

# Confidence intervals for rank statistics: percentile slopes, differences and ratios

Roger Newson  
Imperial College London, UK  
r.newson@imperial.ac.uk

**Abstract.** A program `censlope` is presented for calculating confidence intervals for generalized Theil–Sen median (and other percentile) slopes (and per–unit ratios) of a variable  $Y$  with respect to a variable  $X$ . The confidence intervals are robust to the possibility that the conditional population distributions of  $Y$ , given different values of  $X$ , differ in ways other than location, such as having unequal variances. The program uses the program `somersd`, and is part of the `somersd` package. It can therefore estimate confounder–adjusted percentile slopes, limited to comparisons within strata defined by values of confounders, or by values of a propensity score representing multiple confounders. Iterative numerical methods have been implemented in the Mata language, enabling efficient calculation of percentile slopes and their confidence limits in large samples. Example analyses are given from the `auto` data and from the Avon Longitudinal Study of Pregnancy and Childhood (ALSPAC).

**Keywords:** `st0001`, ALSPAC, robust, confidence interval, rank, nonparametric, median, percentile, slope, difference, ratio, Kendall’s tau, Somers’  $D$ , Theil–Sen, Hodges–Lehmann, confounder–adjusted, propensity score.

## 1 Introduction

The Theil–Sen median slope is a rank–based parameter, defined in terms of Kendall’s tau, but expressed in  $Y$ –axis units per  $X$ –axis unit, and interpreted as a “typical” difference in a  $Y$ –variable associated with a unit difference in an  $X$ –variable. It is therefore useful to know, if we want to use rank methods to make monetary or other practical decisions. It was introduced by Theil (1950) and developed by Sen (1968), who derived a confidence interval formula. If the  $X$ –variable is binary, then the Theil–Sen median slope is known as the Hodges–Lehmann median difference, and is expressed in  $Y$ –axis units. This median difference was introduced by Hodges and Lehmann (1963) and developed by Lehmann (1963), who derived a confidence interval formula which is a special case of the one in Sen (1968). The median difference was popularized by Conover (1980), Campbell and Gardner (1988) and Altman et al. (2000), and implemented in Stata by Wang (1999) and in Patrick Royston’s SSC package `cid`. A good general introduction to confidence interval formulas for median slopes and differences is given in Sprent and Smeeton (2001).

Most existing confidence interval formulas for median slopes and differences assume that, if  $\beta$  is the median slope, then the variable  $Y - \beta X$  is statistically independent of  $X$ . This in turn implies that the conditional distributions of  $Y$ , given different values

of  $X$ , differ only in location, and not in other ways such as unequal variance. These problems are discussed in Wilcox (1998), who describes a possible solution using the percentile bootstrap. Also, the median differences and slopes are usually defined as crude differences and slopes, assumed to apply to the whole population, and not as adjusted differences and slopes, assumed to apply within sub-populations with similar values of stratification or confounding variables. These assumptions may limit the usefulness of these confidence interval formulas.

In a more recent paper (Newson (2002)), it was argued that median differences and slopes belong to a unified family of rank parameters, with a unified system of confidence interval formulas. In this family of parameters, median differences and slopes are naturally defined in terms of Somers'  $D$ , which in turn is naturally defined in terms of Kendall's  $\tau_a$ . This paper introduced the `somersd` package, downloadable from SSC, as a way of calculating *some* of these confidence intervals. At the time, the `somersd` package contained two modules, namely `somersd`, described in Newson (2000a), for calculating confidence intervals for Somers'  $D$  and Kendall's  $\tau_a$ , and `cendif`, described in Newson (2000b), for calculating confidence intervals for Hodges–Lehmann median differences. In 2005, Version 9 of Stata introduced the Mata programming language, which made it possible to update the module `somersd` to estimate many extended versions of Somers'  $D$  and Kendall's  $\tau_a$ , and to do so with increased computational speed. This update was reported in Newson (2006b), which contains the syntax, formulas and methods, and in Newson (2006a), which describes the Mata algorithm used.

The current paper describes a third module `censlope`, which has been added to the `somersd` package in a more recent update, and which estimates generalized Theil–Sen median (and other percentile) slopes, differences and ratios. In particular, these slopes, differences and ratios may be adjusted for confounding variables, allowing the user to use rank methods to answer many questions which could previously only be answered using regression methods. In Section 2, we describe the current version of the program `censlope`. In Section 3, we present in detail, for reference purposes, the methods and formulas used by `censlope`. In Section 4, we demonstrate a range of examples.

## 2 The program `censlope`

### 2.1 Syntax

```
censlope yvarname xvarname [ weight ] [ if exp ] [ in range ] [ ,
    centile(numlist) eform ystargenerate(newvarlist) estaddr somersd_options
    iteration_options ]
```

where *yvarname* and *xvarname* are variable names, *somersd\_options* are any of the options used by `somersd`, and *iteration\_options* are any of the options described in Subsection 2.4.

`fweights`, `iwweights` and `pweights` are allowed; see help for `weight`. They are inter-

preted as for `somersd`.

`bootstrap`, `by`, `jackknife`, and `statsby` are allowed; see help for `prefix`.

## 2.2 Description

`censlope` calculates confidence intervals for generalized Theil–Sen median slopes, and other percentile slopes, of a  $Y$ -variable specified by `yvarname` with respect to an  $X$ -variable specified by `xvarname`. These confidence intervals are robust to the possibility that the population distributions of the  $Y$ -variable, conditional on different values of the  $X$ -variable, are different in ways other than location. This might happen if, for example, the conditional distributions had different variances. For positive-valued  $Y$ -variables, `censlope` can be used to calculate confidence intervals for median per-unit ratios, or other percentile per-unit ratios, associated with a unit increment in the  $X$ -variable. If the  $X$ -variable is binary with values 0 and 1, then the generalized Theil–Sen percentile slopes are the generalized Hodges–Lehmann percentile differences between the group of observations whose  $X$ -value is 1 and the group of observations whose  $X$ -value is 0. `censlope` is part of the `somersd` package, and requires the `somersd` program in order to work. It executes the `somersd` command

```
somersd xvarname yvarname [ weight ] [ if exp ] [ in range ] [ , somersd_options ]
```

and then estimates the percentile slopes. The estimates and confidence limits for the percentile slopes are evaluated using an iterative numerical method, which the user may change from the default, using the `iteration_options`.

## 2.3 Ordinary options

`centile(numlist)` specifies a list of percentile slopes to be reported, and defaults to `centile(50)` (median only) if not specified. Specifying `centile(25 50 75)` will produce the 25th, 50th and 75th percentile differences.

`eform` specifies that exponentiated percentile slopes are to be given. This option is used if `yvarname` specifies the log of a positive-valued variable. In this case, confidence intervals are calculated for percentile ratios or per-unit ratios between values of the original positive variable, instead of for percentile differences or per-unit differences.

`ystargenerate(newvarlist)` specifies a list of variables to be generated, corresponding to the percentile slopes, containing the differences  $Y^*(\beta) = Y - \beta X$ , where  $\beta$  is the percentile slope. The variable names in the `newvarlist` are matched to the list of percentiles specified by the `centiles()` option, sorted in ascending order of percent. If the two lists have different lengths, then `censlope` generates a number `nmin` of new variables equal to the minimum length of the two lists, matching the first `nmin` percentiles with the first `nmin` new variable names. Usually, there is only one percentile slope (the median slope), and one new `ystargenerate()` variable, whose median can be used as the intercept when drawing a straight line through the data points on a scatter plot.

