicit steps of the iterative algorithm as given in Walter (2019) are stated below:

1. Use the midpoints of the intervals as pseudo $\tilde{x}_i$ for the unknown x_i. Estimate a pilot estimate of $\hat{f}_h(x)$ by applying KDE. Note: Choose a sufficiently large bandwidth h in order to avoid rounding spikes.
2. Evaluate $\hat{f}_h(x)$ on an equal-spaced grid $G = \{g_1, \ldots, g_j\}$ with grid points $g_1, \ldots, g_j$. The width of the grid is denoted by δ_g. It is given by

$$\delta_g = \frac{|A_0 - A_{n_k}|}{j - 1},$$

and the grid is defined as:

$$G = \{g_1 = A_0, g_2 = A_0 + \delta_g, g_3 = A_0 + 2\delta_g, \ldots, g_{j-1} = A_0 + (j-2)\,\delta_g, g_j = A_{n_k}\}.$$

3. Sample from $\pi(x|k)$ by drawing randomly from $G_k = \{g_j | g_j \in (A_{k-1}, A_k)\}$ with sampling weights $\hat{f}_h(\tilde{x}_i)$ for $k = 1, \ldots, n_k$. The sample size for each interval is given by the number of observations within each interval. Obtain $\tilde{x}_i$ for $i = 1, \ldots, n$.
4. Estimate any statistical indicator of interest $\hat{I}$ using $\tilde{x}_i$.
5. Recompute the density $\hat{f}_h(x)$ using the pseudo samples $\tilde{x}_i$ obtained in iteration Step 3.
6. Repeat Steps 2-5, with $B^{(KDE)}$ burn-in and $M^{(KDE)}$ additional iterations.
7. Discard the $B^{(KDE)}$ burn-in iterations and estimate the final $\hat{I}$ by averaging the obtained $M^{(KDE)}$ estimates.

For open-ended intervals, e.g., $(15000, +\infty)$, the upper bound has to be replaced by a finite number. Walter (2019) shows through model-based simulations that a value of three times the value of the lower bound $(15000, 45000)$ gives appropriate estimation results when working with income data.

The variance of the statistical indicators is estimated by bootstrapping. Bootstrap methods were first introduced by Efron (1979). These methods serve as an estimation procedure when the variance cannot be stated as a closed-form solution (Shao and Tu, 1995). While bootstrapping avoids the problem of the non-availability of a closed-form solution, it comes with the disadvantage of long computational times. In the package, a non parametric bootstrap that accounts for the additional uncertainty coming from the interval-censored data is implemented. This non parametric bootstrap is introduced in Walter (2019).

Core functionality: Direct estimation of statistical indicators

The presented KDE algorithm is implemented in the function `kdeAlgo()` (see Table 1). The arguments and default settings of `kdeAlgo()` are briefly summarized in Table 2. The function gives back an S3 object of class `"kdeAlgo"`. A detailed explanation of all components of a `"kdeAlgo"` object can be found in the package documentation. The generic functions `plot()` and `print()` can be applied to `"kdeAlgo"` objects to output the main estimation results (see Table 1). In the next section, the function `kdeAlgo()` is used to estimate a variety of statistical indicators from interval-censored EU-SILC data, and its arguments are explained in more detail.

Table 1: Implemented functions for the direct estimation of statistical indicators.

Function Name	Description
`kdeAlgo()`	Estimates the statistical indicators and its standard errors from interval-censored data
`plot()`	Plots convergence of the estimated statistical indicators and estimated density of the pseudo $\tilde{x}_i$
`print()`	Prints the estimated statistical indicators and its standard errors

Example: Direct estimation of statistical indicators

To demonstrate the function `kdeAlgo()`, the equivalized household income and the corresponding household sample weight from the Austrian synthetic EU-SILC survey data set are used. The data set is included in the **laeken** package (Alfons et al., 2020; Alfons and Templ, 2013). Its synthetic nature makes it unusable for inferential statistics. However, the data set has the advantage over the scientific use file by being freely available which enables the easy reproducibility of the presented example. Since the total disposable household income is measured on a continuous scale, it is censored to 22 intervals for demonstration purposes. For a realistic censoring scheme, the interval bounds are chosen such that they closely follow the interval bounds used in the German Microcensus from 2013 (Statistisches Bundesamt, 2014). The German Microcensus is a representative household survey that covers 830,000 persons in 370,000 households (1% of the German population) in which income is only collected as an interval-censored variable (Statistisches Bundesamt, 2016).

In a first step, the variable total disposable household income called `hhincome_net` is interval-censored according to 22 intervals using the function `cut()`. The vector of interval bounds is called `intervals`, and the newly obtained interval-censored income variable is called `c.hhincome`.

Table 2: Arguments of function `kdeAlgo()`.

Argument	Description	Default
`xclass`	Interval-censored variable	
`classes`	Numeric vector of interval bounds	
`threshold`	Threshold used for poverty indicators (% of the median of the target variable)	`0.6`
`burnin`	Number of burn-in iterations $B^{(KDE)}$	`80`
`samples`	Number of additional iterations $M^{(KDE)}$	`400`
`bootstrap.se`	If TRUE, standard errors of the statistical indicators are estimated	`FALSE`
`b`	Number of bootstraps for the estimation of the standard errors	`100`
`bw`	Smoothing bandwidth used	`"nrd0"`
`evalpoints`	Number of evaluation grid points	`4000`
`adjust`	Bandwidth multiplier $bw = adjust * bw$	`1`
`custom_indicator`	A list of user-defined statistical indicators	`NULL`
`upper`	If upper bound of the upper interval is $+\infty$, e.g., $(15000, +\infty)$, then $+\infty$ is replaced by $15000 * upper$	`3`
`weights`	Survey weights	`NULL`
`oecd`	Household weights of equivalence scale	`NULL`

```
R> intervals <- c(
+    0, 150, 300, 500, 700, 900, 1100, 1300, 1500, 1700, 2000, 2300, 2600, 2900,
+    3200, 3600, 4000, 4500, 5000, 5500, 6000, 7500, Inf
+  )
R> c.hhincome <- cut(hhincome_net, breaks = intervals)
```

In order to get a descriptive overview of the distribution of the censored income data, the function `table()` is applied.

```
R> table(c.hhincome)
c.hhincome
          (0,150]         (150,300]         (300,500]
               66               113               280
        (500,700]         (700,900]     (900,1.1e+03]
              462              1137              1433
(1.1e+03,1.3e+03] (1.3e+03,1.5e+03] (1.5e+03,1.7e+03]
             2040              1811              1671
  (1.7e+03,2e+03]   (2e+03,2.3e+03] (2.3e+03,2.6e+03]
             2006              1383               849
(2.6e+03,2.9e+03] (2.9e+03,3.2e+03] (3.2e+03,3.6e+03]
              508               389               242
  (3.6e+03,4e+03]   (4e+03,4.5e+03]   (4.5e+03,5e+03]
              158               107                61
  (5e+03,5.5e+03]   (5.5e+03,6e+03]   (6e+03,7.5e+03]
               21                18                52
    (7.5e+03,Inf]
               17
```