`estaddr` specifies that the results saved in `r()` will also be saved in `e()` (see Subsection 2.5 below). This makes it easier to use `censlope` with `parmbly`, in order to create an output dataset (or `resultsset`) with one observation per by-group and data on confidence intervals for Somers'  $D$  and median slopes. `parmbly` is part of the package `parmest`, downloadable from SSC. An example of the use of the `estaddr` option with `parmbly` appears in the online help for `censlope`.

## 2.4 Iteration options

Table 1: Iteration options used by `censlope`.

<i>options</i>	Description
<code>fromabs(#)</code>	initial estimate for absolute magnitude of slopes
<code>brackets(#)</code>	maximum number of rows for the bracket matrix
<code>technique(<i>algorithm_spec</i>)</code>	iterative numerical solution technique
<code>iterate(#)</code>	perform maximum of # iterations; default is <code>iterate(16000)</code>
<code>tolerance(#)</code>	tolerance for the percentile slopes
<code>log</code>	display an iteration log of the brackets during bracket convergence

where `algorithm_spec` is

```
algorithm [ # [ algorithm [ # ] ] ... ]
```

and `algorithm` is { `bisect` | `regula` | `ridders` }

The `censlope` command calculates estimates and confidence limits for a median or other percentile slope  $\beta$  by solving numerically a scalar equation in  $\beta$ , using an iterative method. The options controlling the exact iterative method will probably not be used very often, because `censlope` is intended to have sensible defaults. Non-technical readers may therefore skip this subsection. However, users who wish to change the default method may do so, using a set of options similar to the maximization options used by Stata maximum-likelihood estimation commands (see [R] `maximize`). These options are listed in Table 1, and are described as follows:

`fromabs(#)` specifies an initial estimate of the typical absolute magnitude of a percentile slope. If `fromabs()` is not specified, then it defaults to the aspect ratio  $(y_{max} - y_{min}) / (x_{max} - x_{min})$  (where  $x_{max}$  and  $x_{min}$  are the maximum and minimum  $X$ -values, and  $y_{max}$  and  $y_{min}$  are the maximum and minimum  $Y$ -values) if that ratio is defined and nonzero, and to 1 otherwise. This magnitude is used in the construction of the bracket matrix. Candidate bracket  $\beta$ -values will have values of zero or of  $\pm fromabs \times 2^K$ , where  $K$  is a nonnegative integer. The bracket matrix is a matrix with 2 columns and 3 or more rows, each row containing a candidate  $\beta$ -value in column 1 and the corresponding  $\zeta^*$ -value in column 2. It is used to find an initial pair of  $\beta$ -values for input into the iterative numerical solution method,

which attempts to find a solution in  $\beta$  between the two initial  $\beta$ -values. The bracket matrix is initialized to have  $\beta$ -values  $-fromabs$ , 0 and  $+fromabs$ , and  $\zeta^*$ -values corresponding to these  $\beta$ -values. If a target  $\zeta$ -value is outside the range of the  $\zeta^*$ -values of the bracket matrix, then the bracket matrix is extended by adding new rows before the first row by successively doubling the  $\beta$ -value in the first row, or by adding new rows after the last row by successively doubling the  $\beta$ -value in the last row, until there is a  $\zeta^*$ -value in the second column on either side of the target  $\zeta$ -value. For an explanation of this terminology, see Section 3.

**brackets**(#) specifies a maximum number of rows for the bracket matrix. The minimum is **brackets**(3). The default is **brackets**(1000).

**technique**(*algorithm\_spec*) specifies an iterative solution method for finding a solution in  $\beta$  to the equation to be solved. The following algorithms are currently implemented in **censlope**:

**technique**(**bisect**) specifies an adapted version of the bisection method for step functions.

**technique**(**regula**) specifies an adapted version of the regula falsi (or false position) method for step functions.

**technique**(**ridders**) specifies an adapted version of the method of Ridders (1979) for step functions.

The default is **technique**(**ridders 5 bisect iterate**), where *iterate* is the value of the **iterate**() option. The bisection method is guaranteed to converge in a number of iterations similar to the binary logarithm of the **tolerance**() option. The regula falsi and Ridders methods are usually faster if the  $\zeta^*$ -function is very nearly continuous, but may sometimes be slower if the  $\zeta^*$ -function is a very discrete step function. All methods are modified versions, for step functions, of the methods of the same names described in Press et al. (1992).

You can switch between algorithms by specifying more than one in the **technique**() option. By default, **censlope** will use an algorithm for five iterations before switching to the next algorithm. To specify a different number of iterations, include the number after the technique in the option. For example, specifying **technique**(**ridders 10 bisect 1000**) requests that **censlope** perform 10 iterations using the Ridders algorithm, perform 1000 iterations using the bisection algorithm, and then switch back to Ridders for 10 iterations, and so on. The process continues until convergence, or until the maximum number of iterations is reached.

**iterate**(#) specifies the maximum number of iterations. When the number of iterations equals **iterate**(), the iterative solution program stops and records failure to converge. If convergence is declared before this threshold is reached, it will stop when convergence is declared. The default value of **iterate**(#) is the current value of **set maxiter**, which is **iterate**(16000) by default.

`tolerance(#)` specifies the tolerance for the percentile differences. When the relative difference between the current  $\beta$ -brackets is less than or equal to `tolerance()`, the `tolerance()` convergence criterion is satisfied. `tolerance(1e-6)` is the default.

`log` specifies that an iteration log showing the progress of the numerical solution method is to be displayed. Note that, if an iteration log is displayed, then there will be 4 separate iteration sequences per percentile, estimating the left estimate, the right estimate, the lower confidence limit, and the upper confidence limit, respectively. For this reason, the default is not to produce an iteration log. However, if `censlope` is expected to be slow (as in the case of very large datasets), then an iteration log can be specified to reassure the user that progress is being made.

## 2.5 Saved results

`censlope` saves the following results in `r()`:

### Scalars

`r(level)` confidence level  
`r(fromabs)` value of the `fromabs()` option  
`r(tolerance)` value of the `tolerance()` option

### Macros

`r(yvar)` name of the Y-variable  
`r(xvar)` name of the X-variable  
`r(eform)` `eform` if specified  
`r(centiles)` list of percents for the percentiles  
`r(technique)` list of techniques from the `technique()` option  
`r(tech_steps)` list of step numbers for the techniques

### Matrices

`r(cimat)` confidence intervals for percentile differences or ratios  
`r(rcmat)` return codes for entries of `r(cimat)`  
`r(bracketmat)` bracket matrix  
`r(techstepmat)` column vector of step numbers for the techniques

The matrix `r(cimat)` has one row per percentile, and columns containing the percents, percentile estimates, lower confidence limits and upper confidence limits, labelled `Percent`, `Pctl_Slope`, `Minimum` and `Maximum` if `eform` is not specified, or `Percent`, `Pctl_Ratio`, `Minimum` and `Maximum` if `eform` is specified. The matrix `r(rcmat)` has the same numbers of rows and columns as `r(cimat)`, with the same labels, and the first column contains the percents, but the other entries contain return codes for the estimation of the corresponding entries of `r(cimat)`. These return codes are equal to 0 if the  $\beta$ -value was estimated successfully, 1 if the corresponding  $\zeta^*$ -value could not be calculated, 2 if the corresponding  $\zeta^*$ -value could not be bracketed, 3 if the  $\beta$ -brackets failed to converge, and 4 if the  $\beta$ -value could not be calculated from the converged  $\beta$ -brackets. The matrix `r(bracketmat)` is the final version of the bracket matrix described in the help for the `fromabs()` and `brackets()` options of `censlope`, and has one row per  $\beta$ -bracket, and two columns, labelled `Beta` and `Zetastar`, containing the

$\beta$ -brackets and the corresponding  $\zeta^*$ -values. The matrix `r(techstepmat)` is a column vector, with one row for each of the techniques listed in the `technique()` option, with a row label equal to the name of the technique and a value equal to the number of steps for that technique. The `fromabs()`, `brackets()`, `tolerance()` and `technique()` options are described in Subsection 2.4 above.

`censlope` also saves in `e()` a full set of estimation results for the `somersd` command

```
somersd xvarname yvarname [ weight ] [ if exp ] [ in range ] [ , somersd_options ]
```

as described in Subsection 2.2 above. If `estaddr` is specified, then this set of estimation results is expanded by adding a set of `e()` results with the same names and contents as the `r()` results. This allows the user to pass a `censlope` command to `parmby`, producing an output dataset (or resultset) with one observation per by-group and data on confidence intervals for Somers'  $D$  and for the median slope.

### 3 Methods and formulas

This section is intended mainly as a reference for the extensive family of methods and formulas used by the `censlope` program. Less technically-minded readers may skip or skim through this section and progress to the Examples.

The Theil–Sen median slope was introduced by Theil (1950) and developed further by Sen (1968). If the  $X$ -variable is binary with values 0 and 1, then the Theil–Sen slope is the Hodges–Lehmann median difference of Hodges and Lehmann (1963) and Lehmann (1963). The methods used by `censlope` are a generalization of the methods of Theil and Sen. They include, as a special case, the methods used by `cendif` (Newson (2000b)), which calculates confidence intervals for generalized Hodges–Lehmann median differences, and is also part of the `somersd` package. However, note that `cendif` (like `ttest`) estimates the median difference between  $Y$ -values associated with the smaller  $X$ -value and  $Y$ -values associated with the larger  $X$ -value, whereas `censlope` (like `regress`), given a binary  $X$ -variable with values 0 and 1, estimates the median difference between  $Y$ -values associated with the larger  $X$ -value and  $Y$ -values associated with the smaller  $X$ -value.

Percentile slopes are defined in terms of the parameters Somers'  $D$  (Somers (1962)) and Kendall's  $\tau_a$  (Kendall and Gibbons (1990)). A discussion of the connections between these parameters appears in Newson (2002). For the purposes of `censlope`, we will define Somers'  $D$  and Kendall's  $\tau_a$  in the very general sense used in Newson (2006b). Given two random variables  $U$  and  $V$ , we denote by  $\tau(U, V)$  the Kendall's  $\tau_a$  of  $U$  and  $V$ , and denote by  $D(U|V)$  the Somers'  $D$  of  $U$  with respect to  $V$ . Briefly, if two  $(U, V)$ -pairs  $(U_i, V_i)$  and  $(U_j, V_j)$  are sampled from some population of such pairs using some sampling scheme, then  $\tau(U, V)$  is the difference between the probability that the two  $(U, V)$ -pairs are concordant (meaning that the higher of the two  $U$ -values is paired with the higher of the two  $V$ -values) and the probability that the two  $(U, V)$ -pairs are discordant (meaning that the higher of the two  $U$ -values is paired with the lower of the two  $V$ -values). We define  $D(U|V)$  as the difference between the corresponding

*conditional* probabilities, given that the two  $V$ -values are strictly ordered (meaning that one  $V$ -value is known to be higher than the other  $V$ -value). Note that both  $\tau(U, V)$  and  $D(U|V)$  are differences between probabilities, and therefore both may have values ranging from -1 (for a “perfect negative association”) to +1 (for a “perfect positive association”), but  $\tau(U, V)$  is always symmetric in  $U$  and  $V$ , whereas  $D(U|V)$  is not. We will use the notation  $\theta(U, V)$  to stand for the value of either  $\tau(U, V)$  or  $D(U|V)$  in the population, and denote the corresponding sample value as  $\hat{\theta}(U, V)$ . The `somersd` package allows us to choose between Somers  $D$  and Kendall’s  $\tau_a$  using the `taua` option, and also provides other options, to specify a version of either parameter corresponding to a specific sampling scheme.

For an outcome variable  $Y$ , a predictor variable  $X$  and a proportion  $q$  such that  $0 \leq q \leq 1$ , a 100 $q$ th percentile slope of  $Y$  with respect to  $X$  is defined as a value  $\beta$  satisfying

$$\theta(Y - \beta X, X) = 1 - 2q \quad (1)$$

If  $q = 0.5$ , then  $1 - 2q = 0$ , and a solution in  $\beta$  to (1) is known as a Theil–Sen median slope, as defined in Theil (1950) and Sen (1968). Note that there is not always a unique solution to (1) in  $\beta$ . If the joint population distribution of  $Y$  and  $X$  is discrete (as are all population distributions sampled by applied statisticians in the real world), then  $\theta(Y - \beta X, X)$  will be a monotonically non-increasing step function of  $\beta$ , and there may be no exact solution, or an interval of exact solutions. However, the confidence intervals derived here will contain any solution with the specified confidence level, *if* a solution exists.

If  $\theta(\cdot, \cdot)$  stands for Somers’  $D$  rather than Kendall’s  $\tau_a$ , then the value of the parameter  $\theta(Y - \beta X, X)$  depends only on the conditional distribution of pairs of bivariate observations  $(X_1, Y_1)$  and  $(X_2, Y_2)$  satisfying  $X_1 < X_2$ . For such pairs of observations, the pairwise slope  $(Y_2 - Y_1)/(X_2 - X_1)$  is always defined. If neither  $X$  nor  $Y$  is subject to left- or right-censorship, then the equality (1) becomes

$$\begin{aligned} 1 - 2q &= D(Y - \beta X | X) \\ &= \Pr(Y_1 - \beta X_1 < Y_2 - \beta X_2) - \Pr(Y_1 - \beta X_1 > Y_2 - \beta X_2) \\ &= \Pr[(Y_2 - Y_1)/(X_2 - X_1) > \beta] - \Pr[(Y_2 - Y_1)/(X_2 - X_1) < \beta] \end{aligned} \quad (2)$$

Therefore, a 0.5th percentile (or median) slope has the expected property that a pairwise slope is equally likely to be above or below it. If in addition the distributions of  $X$  and  $Y$  are limited to finite sets of discrete values, then the distribution of pairwise slopes will be bounded, and a 0th percentile slope will be any number below all possible pairwise slopes, and a 100th percentile slope will be any number above all possible pairwise slopes.