Most incomes are in interval $(1100, 1300]$, and only 17 incomes are in the upper interval. For the estimation of the statistical indicators, the function `kdeAlgo()` of the **smicd** package is called with the following arguments.

```
R> Indicators <- kdeAlgo(
+    xclass = c.hhincome, classes = intervals,
+    bootstrap.se = TRUE, custom_indicator =
+     list(
+      quant05 = function(y, threshold, weights) {
+       wtd.quantile(y, probs = 0.05, weights)
```

```
+      },
+      quant95 = function(y, threshold, weights) {
+       wtd.quantile(y, probs = 0.95, weights)
+      }
+     ),
+    weights = hhweight
+  )
```

The variable `c.hhincome` is assigned to the argument `xclass`, and the vector of interval bounds `intervals` is assigned to the argument `classes`. The default settings of the arguments `burnin`, `samples`, `bw`, `evalpoints`, `adjust`, and `upper` are retained. Simulation results from Walter (2019) and Groß et al. (2017) show that these settings give good results when working with income data. Changing these arguments has an impact on the performance of the KDE algorithm. As default, the statistical indicators: mean, Gini coefficient, headcount ratio (HCR), the quantiles (10%, 25%, 50%, 75%, 90%), the poverty gap (PGAP), and the quintile share ratio (QSR) are estimated (Gini, 1912; Foster et al., 1984). The HCR and PGAP rely on a poverty threshold. The default choice of the `threshold` argument is 60% of the median of the target variable, as suggested by Eurostat (2014). Besides the mentioned indicators, any other statistical indicator can be estimated via the argument `custom_indicator`. In the example, the argument is assigned a list that holds functions to estimate the 5% and 95% quantile. The custom indicators must depend on the target variable, the threshold (even if it is not needed for the specified indicator), and optionally on the weights argument if the estimation of a weighted indicator is required. To estimate the standard errors of all indicators, `bootstrap.se = TRUE`, and the number of bootstrap samples is 100 (the default value as suggested in Walter (2019)). Lastly, the household weight (`hhweight`) is assigned to the argument `weights` in order to estimate weighted statistical indicators. It can also be controlled for households of different sizes by assigning `oecd` a variable with household equivalence weights. By applying the `print()` function to the `"kdeAlgo"` object, the estimated statistical indicators (default and custom indicators) as well as their standard errors are printed. For instance, in this example, the estimated mean is about 1,658 Euro and its standard error is 8.486.

```
R> print(Indicators)
Value:
    mean   gini    hcr  quant10   quant25   quant50
1658.329  0.265  0.145  802.227  1117.714  1507.947
 quant75   quant90   pgap     qsr  quant05   quant95
 2020.063 2654.707  0.040   3.920  630.326  3142.296

Standard error:
   mean   gini    hcr quant10 quant25 quant50
  8.486  0.002  0.002   5.839   5.977   6.605
quant75 quant90   pgap    qsr quant05 quant95
 10.548  21.622  0.001  0.044  10.327  24.401
```

For demonstration purposes, the statistical indicators are also estimated using the continuous household income variable from the synthetic EU-SILC data set (Table 3). The estimation results of the KDE algorithm using the interval-censored data are very close to those based on the continuous data. Slightly larger deviations are observable for the more extreme quantiles. This is due to the fact that these quantile estimates fall into intervals with a lower number of observations (compared to the other quantile estimates). Estimation results for these quantiles could potentially be further improved by increasing the number of `evalpoints` of the `kdeAlgo()`.

Table 3: Estimated weighted statistical indicators using the continuous household income variable from the synthetic EU-SILC data set.

mean	gini	hcr	quant10	quant25	quant50
1657.910	0.265	0.144	805.468	1114.028	1508.657
quant75	**quant90**	**pgap**	**qsr**	**quant05**	**quant95**
2017.585	2653.617	0.040	3.960	619.666	3153.425

In Walter (2019), the performance of the KDE algorithm is evaluated via detailed simulation studies. By applying the function `plot()`, `"kdeAlgo"` objects can be plotted. Thereby, convergence plots for all estimated statistical indicators and a plot of the estimated final density are obtained.

```
R> plot(Indicators)
```

Figure 1 shows convergence plots for two of the estimated indicators. Additionally, a plot of the estimated final density with a histogram of the observed data in the background is given in Figure 2. In Figure 1, the estimated statistical indicator (Gini, 10% quantile) for each iteration step of the KDE algorithm and the average over the estimates up to iteration step M (excluding the burn-in iterations) are plotted. A vertical line marks the end of the burn-in period. The horizontal line gives the value of the final estimate (average over the M iterations). All convergence plots indicate that the number of iterations is chosen sufficiently large for the estimates to converge.

If convergence were not achieved, the arguments `burnin` and `samples` should be increased. It is notable that the estimated 10% quantile has the same value for almost all iterations steps. This is the case because the quantile, as any other statistical indicator, is estimated using the pseudo samples that are drawn on 4,000 grid points G. Estimating a quantile on only 4,000 unique outcomes (pseudo values) leads to equal quantile estimates for numerous iteration steps of the KDE algorithm.

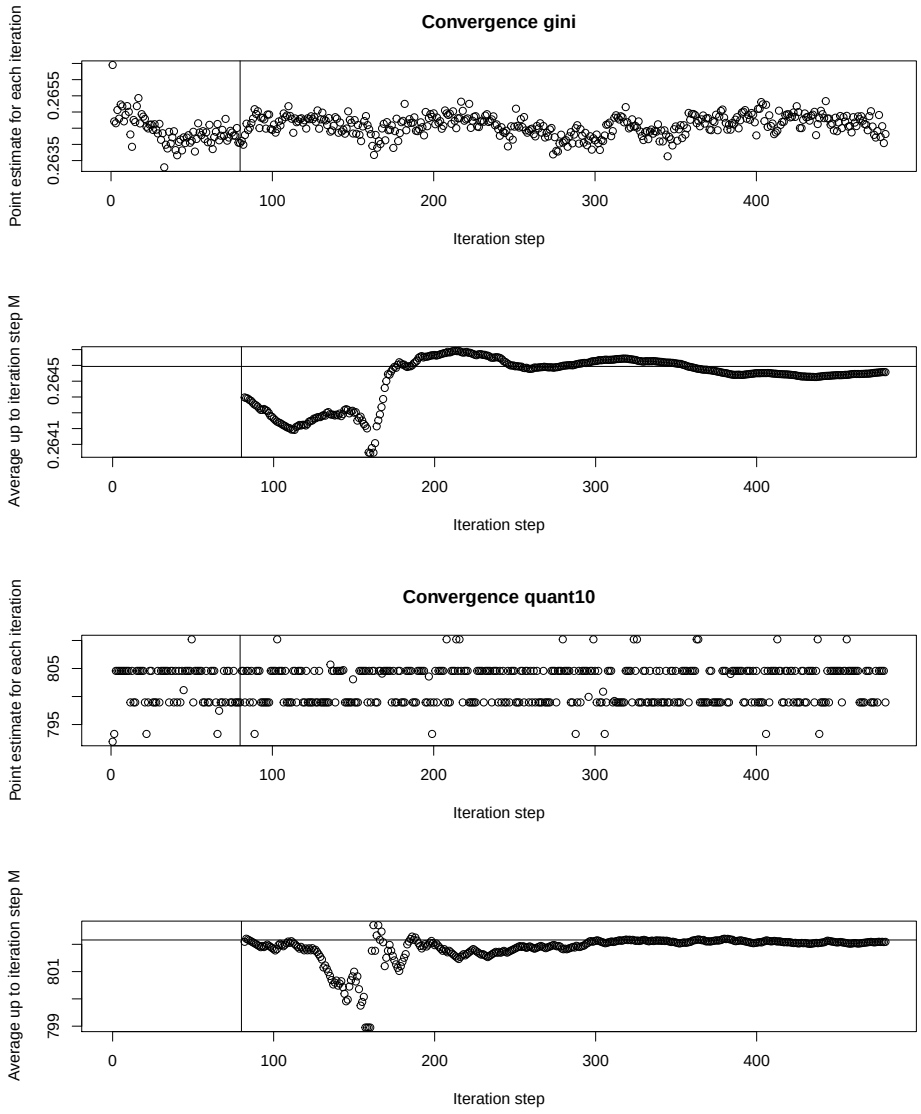

Figure 1: Convergence plots of the estimated indicators (Gini and 10% quantile).