We aim to include a value  $\beta$  in a confidence interval for a 100 $q$ th percentile slope if and only if the sample  $\hat{\theta}(Y - \beta X, X)$  is compatible with a *population*  $\theta(Y - \beta X, X)$  equal to  $1 - 2q$ . The methods of Newson (2006b), used by the program `somersd`, typically use a monotonically-increasing transformation  $\zeta(\cdot)$ , which *may* be Normalizing and/or



variance-stabilizing when applied to  $\hat{\theta}(Y - \beta X, X)$ . We define

$$\zeta^*(\beta) = \zeta[\hat{\theta}(Y - \beta X, X)] \quad (3)$$

Note that  $\zeta^*(\beta)$  is a randomly variable function of  $\beta$ , with a population standard error  $\text{SE}[\zeta^*(\beta)]$ , estimated consistently by a corresponding *sample* standard error  $\widehat{\text{SE}}[\zeta^*(\beta)]$ , whose formula is one of those described in Newson (2006b). We will assume that, if  $\theta(Y - X\beta, X) = 1 - 2q$ , then the pivotal quantity

$$[\zeta^*(\beta) - \zeta(1 - 2q)] / \text{SE}[\zeta^*(\beta)] \quad (4)$$

has a standard Normal distribution. In general, the sample  $\zeta^*(\beta)$  is a monotonically non-increasing step function of  $\beta$ , bounded above by  $\zeta(-1)$  and below by  $\zeta(1)$ , either of which bounds may be infinite, depending on the choice of transformation  $\zeta(\cdot)$ .

Figure 1 illustrates an example of a function  $\zeta^*(\beta)$  from the `auto` data. Here, the observations are car models, the  $Y$ -variable is `trunk` (trunk space in cubic feet), the  $X$ -variable is `foreign` (a binary variable indicating non-US origin), the transformation is the hyperbolic arctangent or Fisher's  $z$  (as recommended by Edwardes (1995)), and a slope  $\beta$  is a difference (expressed in cubic feet) between cars made by non-US and US companies. The function  $\zeta^*(\beta)$  is plotted against the differences  $\beta$  over the range of differences for which the absolute value of  $\zeta^*(\beta)$  is finite. (As there are no differences between non-US and US cars above 9 cubic feet or below -18 cubic feet, the value of  $\zeta^*(\beta)$  is  $-\infty$  for  $\beta > 9$  and  $+\infty$  for  $\beta < -18$ .) This plot was made using the program `condif`, which is restricted to binary  $X$ -variables, and calculates the full set of differences in the  $Y$ -variable between observations in the two groups. The square data points give values of  $\zeta^*(\beta)$  for differences  $\beta$  actually observed in the `auto` data, and the solid line gives values of  $\zeta^*(\beta)$  for values of  $\beta$  between these observed values. Note that the sample  $\zeta^*(\beta)$  is a monotonically non-increasing step function of  $\beta$ , which is discontinuous at the observed differences and constant within the open intervals between consecutive observed differences. This implies that a unique exact solution for (1) does not usually exist, as there is usually either no exact solution or an interval of exact solutions between two consecutive observed differences. In a finite sample, this will be true for observed slopes in general, whether or not the  $X$ -variable is binary.

If we knew the value of  $\text{SE}[\hat{\zeta}^*(\beta)]$ , then a  $100(1 - \alpha)\%$  confidence interval for a  $100q$ th percentile difference might be the interval of values  $\beta$  for which

$$\zeta(1 - 2q) - z_\alpha \text{SE}[\hat{\zeta}^*(\beta)] \leq \zeta^*(\beta) \leq \zeta(1 - 2q) + z_\alpha \text{SE}[\hat{\zeta}^*(\beta)] \quad (5)$$

where  $z_\alpha$  is the  $100(1 - \frac{1}{2}\alpha)$ th percentile of the standard Normal distribution. To construct such a confidence interval, we proceed as follows. Given a value  $\zeta$  in the range

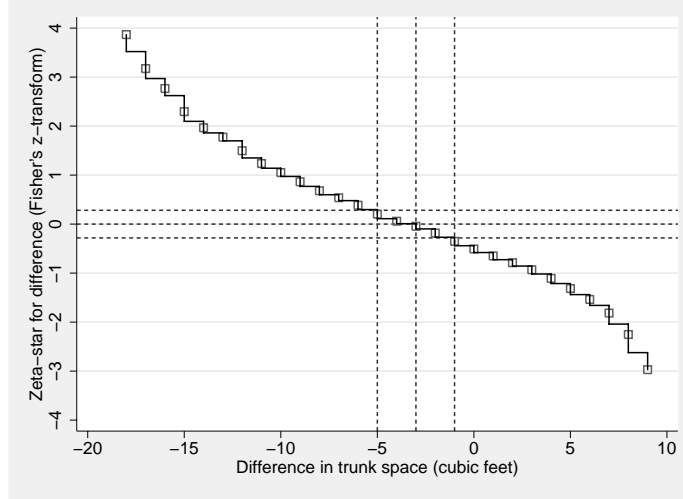


Figure 1:  $\zeta^*(\beta)$  plotted against the difference  $\beta$  in trunk space between non-US and US cars.

of  $\zeta(\cdot)$ , we define

$$\begin{aligned}
 B_L(\zeta) &= \sup \{ \beta : \zeta^*(\beta) > \zeta \} \\
 B_R(\zeta) &= \inf \{ \beta : \zeta^*(\beta) < \zeta \} \\
 B_C(\zeta) &= \begin{cases} \text{Undefined,} & \text{if } B_L(\zeta) = -\infty \text{ and } B_R(\zeta) = \infty, \\ B_L(\zeta), & \text{if } B_L(\zeta) > -\infty \text{ and } B_R(\zeta) = \infty, \\ B_R(\zeta), & \text{if } B_R(\zeta) < +\infty \text{ and } B_L(\zeta) = -\infty, \\ [B_L(\zeta) + B_R(\zeta)]/2, & \text{otherwise.} \end{cases} \quad (6)
 \end{aligned}$$

(By convention, the supremum (or infimum) of a set unbounded to the right (or left) are defined as  $+\infty$  (or  $-\infty$ ), respectively, and the supremum and infimum for an empty set are  $-\infty$  and  $+\infty$ , respectively.) Clearly,  $B_L(\zeta) \leq B_C(\zeta) \leq B_R(\zeta)$ , and the values of  $B_L(\zeta)$  and  $B_R(\zeta)$  (if finite) can be either the same observed slope, or two successive observed slopes. The confidence interval for a  $100q$ th percentile slope is centered on the sample  $100q$ th percentile slope, defined as

$$\hat{\xi}_q = B_C[\zeta(1 - 2q)] \quad (7)$$

The lower and upper confidence limits for a  $q$ th percentile slope are, respectively,

$$\hat{\xi}_q^{(\min)} = B_L\{ \zeta(1 - 2q) - z_\alpha \widehat{\text{SE}}[\zeta^*(\hat{\xi}_q)] \}, \quad \hat{\xi}_q^{(\max)} = B_R\{ \zeta(1 - 2q) + z_\alpha \widehat{\text{SE}}[\zeta^*(\hat{\xi}_q)] \} \quad (8)$$

If `tdist` is specified, then `censlope` uses the  $t$ -distribution with  $\nu = N - 1$  degrees of freedom if there are  $N$  unclustered observations, or with  $\nu = N_{\text{clust}} - 1$  degrees of freedom if there are  $N_{\text{clust}}$  clusters, instead of the normal distribution, and therefore  $t_{\nu, \alpha}$  replaces

$z_\alpha$  in (8). Note that the upper and lower confidence limits may occasionally be infinite, in the case of extreme percentiles and/or very small sample numbers. `censlope` codes these infinite limits as plus or minus the Stata `creturn` value `c(maxdouble)`, which is the system maximum double precision value (see on-line help for `creturn`).

Figure 1 illustrates these formulas in the case of the  $Y$ -variable `trunk` and the  $X$ -variable `foreign` in the `auto` data. The median difference in trunk capacity  $\hat{\zeta}_{0.5}$ , and its lower and upper 95% confidence limits, are shown as reference lines on the horizontal axis. The estimated median difference in trunk space between non-US and US cars is -3 cubic feet, with 95% confidence limits from -5 to -1 cubic feet. The reference lines on the vertical axis are the optimum, minimum and maximum values of  $\zeta^*(\beta)$  required for  $\beta$  to be in the confidence interval.

Note that `censlope` inherits *all* the options of `somersd`, so  $\theta(X, Y - \beta X)$  in (1) can stand for any of the generalized versions of Somers'  $D$  and Kendall's  $\tau_a$  described in Newson (2006b). We can therefore estimate generalized percentile slopes or differences, defined in terms of generalized Somers'  $D$  or Kendall's  $\tau_a$  parameters. For instance, we can use the `wstrata()` option to estimate median slopes and differences restricted to comparisons within strata defined by a confounding variable, or we might use the option `funtype(wcluster)` to estimate within-cluster median differences and slopes. In the terminology of Serfling (1980), the Theil-Sen percentile slope is an  $M$ -estimate if `funtype(wcluster)` is specified, a hybrid between an  $M$ -estimate and a  $U$ -statistic if `funtype(bcluster)` is specified, and a hybrid between an  $M$ -estimate and a  $V$ -statistic if `funtype(vonmises)` is specified.

### 3.1 Numerical evaluation of $B_L(\zeta)$ and $B_R(\zeta)$

We can see, by (6), (7) and (8), that the key to calculating confidence intervals for percentile slopes is calculating  $B_L(\zeta)$  and  $B_R(\zeta)$  for a given  $\zeta$ . Traditionally, this has been done by calculating every possible pairwise slope  $(Y_i - Y_j)/(X_i - X_j)$  for each pair of observations in the sample to make a dataset of all pairwise slopes, and by using this dataset to find the median and other percentile slopes. This requires an amount of computational time, and data storage space, proportional to  $N^2$ , where  $N$  is the number of observations. For this reason, confidence intervals for median slopes have traditionally only been calculated for small samples, as have confidence intervals for other rank statistics, such as Somers'  $D$  and Kendall's  $\tau_a$ , which are also commonly calculated by comparing every pair of  $(X, Y)$ -pairs. See Sprent and Smeeton (2001) for some worked examples using traditional methods.

It is not necessary to compare each pair of  $(X, Y)$ -pairs. `somersd` uses the algorithm of Newson (2006a), which calculates Somers'  $D$ , Kendall's  $\tau_a$  and their jackknife variances in a time asymptotically proportional to  $N \log N$ , using a search tree to avoid having to compare every pair of  $(X, Y)$ -pairs. We can therefore use `somersd` to calculate  $\zeta^*(\beta)$  for any  $\beta$  in a time proportional to  $N \log N$ . `censlope` uses versions of some of the iterative numerical methods of Chapter 9 of Press et al. (1992), modified for step functions, to evaluate  $B_L(\zeta)$  and  $B_R(\zeta)$ , for a given  $\zeta$ . This is done by defining the

object function  $\omega(\beta) = \zeta^*(\beta) - \zeta$  and attempting to find a solution in  $\beta$  to the equation

$$0 = \omega(\beta) = \zeta^*(\beta) - \zeta \quad (9)$$

using `somersd` to calculate  $\omega(\beta)$ . This requires a computational time of order  $N_{\text{eval}}N \log N$ , where  $N_{\text{eval}}$  is the number of evaluations of the object function in the iteration sequence. For very large datasets ( $N > 1000$ ), this will typically take less time than a quadratic algorithm that compares all pairs of  $(X, Y)$ -pairs. However, in small datasets, such as the `auto` data, `cendif` typically takes *much* less time to calculate a Hodges–Lehmann median difference, using its quadratic algorithm, than `censlope` takes using one of its iterative algorithms to do the same. This is not surprising. The performance study of Newson (2006a) seems to imply that, if there are less than 100 observations, then the execution time of `somersd` is dominated by “constant” terms not dependent on sample size, whether `somersd` is using a quadratic algorithm or a search tree algorithm. Therefore, we would expect the computational time for an iteration sequence, involving  $N_{\text{eval}}$  calls to `somersd`, to have a component proportional to  $N_{\text{eval}}$ , which will dominate execution time if the sample size is small and the number of iterations is large.

The algorithms used by `censlope` are implemented in the Mata language, and use versions of standard bracket convergence methods for finding roots, modified for step functions. To solve an equation of the form (9), we would normally start with two  $\beta$ -values  $\beta_0$  and  $\beta_1$ , whose corresponding respective  $\omega$ -values  $\omega_0$  and  $\omega_1$  bracket zero, meaning that  $\omega_0 \omega_1 < 0$  (because the two  $\omega$ -values have opposite signs). If  $\omega(\cdot)$  is continuous, then, by the intermediate-value theorem, there will be a solution to (9) between  $\beta_0$  and  $\beta_1$ , and this solution will be unique if  $\omega(\cdot)$  is strictly monotonic. However, in this case,  $\omega(\cdot)$  is not continuous, but a nonincreasing step function similar to Figure 1. Therefore, instead of expecting to find a unique solution to (9), we try to find a supremum (or infimum) of the set of  $\beta$ -values with positive (or negative) values of the object function. In this case, the two  $\omega$ -values are said to bracket zero if and only if

$$\text{sign}(\omega_1) \neq 0 \quad \text{and} \quad \text{sign}(\omega_1) \neq \text{sign}(\omega_0) \quad (10)$$

In other words,  $\omega_1$  is a strict bracket, which must not be zero, whereas  $\omega_0$  is a partial bracket, which may either be zero or have the opposite sign to  $\omega_1$ . During each iteration, we compute a new  $\beta$ -value  $\beta_{\text{new}}$ , between  $\beta_0$  and  $\beta_1$ , with a corresponding  $\omega$ -value  $\omega_{\text{new}} = \omega(\beta_{\text{new}})$ . In the next iteration, the pair  $(\beta_{\text{new}}, \omega_{\text{new}})$  will replace  $(\beta_1, \omega_1)$  if  $\text{sign}(\omega_{\text{new}}) = \text{sign}(\omega_1)$ , and will replace  $(\beta_0, \omega_0)$  otherwise. Iterations proceed until  $\beta_0$  and  $\beta_1$  have a relative difference no more than the value of the `tolerance()` option. When this has happened, we can use either of the  $\beta$ -values to estimate  $B_L(\zeta)$  or  $B_R(\zeta)$  (depending on whether we initialized  $\beta_1 < \beta_0$  or  $\beta_0 < \beta_1$ ).