Lastly, it should be mentioned that the computation time can be very long if the estimation of the standard errors is enabled. Hence, if the estimation of the standard errors is not required, the argument `bootstrap.se` should be set to `FALSE`. Furthermore, it should always be checked how many iterations are needed for the desired statistical indicator to converge. Reducing the number of required iterations (arguments `burnin` and `samples`) lowers the computation time significantly (with and without standard errors).

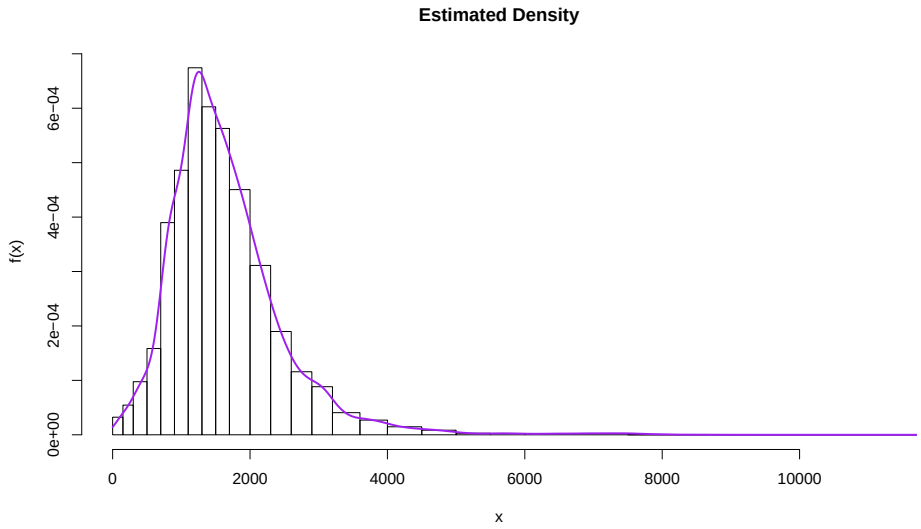

Figure 2: Estimated final density with a histogram of the observed distribution of the data in the background.

Regression analysis

In the following three subsections, the statistical methodology for linear and linear mixed regression models with an interval-censored dependent variable is introduced, the core functionality of the functions `semLM()` and `semLME()` is presented, and examination scores of students from schools in London are exemplary modeled.

Methodology: Regression analysis

The theoretical introduction of the new regression method, proposed by Walter (2019), is presented for linear mixed regression models. The theory for linear regression models can be obtained by simplifying the introduced method. In its standard form, the linear mixed regression model serves to analyze the linear relationship between a continuous dependent variable and some independent variables (Goldstein, 2010). Random parameters (random slopes and random intercepts) are included in the model to account for correlated data, e.g., students within schools. The model in matrix notation (Laird and Ware, 1983) is given by

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{v} + \mathbf{e}, \tag{1}$$

where $\mathbf{y}$ is an $n \times 1$ column vector of the dependent variable, n is the sample size, $\mathbf{X}$ is a $n \times p$ matrix where p is equal to the number of predictors, $\boldsymbol{\beta}$ is a column vector of the fixed effects regression parameters of size $p \times 1$, $\mathbf{Z}$ is the $n \times q$ design matrix with q random effects, $\mathbf{v}$ is a $q \times 1$ vector of random effects, and $\mathbf{e}$ is the residual vector of size $n \times 1$. The distribution of the random effects is given by

$$\mathbf{v} \sim N(\mathbf{0}, \mathbf{G}), \qquad \text{where } \mathbf{G} = \begin{bmatrix} \sigma_0^2 & \sigma_{01} & \dots & \sigma_{0q} \\ \sigma_{10} & \sigma_1^2 & \dots & \sigma_{1q} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{q0} & \sigma_{q1} & \dots & \sigma_q^2 \end{bmatrix},$$

and the distribution of the residuals is given by $\mathbf{e} \sim N(\mathbf{0}, \mathbf{R})$ with $\mathbf{R} = \mathbf{I}_n \sigma_e^2$, where $\mathbf{I}_n$ is the identity matrix, and σ_e^2 is the residual variance. The random effects $\mathbf{v}$ and the residuals $\mathbf{e}$ are assumed to be independent. For a more detailed introduction of mixed models, see Searle et al. (1992); McCulloch et al. (2008); Snijders and Bosker (2011). In the case of an interval-censored dependent variable, the parameters of Model (1) have to be estimated without observing $\mathbf{y}$ on a continuous scale. Instead, only the interval identifier $\mathbf{k}$, now defined as $n \times 1$ column vector, is observed. Open-ended interval bounds $A_0 = -\infty$ and $A_{n_k} = +\infty$ and unequal interval widths are allowed. Since the true distribution of $\mathbf{y}$ is unknown, the aim is to reconstruct the distribution of $\mathbf{y}$ using the known intervals $\mathbf{k}$ and the linear relationship stated in Model (1). As presented in Walter (2019), in order to reconstruct the unknown distribution of $f(\mathbf{y}|\mathbf{X}, \mathbf{Z}, \mathbf{v}, \mathbf{k}, \boldsymbol{\theta})$, where $\boldsymbol{\theta} = (\boldsymbol{\beta}, \mathbf{R}, \mathbf{G})$, the Bayes theorem (Bayes, 1763) is

applied. Hence,

$$f\left(\mathbf{y}|\mathbf{X},\mathbf{Z},\mathbf{v},\mathbf{k},\boldsymbol{\theta}\right) \propto f\left(\mathbf{k}|\mathbf{y},\mathbf{X},\mathbf{Z},\mathbf{v},\boldsymbol{\theta}\right) f\left(\mathbf{y}|\mathbf{X},\mathbf{Z},\mathbf{v},\boldsymbol{\theta}\right),$$

with $f\left(\mathbf{k}|\mathbf{y},\mathbf{X},\mathbf{Z},\mathbf{v},\boldsymbol{\theta}\right) = f\left(\mathbf{k}|\mathbf{y}\right)$ because the conditional distribution of the interval identifier $\mathbf{k}$ only depends on $\mathbf{y}$. It is given by $f\left(\mathbf{k}|\mathbf{y}\right) = \mathbf{r}$, with $\mathbf{r}$ being an $n \times 1$ column vector $\mathbf{r} = \left(r_1, r_2, \ldots, r_n\right)^T$, with

$$r_i = \begin{cases} 1 & \text{if } A_{k_i-1} \leq y_i \leq A_{k_i}, \\ 0 & \text{else,} \end{cases}$$

for $i = 1, \ldots, n$ and

$$f\left(\mathbf{y}|\mathbf{X},\mathbf{Z},\mathbf{v},\boldsymbol{\theta}\right) \sim \quad N\left(\mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{v}, \mathbf{R}\right). \tag{2}$$

The relationship in Equation (2) follows from the linear mixed model assumptions (Model (1)). The unknown parameters $\boldsymbol{\theta} = \left(\boldsymbol{\beta}, \mathbf{R}, \mathbf{G}\right)$ are estimated based on pseudo samples $\tilde{\mathbf{y}}$ (since $\mathbf{y}$ is unknown) that are iteratively drawn from $f\left(\mathbf{y}|\mathbf{X},\mathbf{Z},\mathbf{v},\mathbf{k},\boldsymbol{\theta}\right)$. The next subsection states the computational details of the SEM algorithm.

Estimation and computational details

To fit Model (1), the parameter vector $\hat{\boldsymbol{\theta}} = \left(\widehat{\boldsymbol{\beta}}, \widehat{\mathbf{R}}, \widehat{\mathbf{G}}\right)$ is estimated, and pseudo samples of the unknown $\mathbf{y}$ are iteratively generated by the following SEM algorithm. The pseudo samples $\tilde{\mathbf{y}}$ are drawn from the conditional distribution

$$f\left(\mathbf{y}|\mathbf{X},\mathbf{Z},\mathbf{v},\mathbf{k},\boldsymbol{\theta}\right) \propto \mathbf{I}\left(A_{\mathbf{k}-1} \leq \mathbf{y} \leq A_{\mathbf{k}}\right) \times N\left(\mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{v}, \mathbf{R}\right),$$

where $\mathbf{I}\left(\cdot\right)$ denotes the indicator function. Hence, for $\mathbf{y}$ with explanatory variables $\mathbf{X}$, the corresponding $\tilde{\mathbf{y}}$ is drawn from $N\left(\mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{v}, \mathbf{R}\right)$ conditional on the given interval $\left(A_{\mathbf{k}-1} \leq \mathbf{y} \leq A_{\mathbf{k}}\right)$. If $\hat{\boldsymbol{\theta}}$ is estimated, the conditional distribution $f\left(\mathbf{y}|\mathbf{X},\mathbf{Z},\mathbf{v},\mathbf{k},\boldsymbol{\theta}\right)$ follows a two-sided truncated normal distribution. Its probability density function equals

$$\hat{f}\left(\mathbf{y}|\mathbf{X},\mathbf{Z},\hat{\mathbf{v}},\mathbf{k},\hat{\boldsymbol{\theta}}\right) = \frac{\phi\left(\frac{\mathbf{y}-\hat{\boldsymbol{\mu}}}{\hat{\sigma}_e}\right)}{\hat{\sigma}_e\left(\Phi\left(\frac{A_{\mathbf{k}}-\hat{\boldsymbol{\mu}}}{\hat{\sigma}_e}\right) - \Phi\left(\frac{A_{\mathbf{k}-1}-\hat{\boldsymbol{\mu}}}{\hat{\sigma}_e}\right)\right)}, \tag{3}$$

with $\hat{\boldsymbol{\mu}} = \mathbf{X}\widehat{\boldsymbol{\beta}} + \mathbf{Z}\hat{\mathbf{v}}$. $\phi\left(\cdot\right)$ denotes the probability density function of the standard normal distribution, and $\Phi(\cdot)$ denotes its cumulative distribution function. From its definition, it follows that $\Phi\left(\frac{A_{\mathbf{k}}-\hat{\boldsymbol{\mu}}}{\hat{\sigma}_e}\right) = 1$ if $A_{\mathbf{k}} = +\infty$, and $\Phi\left(\frac{A_{\mathbf{k}-1}-\hat{\boldsymbol{\mu}}}{\hat{\sigma}_e}\right) = 0$ if $A_{\mathbf{k}-1} = -\infty$. The steps of the SEM algorithm as described in Walter (2019) are:

1. Estimate $\hat{\boldsymbol{\theta}} = \left(\widehat{\boldsymbol{\beta}}, \widehat{\mathbf{R}}, \widehat{\mathbf{G}}\right)$ from Model (1) using the midpoints of the intervals as substitutes for the unknown $\mathbf{y}$. The parameters are estimated by restricted maximum likelihood theory (REML) (Thompson, 1962).
2. **Stochastic step:** For $i = 1, \ldots, n$, draw randomly from $N\left(\mathbf{X}\widehat{\boldsymbol{\beta}} + \mathbf{Z}\hat{\mathbf{v}}, \widehat{\mathbf{R}}\right)$ within the given interval $\left(A_{\mathbf{k}-1} \leq \mathbf{y} \leq A_{\mathbf{k}}\right)$ (the two-sided truncated normal distribution given in Equation (3)) obtaining $\left(\tilde{\mathbf{y}}, \mathbf{X}, \mathbf{Z}\right)$. The drawn pseudo $\tilde{\mathbf{y}}$ are used as replacements for the unobserved $\mathbf{y}$.
3. **Maximization step:** Re-estimate the parameter vector $\hat{\boldsymbol{\theta}}$ from Model (1) by using the pseudo samples $\left(\tilde{\mathbf{y}}, \mathbf{X}, \mathbf{Z}\right)$ from Step 2. Again, parameter estimation is carried out by REML.
4. Iterate Steps 2-3 $B^{(SEM)} + M^{(SEM)}$ times, with $B^{(SEM)}$ burn-in iterations and $M^{(SEM)}$ additional iterations.
5. Discard the burn-in iterations $B^{(SEM)}$ and estimate $\hat{\boldsymbol{\theta}}$ by averaging the obtained $M^{(SEM)}$ estimates.

If open-ended intervals $A_0 = -\infty$ and $A_{n_k} = +\infty$ are present, the midpoints M_1 and M_{n_k} of these intervals in iteration Step 1 are computed as follows:

$$M_1 = \left(A_1 - \overline{D}\right)/2,$$
$$M_{n_k} = \left(A_{n_k-1} + \overline{D}\right)/2,$$

where

$$\overline{D} = \frac{1}{(n_k - 2)} \sum_{k=2}^{n_k-1} |A_{k-1} - A_k|.$$

These midpoints serve as proxies for the unknown interval midpoints in Step 1 of the algorithm. The SEM algorithm for the linear regression model is obtained by simplifying the conditional distribution $f(\mathbf{y}|\mathbf{X},\mathbf{Z},\mathbf{v},\boldsymbol{\theta}) \sim N(\mathbf{X}\boldsymbol{\beta}+\mathbf{Z}\mathbf{v},\mathbf{R})$ to $f(\mathbf{y}|\mathbf{X},\boldsymbol{\beta},\sigma_e^2) \sim N(\mathbf{X}\boldsymbol{\beta},\sigma_e^2)$ according to the model assumptions of a linear regression model. In the SEM algorithm for linear models, it is then drawn from $N(\mathbf{X}\boldsymbol{\beta},\sigma_e^2)$ within the given interval.

The standard errors of the regression parameters are estimated using bootstrap methods. For the linear regression model, a non parametric bootstrap (Efron and Stein, 1981; Efron, 1982; Efron and Tibshirani, 1986, 1993) and for the linear mixed regression model, a parametric bootstrap (Wang et al., 2006; Thai et al., 2013) is used to estimate the standard errors. The non parametric, as well as the parametric bootstrap, are further developed to account for the additional uncertainty that is due to the interval-censored dependent variable. Both newly proposed bootstraps are available in the **smicd** package, and the theory is explained in (Walter, 2019).