The numerical methods specified by the `technique()` option differ in the method used to calculate  $\beta_{\text{new}}$ . The technique `bisect` does this using the simple bisection formula  $\beta_{\text{new}} = (\beta_0 + \beta_1)/2$ . The technique `regula` uses simple bisection if  $\omega_0 = 0$ , and uses the regula falsi (or false position) method otherwise. The technique `ridders` uses simple bisection if  $\omega_0 = 0$ , and uses the method of Ridders (1979) otherwise. The simple bisection method is guaranteed to converge slowly, whereas the modified regula falsi and

Ridders methods will be faster if the object function  $\omega(\cdot)$  is nearly continuous, but may be a lot slower if  $\omega(\cdot)$  is very discrete. The user may specify a combination of methods, such as starting with the regula falsi or Ridders method for earlier iterations (when the object function is nearly continuous over a long interval), and moving to the bisection method later (when the object function is highly discrete over a short interval).

For each percentage  $100q$ , `censlope` attempts to evaluate  $B_L[\zeta(1-2q)]$  and  $B_R[\zeta(1-2q)]$  in order to evaluate the percentile estimate  $\hat{\xi}_q$ , and then (if this evaluation is successful) evaluates the two confidence limits. This implies 4 sequences of iterations, to evaluate the “left estimate”, the “right estimate” and the two confidence limits, respectively. Typically, using the default tolerance of  $1e-6$ , and the “slow but sure” bisection method, this implies 4 sets of around 20 iterations. Together with the initialization of the brackets, this implies a large number (80–100) of calls to `somersd`. However, that number is usually fewer than 100 evaluations per percentile, implying less work than (say) bootstrapping Somers’  $D$ , which would typically involve at least 1000 evaluations. On the other hand, if the sample size is large, then this method would probably be unthinkable for practical statisticians without the algorithm of Newson (2006a).

### 3.2 Comparisons with existing methods

Sen (1968) developed a confidence interval formula for  $\hat{\xi}_q$  in the special case where  $q = 0.5$ ,  $\theta(Y, X) = \tau(Y, X)$  and  $\zeta(\theta) = \theta$ , using methods similar to the present ones. In this special case, (1) becomes simply  $\tau(Y - \beta X, X) = 0$ . The main difference from the present method was in the method used for calculating the distribution of  $\zeta^*(\beta)$ . Sen assumed that the variables  $X$  and  $Y - \beta X$  were not only “Kendall-uncorrelated”, but also statistically independent. For small sample sizes ( $N \leq 10$ ), the confidence interval was calculated using tables of the exact distribution of the sample Kendall’s  $\tau_a$ , based on that assumption. For larger sample sizes, the population standard error  $SE[\zeta^*(\beta)]$  was calculated from the marginal sample distribution of  $X$ , using the same assumption. (See Kendall and Gibbons (1990) for tables of the exact distribution for small sample sizes, and also for a demonstration that the Central Limit Theorem works *very* well at sample sizes as small as 8 for the sample Kendall’s  $\tau_a$  under the null hypothesis of independence.) The assumption of independence between the predictor variable  $X$  and the “residuals”  $Y - \beta X$  implies that the conditional population distributions of  $Y$ , given each value of  $X$ , are different only in location, and may not differ in the conditional variance, or indeed in any other conditional moment about the mean. The original Sen method therefore does not use the assumption of Normality, but does use the assumption of homoskedasticity, which typically causes more problems when it is wrong.

Lehmann (1963) derived a confidence interval for the Hodges–Lehmann median difference, which is the Theil–Sen slope for binary  $X$ -variables, based on the same assumption of independence. This method was popularized by Conover (1980), Campbell and Gardner (1988) and Altman et al. (2000), and is available in unofficial Stata, using Duolao Wang’s `npshift` routine (Wang (1999)) or Patrick Royston’s `cid` routine, downloadable from SSC. The method is essentially a special case of the Sen (1968) method, and is presumably subject to the same cautions.

The method used by `censlope` and `cendif`, by contrast, can estimate percentile differences other than the median difference. Even in the case of a median difference, the predictor variable  $X$  and the “residuals”  $Y - \beta X$  are only assumed to be “Kendall-uncorrelated”, and not necessarily independent. The population standard error  $\text{SE}[\zeta^*(\beta)]$  is estimated using the sample standard error  $\widehat{\text{SE}}[\zeta^*(\beta)]$ , which is calculated using an infinitesimal jackknife method described in Newson (2006b). This method is robust to heteroskedasticity, *probably* at the price of being less robust to extremely small sample sizes than the traditional methods. Therefore, the method of `censlope` can be compared to the original Sen method as Huber confidence intervals can be compared to maximum-likelihood or quasi-likelihood confidence intervals, and the method of `cendif` can be compared to the Lehmann method as the unequal-variance  $t$ -test can be compared to the equal-variance  $t$ -test. Lehmann’s method, like the equal-variance  $t$ -test, assumes that you can use data from the larger of two samples to estimate the population variability of the smaller sample. The `censlope` method, like the unequal-variance  $t$ -test, assumes that you can use data from the *smaller* of the two samples to estimate the population variability of the smaller sample. At present, if the `tdist` option is specified for `censlope` or `cendif`, then the number of degrees of freedom is set to one less than the sum of the two sample numbers. This is in contrast to the unequal-variance  $t$ -test, which typically uses a more complicated formula (Satterthwaite (1946)), which is usually less generous with degrees of freedom if the smaller sample size is very small.

The issue of heteroskedasticity, as it affects the  $t$ -test, is discussed in Moser et al. (1989) and in Moser and Stevens (1992), who explored the issue, using exact analytical formulas to compare the equal-variance  $t$ -test with the Satterthwaite unequal-variance  $t$ -test. Their conclusion (as I understand it) appears to be that we should view the equal-variance  $t$ -test as a special method for use only when we “know” that the sub-population variances are equal, rather than to follow the more “traditional” practice of viewing the unequal-variance  $t$ -test as a special method for use only when we “know” that the sub-population variances are unequal. I have carried out some unpublished simulations, comparing `cendif` to the Lehmann method, and to the two  $t$ -tests. These simulations, some of which are briefly described in Newson (2000b) and in Newson (2002), seem to point to a similar recommendation regarding the two types of rank-based methods for median differences. However, more work is probably required on this issue.

An alternative method of defining heteroskedasticity-consistent confidence intervals for the Theil–Sen median slope is the percentile bootstrap, recommended by Wilcox (1998). Bootstrapping `censlope` or `cendif` may be an option, at least for small samples, where the computational cost of evaluating a single sample median slope or difference, using a quadratic or iterative method, is low enough to allow us to evaluate a large number of subsample median slopes or differences. `censlope` adds the options of estimating clustered and/or stratified median slopes and differences, and also the option of non-bootstrap confidence intervals for very large samples. The infinitesimal jackknife method, used by `somersd`, is usually considered to be an inferior substitute for the bootstrap method applied to the same parameter. However, in this case, the infinitesimal

jackknife standard error calculated by `censlope` is not for the median slope itself, but for another parameter (Somers'  $D$  or Kendall's  $\tau_a$ ), for which the Central Limit Theorem works very fast, especially under the null hypothesis (Kendall and Gibbons (1990)). This *might* limit the advantage of the bootstrap over the infinitesimal jackknife. On the other hand, a possible future compromise *might* be to modify `censlope` to allow it to bootstrap Somers'  $D$  or Kendall's  $\tau_a$ , and thereby to substitute bootstrap-based formulas for formulas (5) and (8) when calculating confidence intervals for the percentile slope itself. Whether we use the bootstrap or the infinitesimal jackknife, it is probably a good idea, if the sample size is large, to calculate the Theil–Sen median slope using a non-quadratic algorithm, which does not require calculation of all the individual pairwise slopes.