To assure that the model assumptions are fulfilled, the logarithmic and the Box-Cox transformations are incorporated into the function `semLm()` and `semLme()`.

Core functionality: Regression analysis

The introduced SEM algorithm is implemented in the functions described in Table 4. The arguments and default settings of the estimation functions `semLm()` and `semLme()` are summarized in Table 5. Both functions return an S3 object of class `"sem"`, `"lm"` or `"sem"`, `"lme"`. A detailed explanation of all the components of these objects can be found in the **smicd** package documentation. The generic functions `plot()`, `print()`, and `summary()` can be applied to objects of class `"sem"`,`"lm"` and `"sem"`,`"lme"` in order to summarize the main estimation results. In the next section, the functionality of `semLm()` and `semLme()` is demonstrated based on an illustrative example.

Table 4: Implemented functions for the estimation of linear and linear mixed regression models.

Function Name	Description
`semLm()`	Estimates linear regression models with an interval-censored dependent variable
`semLme()`	Estimates linear mixed regression models with an interval-censored dependent variable
`plot()`	Plots convergence of the estimated parameters and estimated density of the pseudo $\tilde{\mathbf{y}}$ from the last iteration step
`print()`	Prints basic information of the estimated linear and linear mixed regression models
`summary()`	Summary of the estimated linear and linear mixed regression models

Table 5: Arguments of functions `semLm()` and `semLme()`.

Argument	Description	Default
`formula`	A two-sided linear formula object	
`data`	A data frame containing the variables of the model	
`classes`	Numeric vector of interval bounds	
`burnin`	Burn-in iterations	`40`
`samples`	Additional iterations	`200`
`trafo`	Transformation of the dependent variable: None, logarithmic, or Box-Cox transformation	`"None"`
`adjust`	Extends the number of iterations for the estimation of the Box-Cox transformation parameter: $(burnin + samples) * adjust$	`2`
`bootstrap.se`	If TRUE, standard errors and confidence intervals of the regression parameters are estimated	`FALSE`
`b`	Number of bootstraps for the estimation of the standard errors	`100`

Example: Regression analysis

To demonstrate the functions `semLm()` and `semLme()`, the famous London school data set that is analyzed in Goldstein et al. (1993) is used. The data set contains the examination results of 4,059 students from 65 schools in six Inner London Education Authorities. The data set is available in the R package **mlmRev** (Bates et al., 2020a) and also included in the package **smicd**. The variables used in the following example are: general certificate of secondary examination scores (`examsc`), the standardized London reading test scores at the age of 11 years (`standLRT`), the sex of the student (`sex`), and the school identifier (`school`). In the original data set, the variable `examsc` is measured on a continuous scale. In order to demonstrate the functionality of the functions `semLm()` and `semLme()`, the variable is arbitrarily censored to nine intervals. As before, the censoring is carried out by the function `cut()`, and the vector of interval bounds is called `intervals`.

```
R> intervals <- c(1, 1.5, 2.5, 3.5, 4.5, 5.5, 6.5, 7.7, 8.5, Inf)
R> Exam$examsc.class <- cut(Exam$examsc, intervals)
```

The newly created interval-censored variable is called `examsc.class`. The distribution is visualized by applying the function `table()`.

```
R> table(Exam$examsc.class)
  (1,1.5] (1.5,2.5] (2.5,3.5] (3.5,4.5] (4.5,5.5] (5.5,6.5]
        1        32       249       937      1606       951
(6.5,7.7] (7.7,8.5] (8.5,Inf]
      267        15         1
```

It can be seen that most examination scores are concentrated in the center intervals. To fit the linear regression model, the function `semLM()` is called.

```
R> LM <- semLm(
+    formula = examsc.class ~ standLRT + sex, data = Exam,
+    classes = intervals,  bootstrap.se = TRUE
+  )
```

The formula argument is assigned the model equation, where `examsc.class` is regressed on `standLRT` and `sex`. The argument `data` is assigned the name of the data set `Exam`, and the vector of interval bounds `intervals` is assigned to the `classes` argument. The arguments `burnin` and `samples` are left as defaults. The specified number of default iterations is sufficiently large for most regression models. However the convergence of the parameters has to be checked by plotting the estimation results with the function `plot()` after the estimation. No transformation is specified for the interval-censored dependent variable and therefore, `trafo` is assigned its default value. The argument `adjust` is only relevant if the Box-Cox transformation `trafo="bc"` is chosen. In this case, the number of iterations for the estimation of the Box-Cox transformation parameter λ can be specified by this argument. The convergence of the transformation parameter λ has to be checked using the function `plot()`. More information on the Box-Cox transformation and on the estimation of the transformation parameter is given in Walter (2019). For the estimation of the standard errors of the regression parameters, the argument `bootstrap.se` is set to `TRUE`. The number of bootstrap samples `b` is 100, its default value, which again is reasonable for most settings. A summary of the estimation results is obtained by the application of the function `summary()`.

```
R> summary(LM)
Call:
semLm(formula = examsc.class ~ standLRT + sex, data = Exam,
     classes = intervals, bootstrap.se = TRUE)

Fixed effects:
             Estimate  Std.Error Lower 95%-level Upper 95%-level
(Intercept) 5.0702791 0.01816102       5.0326739       5.1033905
standLRT    0.5908015 0.01349275       0.5614197       0.6163845
sexM       -0.1715966 0.03093346      -0.2308930      -0.1010877

Multiple R-squared: 0.3501  Adjusted R-squared: 0.3498
Variable examsc.class is divided into 9 intervals.
```

The output shows the function call, the estimated regression coefficients, the bootstrapped standard errors, and the confidence intervals, as well as the R-squared and the adjusted R-squared. Furthermore, the output reminds the user that the dependent variable is censored to nine intervals. All estimates are

interpreted as in a linear regression model with a continuous dependent variable. Hence, if `standLRT` increases by one unit and all other parameters are kept constant, `examsc.class` increases by 0.59 on average. The bootstrapped confidence intervals indicate that all regressors have a significant effect on the dependent variable.

By using the generic function `plot()` on an object of class `"sem"` and `"lm"`, convergence plots of each estimated regression parameter and of the estimated residual variance are obtained. Furthermore, the density of the generated pseudo $\tilde{\mathbf{y}}$ variable from the last iteration step is plotted with a histogram of the observed distribution of the interval-censored variable `examsc.class` in the background.

```
R> plot(LM)
```

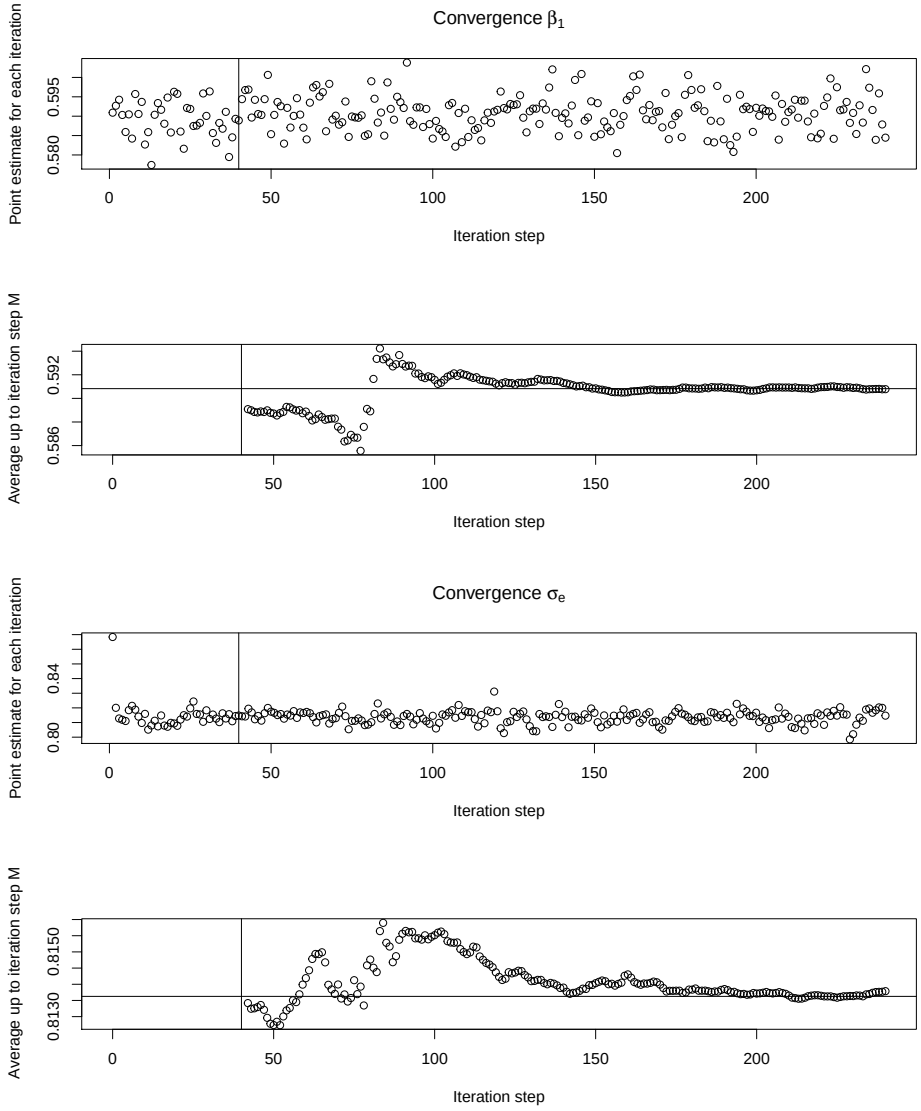

Figure 3: Convergence plots of estimated model parameters (β_1 and σ_e).

In Figure 3, a selection of convergence plots is given, and in Figure 4, the density of the pseudo $\tilde{\mathbf{y}}$ from the last iteration step of the SEM algorithm is plotted. In the convergence plots, the estimated parameter and the average up to iteration step M (excluding B) are plotted for each iteration step of the SEM algorithm. A vertical line indicates the end of the burn-in period (40 iterations). The final parameter estimate marked by the horizontal line is obtained by averaging the $M^{(SEM)}$ additional iterations (200). The selected 240 iterations are enough to obtain reliable estimates in this example because the estimates have converged.

As already mentioned, the **smicd** package also enables the estimation of linear mixed regression models by the function `semLme()`. In the London school data set, students are nested within schools, and therefore, it is necessary to control for the correlation within-schools. In order to do that, the

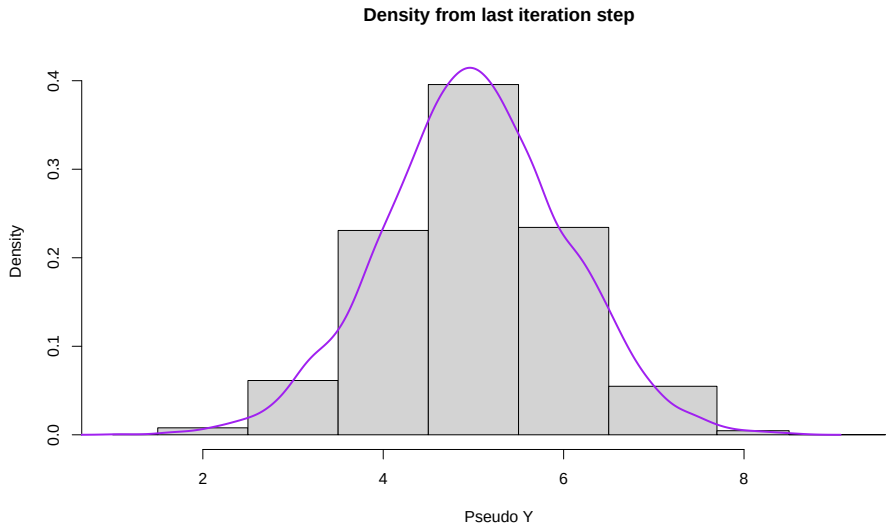

Figure 4: Estimated final density with a histogram of the observed distribution of the data in the background.

variable `school` is specified as a random intercept. If necessary, a random slope parameter could also be included in the model. Again, the variable `sex` is included as an additional regressor. Hence, the `formula` argument is assigned the following model equation `examsc.class` $\sim$ `standLRT + sex + (1|school)`. So far, the function `semLme()` enables the estimation of linear mixed models with a maximum of one random slope and one random intercept parameter. Regarding all other arguments, the same specifications as before are made.

```
R> LME <- semLme(
+    formula = examsc.class ~ standLRT + sex + (1|school),
+    data = Exam, classes = intervals, bootstrap.se = TRUE
+  )
```

By using the generic function `summary()`, the estimation results are printed. In addition to the fixed effects, the estimated random effects are obtained as in the **lme4** and **nlme** packages. Since the R-squared and the adjusted R-squared are not defined for mixed models, the `summary()` function prints the marginal R-squared and conditional R-squared (Nakagawa and Schielzeth, 2013; Johnson, 2014).

```
> summary(LME)
Call:
semLme(formula = examsc.class ~ standLRT + sex + (1 | school),
    data = Exam, classes = intervals, bootstrap.se = TRUE)

Random effects:
    Groups        Name   Variance  Std.Dev.
    school (Intercept) 0.08755431 0.2958958
  Residual             0.58417586 0.7643140

Fixed effects:
             Estimate    Std.Error Lower 95%-level Upper 95%-level
(Intercept) 5.0777581 0.0005188930       5.0769749       5.0791095
standLRT    0.5605049 0.0003665976       0.5599456       0.5613711
sexM       -0.1711065 0.0008159909      -0.1724193      -0.1692369

Marginal R-squared: 0.324 Conditional R-squared: 0.4121
Variable examsc.class is divided into 9 intervals.
```

Again, interpretation is the same as in linear mixed models with a continuous dependent variable. By applying the generic function `plot()` to a `"sem" "lme"` object, the same plots as for the linear regression model are plotted.

Discussion and outlook

Asking for interval-censored data can lead to lower item non-response rates and increased data quality. While item non-response is potentially avoided, applying traditional statistical methods becomes infeasible because the true distribution of the data within each interval is unknown. The functions of the **smicd** package enable researchers to easily analyze this kind of data. The paper briefly introduces the new statistical methodology and presents, in detail, the core functions of the package:

- `kdeAlgo()` for the direct estimation of any statistical indicator,
- `semLm()` to estimate linear models with an interval-censored dependent variable,
- `semLme()` to estimate linear mixed models with an interval-censored dependent variable.

The functions are applied in order to estimate statistical indicators from interval-censored synthetic EU-SILC income data and to analyze interval-censored examination scores of students from London with linear and linear mixed regression models.