## 4 Examples

These examples introduce some of the capabilities of `censlope`. There are more examples in the online help for `censlope`, and in the manual `censlope.pdf`, which is distributed with the `somersd` package as an ancillary file.

### 4.1 Weight per inch in the auto data

In the `auto` data, we can use `censlope` to estimate the median slope of `weight` (in US pounds) with respect to `length` (in US inches) as follows:

```
. censlope weight length, tdist
Outcome variable: weight
Somers' D with variable: length
Transformation: Untransformed
Valid observations: 74
Degrees of freedom: 73
Symmetric 95% CI
```

length	Coef.	Jackknife Std. Err.	t	P> t	[95% Conf. Interval]	
weight	.8286359	.0275321	30.10	0.000	.7737644	.8835073

```
95% CI(s) for percentile slope(s)
Percent Pctl_Slope Minimum Maximum
50 32.745114 30.508468 35.185195
```

The untransformed Somers'  $D$  of `weight` with respect to `length` is 0.83, with a confidence interval from 0.77 to 0.88, indicating that, in the population from which these cars were sampled, a longer car is 77% to 88% more likely to be heavier than a shorter car than to be lighter than a shorter car. Each additional inch of length typically adds 30.51 to 35.19 pounds of weight.

If we use the  $z$ -transform for Somers'  $D$ , then the results are as follows:

```
. censlope weight length, tdist transf(z)
```

```
Outcome variable: weight
Somers' D with variable: length
Transformation: Fisher's z
Valid observations: 74
Degrees of freedom: 73
Symmetric 95% CI for transformed Somers' D
```

length	Jackknife		t	P> t	[95% Conf. Interval]	
	Coef.	Std. Err.				
weight	1.183767	.0878602	13.47	0.000	1.008662	1.358873

```
Asymmetric 95% CI for untransformed Somers' D
Somers_D      Minimum      Maximum
weight      .82863585   .76520811   .87613131
```

```
95% CI(s) for percentile slope(s)
Percent Pctl_Slope      Minimum      Maximum
50      32.745093      30.571414   35.121969
```

This time, Somers'  $D$  is 0.77 to 0.88, implying (again) that longer cars are 77% to 88% more likely to be heavier than shorter cars than to be lighter than shorter cars. The typical increase in weight per additional inch of length is 30.57 to 35.12 pounds per inch, which is very similar to the previous confidence interval.

Transformations such as Fisher's  $z$  are more likely to be important in estimating percentile slopes other than the median. We can ask for the 25th and 75th percentiles as well, using the `centile()` option:

```
. censlope weight length, tdist transf(z) centile(25(25)75)
Outcome variable: weight
Somers' D with variable: length
Transformation: Fisher's z
Valid observations: 74
Degrees of freedom: 73
Symmetric 95% CI for transformed Somers' D
```

length	Jackknife		t	P> t	[95% Conf. Interval]	
	Coef.	Std. Err.				
weight	1.183767	.0878602	13.47	0.000	1.008662	1.358873

```
Asymmetric 95% CI for untransformed Somers' D
Somers_D      Minimum      Maximum
weight      .82863585   .76520811   .87613131
```

```
95% CI(s) for percentile slope(s)
Percent Pctl_Slope      Minimum      Maximum
25      24.102562      19.999992   27.06897
50      32.745093      30.571414   35.121969
75      41.818174      38.620683   46.200022
```

We see that the 25th percentile slope is 20.00 to 27.07 pounds per inch, and that the 75th percentile slope is 38.62 to 46.20 pounds per inch.

We can also produce plots of observed and fitted values, using the `ystargenerate()` option of `censlope`. This can be done as follows:



```

. censlope weight length, tdist transf(z) ystar(resid)
Outcome variable: weight
Somers' D with variable: length
Transformation: Fisher's z
Valid observations: 74
Degrees of freedom: 73
Symmetric 95% CI for transformed Somers' D

```

length	Coef.	Jackknife Std. Err.	t	P> t	[95% Conf. Interval]	
weight	1.183767	.0878602	13.47	0.000	1.008662	1.358873

```

Asymmetric 95% CI for untransformed Somers' D
Somers_D Minimum Maximum
weight .82863585 .76520811 .87613131
95% CI(s) for percentile slope(s)
Percent Pctl_Slope Minimum Maximum
50 32.745093 30.571414 35.121969
. egen intercept = median(resid)
. gene wthat = weight - resid + intercept
. label var wthat "Fitted weight"
. twoway scatter weight length || line wthat length, lpattern(solid)

```

After executing `censlope`, we use `egen` to calculate the median of the variable `resid`, generated by the `ystargenerate()` option, which stores the “residuals”  $Y - \beta X$ , where  $Y$  is `weight`,  $X$  is `length`, and  $\beta$  is the median slope. This median is stored in a new variable, named `intercept`. Then, we generate the fitted values of `weight` in a new variable `wthat`, calculated by subtracting  $Y - \beta X$  from  $Y$  to obtain  $\beta X$  and then adding `intercept`. These fitted values are plotted as a line against `length`, and the observed weight values are superimposed to create the graph of Figure 2.

## 4.2 Prenatal paracetamol and immunoglobulin E

The Avon Longitudinal Study of Pregnancy and Childhood (ALSPAC) is a birth cohort study based at the University of Bristol, UK. For further information, refer to the study website at <http://www.alspac.bris.ac.uk>. As part of the study, the mothers of 12127 children were asked whether they ever used paracetamol (acetaminophen) in weeks 20–32 of pregnancy. At 7 years of age, total immunoglobulin E (IgE) was measured in the blood of 4848 of these children. IgE is viewed as a measure of allergic tendency, and is raised in individuals suffering from allergic diseases such as asthma. Shaheen et al. (2005) reported that, in ALSPAC, the children of paracetamol users typically had higher IgE levels than children of paracetamol non-users, based on estimates of geometric mean ratios.

The distribution of total IgE, expressed in kilounits per litre (kU/l), in the 4848 children with data on IgE and on maternal paracetamol use in late pregnancy, is given in Figure 3. Note that the distribution is non-Normal and has a long tail of extremely high values. 2051 of these children had mothers who reported using paracetamol in late

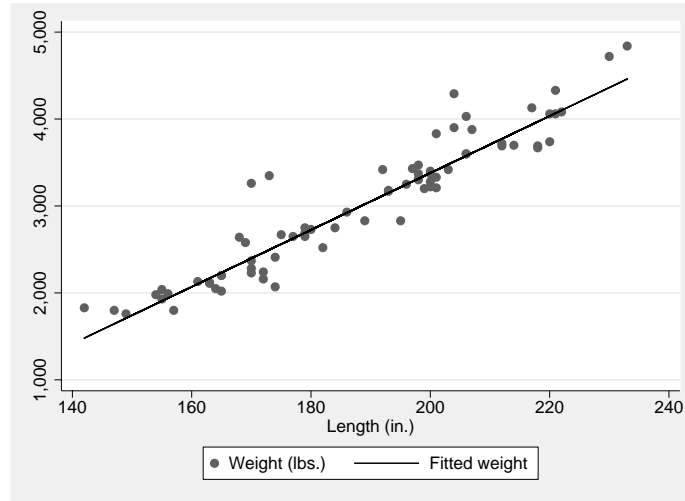


Figure 2: Observed and fitted car weights plotted against car length.

pregnancy, and the remaining 2797 had mothers who reported not using paracetamol.