Further developments of the **smicd** package will include the possibility to estimate the bootstrapped standard errors in parallel computing environments. Additionally, it is planned to allow for the use of survey weights in the linear (mixed) regression models.

Bibliography

A. Agresti. *Analysis of Ordinal Categorical Data*. Wiley, New Jersey, 2010. URL https://doi.org/10.1002/9780470594001. [p397]

A. Alfons and M. Templ. Estimation of social exclusion indicators from complex surveys: The R package laeken. *Journal of Statistical Software*, 54(15):1–25, 2013. URL http://www.jstatsoft.org/v54/i15/. [p399]

A. Alfons, J. Holzer, and M. Templ. *laeken: Estimation of Indicators on Social Exclusion and Poverty*, 2020. URL https://CRAN.R-project.org/package=laeken. R package version 0.5.1. [p399]

Australian Bureau of Statistics. Census household form, 2011. URL https://unstats.un.org/unsd/demographic/sources/census/quest/AUS2011en.pdf. Accessed: 2020-07-18. [p396]

R. Bandourian, J. McDonald, and R. S. Turley. A comparison of parametric models of income distribution across countries and over time. Technical report, Luxembourg Income Study, 2002. [p396]

D. Bates, M. Mächler, B. Bolker, and S. Walker. Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48, 2015. URL https://doi.org/10.18637/jss.v067.i01. [p397]

D. Bates, M. Maechler, and B. Bolker. *mlmRev: Examples from Multilevel Modelling Software Review*, 2020a. URL https://CRAN.R-project.org/package=mlmRev. R package version 1.0-8. [p406]

D. Bates, M. Maechler, B. Bolker, and S. Walker. *lme4: Linear Mixed-Effects Models using 'Eigen' and S4*, 2020b. URL https://CRAN.R-project.org/package=lme4. R package version 1.1-23. [p397]

T. Bayes. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S. *Philosophical Transactions*, 53:370–418, 1763. URL https://doi.org/10.1098/rstl.1763.0053. [p398, 403]

Y. G. Berger and E. L. Escobar. Variance estimation of imputed estimators of change for repeated rotating surveys. *International Statistical Review*, 85(3):421–438, 2016. URL https://doi.org/10.1111/insr.12197. [p396]

G. E. P. Box and D. R. Cox. An analysis of transformations. *Journal of the Royal Statistical Society: Series B*, 26(2):211–252, 1964. URL http://www.jstor.org/stable/2984418. [p397]

T. A. Cameron. The impact of grouping coarseness in alternative grouped-data regression models. *Journal of Econometrics*, 35(1):37–57, 1987. URL https://doi.org/10.1016/0304-4076(87)90080-7. [p397]

R. H. B. Christensen. *ordinal: Regression Models for Ordinal Data*, 2019. URL https://CRAN.R-project.org/package=ordinal. R package version 2019.12-10. [p397]

C. Dagum. *A New Model of Personal Income Distribution: Specification and Estimation*, pages 3–25. Springer New York, New York, NY, 2008. URL https://doi.org/10.1007/978-0-387-72796-7_1. [p396]

M. L. Delignette-Muller and C. Dutang. fitdistrplus: An R package for fitting distributions. *Journal of Statistical Software*, 64(4):1–34, 2015. URL http://www.jstatsoft.org/v64/i04/. [p396]

M.-L. Delignette-Muller, C. Dutang, and A. Siberchicot. *fitdistrplus: Help to Fit of a Parametric Distribution to Non-Censored or Censored Data*, 2020. URL https://CRAN.R-project.org/package=fitdistrplus. R package version 1.1-1. [p396]

Departamento Administrativo Nacional De Estadística. Censo general 2005, 2005. URL https://www.dane.gov.co/files/censos/libroCenso2005nacional.pdf?&. Accessed: 2020-07-18. [p396]

C. Dutang, V. Goulet, and M. Pigeon. actuar: An r package for actuarial science. *Journal of Statistical Software*, 25(7):38, 2008. URL http://www.jstatsoft.org/v25/i07. [p396]

B. Efron. Bootstrap methods: another look at the jackknife. *The Annals of Statistics*, 7(1):1–26, 1979. URL https://www.jstor.org/stable/2958830. [p399]

B. Efron. *The Jackknife, the Bootstrap and Other Resampling Plans*. Stanford University, Philadelphia, 1982. URL https://doi.org/10.1137/1.9781611970319. [p405]

B. Efron and C. Stein. The jackknife estimate of variance. *The Annals of Statistics*, 9(3):586–596, 1981. URL https://doi.org/10.1214/aos/1176345462. [p405]

B. Efron and R. Tibshirani. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. *Statistical Science*, 1(1):54–75, 1986. URL https://doi.org/10.1214/ss/1177013815. [p405]

B. Efron and R. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall, New York, 1993. [p405]

Eurostat. Statistics explained: at-risk-of-poverty rate, 2014. URL http://ec.europa.eu/eurostat/statistics-explained/index.php/Glossary:At-risk-of-poverty_rate. Accessed: 2020-07-11. [p401]

L. Fahrmeir, R. Künstler, I. Pigeot, and G. Tutz. *Statistik - Der Weg zur Datenanalyse*. Springer, Berlin, 2016. URL https://doi.org/10.1007/978-3-662-50372-0. [p396]

J. Foster, J. Greer, and E. Thorbecke. A class of decomposable poverty measures. *Econometrica*, 52(3): 761–766, 1984. URL https://doi.org/10.2307/1913475. [p401]

J. G. Fryer and R. J. Pethybridge. Maximum likelihood estimation of a linear regression function with grouped data. *Journal of the Royal Statistical Society: Series C*, 21(2):142–154, 1972. URL https://doi.org/10.2307/2346486. [p397]

C. Gini. *Variabilità e mutabilità: contributo allo studio delle distribuzioni e delle relazioni statistiche*. Studi economico-giuridici pubblicati per cura della facoltà di Giurisprudenza della R. Università di Cagliari. Tipogr. di P. Cuppini, Bologna, 1912. URL https://books.google.de/books?id=fqjaBPMxB9kC. [p401]

H. Goldstein. *Multilevel Statistical Models*. Wiley, New York, 2010. URL https://doi.org/10.1002/9780470973394. [p403]

H. Goldstein, J. Rasbash, M. Yang, G. Woodhouse, H. Pan, D. Nuttall, and S. Thomas. A multilevel analysis of school examination results. *Oxford Review of Education*, 19(4):425–433, 1993. URL https://doi.org/10.1080/0305498930190401. [p406]

V. Goulet, C. Dutang, M. Pigeon, J. A. Ryan, R. Gentleman, R. Ihaka, R Core Team, and R Foundation. *actuar: Actuarial Functions and Heavy Tailed Distributions*, 2020. URL https://CRAN.R-project.org/package=actuar. R package version 3.0-0. [p396]

M. Groß, U. Rendtel, T. Schmid, S. Schmon, and N. Tzavidis. Estimating the density of ethnic minorities and aged people in Berlin: multivariate kernel density estimation applied to sensitive georeferenced administrative data protected via measurement error. *Journal of the Royal Statistical Society: Series A*, 180(1):161–183, 2017. URL https://doi.org/10.1111/rssa.12179. [p398, 401]

M. J. Gurka, L. J. Edwards, K. E. Muller, and L. Kupper. Extending the Box-Cox transformation to the linear mixed model. *Journal of the Royal Statistical Society: Series A*, 169(2):273–288, 2006. URL https://doi.org/10.1111/j.1467-985X.2005.00391.x. [p397]

A. Hagenaars and K. D. Vos. The definition and measurement of poverty. *Journal of Human Resources*, 23(2):211–221, 1988. URL https://doi.org/10.2307/145776. [p396]

P. Johnson. Extension of Nakagawa & Schielzeth's R^2_{GLMM} to random slopes models. *Methods in Ecology and Evolution*, 5(9):944–946, 2014. URL https://doi.org/10.1111/2041-210X.12225. [p408]

M. C. Jones, J. S. Marron, and S. J. Sheather. A brief survey of bandwidth selection for density estimation. *Journal of the American Statistical Association*, 91(433):401–407, 1996. URL https://doi.org/10.2307/2291420. [p398]

M. N. Laird and J. H. Ware. Random-effects models for longitudinal data. *Biometrics*, 38(4):963–74, 1983. URL https://doi.org/10.2307/2529876. [p403]

S. Lenau and R. Münnich. Estimating income poverty and inequality from income classes. Technical report, InGRID Integrating Expertise in Inclusive Growth: Case Studies, 2016. [p396, 397]

C. R. Loader. Bandwidth selection: classical or plug-in? *Annals of Statistics*, 27(2):415–438, 1999. URL https://doi.org/10.1214/aos/1018031201. [p398]

P. McCullagh. Regression models for ordinal data. *Journal of the Royal Statistical Society: Series B*, 42(2): 109–142, 1980. URL http://www.jstor.org/stable/2984952. [p397]

C. E. McCulloch, S. R. Searle, and J. M. Neuhaus. *Generalized, Linear, and Mixed Models*. Wiley, New Jersey, 2008. [p403]

J. B. McDonald. Some generalized functions for the size distribution of income. *Econometrica*, 52(3): 647–663, 1984. URL https://doi.org/10.2307/1913469. [p396]

J. C. Moore and E. J. Welniak. Income measurement error in surveys: a review. *Journal of Official Statistics*, 16(4):331–361, 2000. [p396]

S. Nakagawa and H. Schielzeth. A general and simple method for obtaining R^2 from generalized linear mixed-effects models. *Methods in Ecology and Evolution*, 4(2):133–142, 2013. URL https://doi.org/10.1111/j.2041-210x.2012.00261.x. [p408]

E. Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962. URL https://doi.org/10.1214/aoms/1177704472. [p398]

J. Pinheiro, D. Bates, and R-core. *nlme: Linear and Nonlinear Mixed Effects Models*, 2020. URL https://CRAN.R-project.org/package=nlme. R package version 3.1-148. [p397]

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020. URL https://www.R-project.org/. [p397]

B. Ripley. *MASS: Support Functions and Datasets for Venables and Ripley's MASS*, 2019. URL https://CRAN.R-project.org/package=MASS. R package version 7.3-51.5. [p397]

M. Rosenblatt. Remarks on some nonparametric estimates of a density function. *The Annals of Mathematical Statistics*, 27(3):832–837, 1956. URL https://www.jstor.org/stable/2237390. [p398]

R. N. Rosett and F. D. Nelson. Estimation of the two-limit probit regression model. *Econometrica*, 43(1): 141–146, 1975. URL https://doi.org/10.2307/1913419. [p397]

S. R. Searle, G. Casella, and C. E. McCulloch. *Variance Components*. Wiley, New York, 1992. [p403]

J. Shao and D. Tu. *The Jackknife and Bootstrap*. Springer, New York, 1995. URL https://doi.org/10.1007/978-1-4612-0795-5. [p399]

T. Snijders and R. Bosker. *Multilevel Analysis: An Introduction to Basic and Advanced Multilevel Modeling*. Sage, London, 2011. [p403]

Statistisches Bundesamt. Codebook microsensus 2014, 2014. URL https://www.forschungsdatenzentrum.de/sites/default/files/mz_2014_suf_dhb.pdf. Accessed: 2020-07-18. [p399]

Statistisches Bundesamt. Data supply: microcensus, 2016. URL https://www.forschungsdatenzentrum.de/de/haushalte/mikrozensus. Accessed: 2020-07-18. [p399]

Statistisches Bundesamt. Datenhandbuch zum Mikrozensus Scientific-Use-File 2012. http://www.forschungsdatenzentrum.de/bestand/mikrozensus/suf/2012/fdz_mz_suf_2012_schluesselverzeichnis.pdf, 2017. Accessed: 2020-07-18. [p396]

M. Stewart. On least square estimation when the dependent varaible is grouped. *The Review of Economic Studies*, 50(4):737–753, 1983. URL https://doi.org/10.2307/2297773. [p397]

H. Thai, F. Mentre, N. Holford, C. Veyrat-Follet, and E. Comets. A comparison of bootstrap approaches for estimating uncertainty of parameters in linear mixed-effects models. *Pharmaceutical Statistics*, 12 (3):129–140, 2013. URL https://doi.org/10.1002/pst.1561. [p405]

T. M. Therneau. *survival: Survival Analysis*, 2020. URL https://CRAN.R-project.org/package=survival. R package version 3.2-3. [p397]

T. M. Therneau and P. M. Grambsch. *Modeling Survival Data: Extending the Cox Model*. Springer, New York, 2000. [p397]

J. W. A. Thompson. The problem of negative estimates of variance components. *Annals of Mathematical Statistics*, 33(1):273–289, 1962. URL https://doi.org/10.1214/aoms/1177704731. [p404]

M. L. Thompson and K. Nelson. Linear regression with type I interval- and left-censored response data. *Environmental and Ecological Statistics*, 10(2):221–230, 2003. URL https://doi.org/10.1023/A:1023630425376. [p397]

J. Tobin. Estimation of relationships for limited dependent variables. *Econometrica*, 26(1):24–36, 1958. URL https://doi.org/10.2307/1907382. [p397]

W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, New York, 2002. URL https://doi.org/10.1007/978-0-387-21706-2. [p397]

P. Walter. *A Selection of Statistical Methods for Interval-Censored Data with Applications to the German Microcensus*. PhD thesis, Freie Universität Berlin, 2019. URL https://doi.org/10.17169/refubium-1621. [p397, 398, 399, 401, 403, 404, 405, 406]

P. Walter. *smicd: Statistical Methods for Interval Censored Data*, 2020. URL https://CRAN.R-project.org/package=smicd. R package version 1.1.1. [p397]

B. Wang and M. Wertelecki. Density estimation for data with rounding errors. *Computational Statistics & Data Analysis*, 65:4–12, 2013. URL https://doi.org/10.1016/j.csda.2012.02.016. [p398]

J. Wang, J. R. Carpenter, and M. A. Kepler. Using SAS to conduct nonparametric residual bootstrap multilevel modeling with a small number of groups. *Computer Methods and Programs in Biomedicine*, 82(2):130–143, 2006. URL https://doi.org/10.1016/j.cmpb.2006.02.006. [p405]

A. Z. Zambom and R. Dias. A review of kernel density estimation with applications to econometrics. *International Econometric Review*, 5(1):20–42, 2012. URL https://EconPapers.repec.org/RePEc:erh:journl:v:5:y:2013:i:1:p:20-42. [p398]

Paul Walter
Freie Universität Berlin
Garystraße 21, 14195 Berlin
Germany
paul.walter@fu-berlin.de