We used `censlope` to compare the IgE levels in children exposed and unexposed to maternal paracetamol use in late pregnancy, using Somers'  $D$  and the Hodges–Lehmann median IgE ratio. Given a randomly-sampled paracetamol-exposed child and a randomly-sampled unexposed child, Somers'  $D$  is the difference between the probability that the exposed child has the higher IgE level and the probability that the unexposed child has the higher IgE level. The Hodges–Lehmann median ratio is the median ratio between IgE levels in two such randomly-sampled children, and is defined as the exponential of the Hodges–Lehmann median difference between the logged IgE values in the two groups, and estimated using the `eform` option of `censlope`. The results were as follows:

```
. censlope lnigetot para32g, transf(z) eform
Outcome variable: lnigetot
Somers' D with variable: para32g
Transformation: Fisher's z
Valid observations: 4848
Symmetric 95% CI for transformed Somers' D
```

para32g	Coef.	Jackknife Std. Err.	z	P> z	[95% Conf. Interval]
lnigetot	.0533954	.0168421	3.17	0.002	.0203856 .0864053

```
Asymmetric 95% CI for untransformed Somers' D
Somers_D Minimum Maximum
lnigetot .05334475 .02038276 .0861909
95% CI(s) for percentile ratio(s)
Percent Pctl_Ratio Minimum Maximum
```

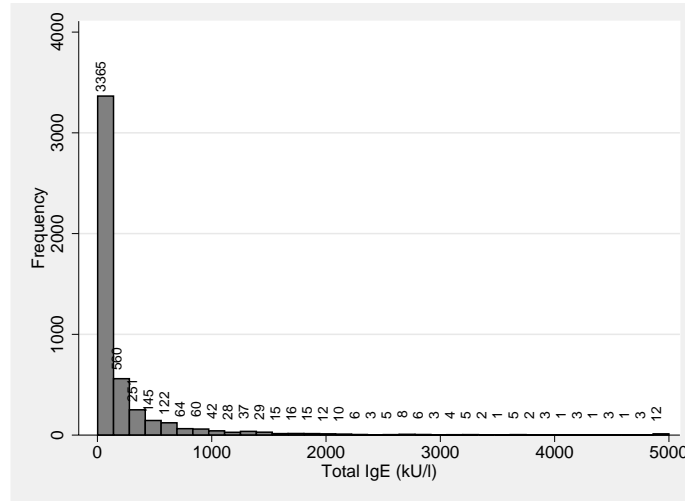


Figure 3: Distribution of blood IgE in 4848 children in ALSPAC.

50    1.172549    1.0616111    1.2944986

Here, `lnigetot` is the natural log of total IgE, and `para32g` is a binary variable, indicating paracetamol exposure during weeks 20–32 of gestation. From the asymmetric confidence interval for the untransformed Somers'  $D$ , we see that, if we choose an exposed child and an unexposed child at random, then the exposed child is 2.0% to 8.6% more likely than the unexposed child to have the higher IgE. From the confidence interval for the 50th percentile (or median) ratio, we can see that the median ratio is 1.06 to 1.29, implying that the exposed child typically has 6% to 29% more IgE than the unexposed child.

However, these are only crude, unadjusted estimates, and the effects that they represent could be due to potential confounding variables. To produce confounder-adjusted estimates, we used a propensity score, as defined in Rosenbaum and Rubin (1983) and Rosenbaum (2002). We defined this score by fitting a logistic regression model, with `para32g` as the outcome, to data from the 12127 children with paracetamol data. The predictors in this model were the following confounders: gender, maternal age, prenatal tobacco exposure, mother's education, housing tenure, parity, maternal anxiety, maternal ethnic origin, multiple pregnancy, birth weight, gestational age at birth, head circumference, antibiotics in pregnancy, alcohol intake in pregnancy, maternal disease and infection history, younger siblings, presence of pets, breast feeding, day care, dampness problems, passive smoking exposure after birth, obesity index at 7 years. (Not all of these confounders could have had a causal effect on prenatal paracetamol exposure, but they could all be indirect indicators of prenatal proneness to paracetamol exposure.) The propensity score was defined as the predicted log odds of paracetamol exposure from this regression model. Using the `xtile` command (see [D] `pctile`), we defined 32

paracetamol propensity groups, with approximately equal numbers.

`somersd`, and therefore `censlope`, has a `wstrata()` option, allowing stratified versions of Somers'  $D$  and median slopes, restricted to comparisons between pairs of observations in the same stratum. We measured the confounder-adjusted paracetamol effect using `censlope` with the option `wstrata(pg_para32g)`, where `pg_para32g` is a discrete variable indicating which of the 32 paracetamol-propensity groups a child belongs to, based on that child's confounder values. The results were as follows:

```
. censlope lnigetot para32g, transf(z) eform wstrata(pg_para32g)
Outcome variable: lnigetot
Somers' D with variable: para32g
Transformation: Fisher's z
Within strata defined by: pg_para32g
Valid observations: 4848
Symmetric 95% CI for transformed Somers' D
```

para32g	Coef.	Jackknife Std. Err.	z	P> z	[95% Conf. Interval]	
lnigetot	.0416191	.018089	2.30	0.021	.0061653	.0770729

```
Asymmetric 95% CI for untransformed Somers' D
Somers_D      Minimum      Maximum
lnigetot      .04159508    .00616518    .07692067
95% CI(s) for percentile ratio(s)
Percent  Pctl_Ratio  Minimum      Maximum
50       1.1256541   1.0165742    1.2556066
```

This time, the adjusted Somers'  $D$  is between 0.006 and 0.077, and the adjusted Hodges–Lehmann median ratio is between 1.017 and 1.256. Therefore, if we sample a random exposed child and a random unexposed child *from the same propensity stratum*, then it is 0.6% to 7.7% more likely that the exposed child will have the higher IgE than that the unexposed child will have the higher IgE, and the exposed child will typically have 1.7% to 25.6% more IgE than the unexposed child. Therefore, sampling similarly paracetamol-prone children does not seem to alter the relative exposed–unexposed IgE difference very much. These conclusions are (reassuringly) similar to those of Shaheen et al. (2005).

Note that the sample size of 4848 is much larger than those of most samples conventionally analysed using rank methods, and is in the range at which the computational methods used by `censlope` begin to have an advantage. The unadjusted analysis presented above typically takes 2 minutes using `censlope`, and 4 minutes using `cendif`, on my system, which is based on a 2.79Ghz Intel Pentium 4 CPU with 0.99Gb RAM running Windows XP. As sample size increases further, so will the ratio of time and space requirements between `cendif` (which uses a quadratic–time and quadratic–space algorithm) and `censlope`.

## 5 Summary

The `censlope` module is a major extension to the `somersd` package, enabling the estimation of generalized Theil–Sen percentile slopes and Hodges–Lehmann percentile differences, corresponding to the generalized Somers’  $D$  and Kendall’s  $\tau_a$  parameters introduced in Newson (2006b). All of these generalized parameters are estimated with confidence intervals, and may be restricted to comparisons within or between clusters and/or strata defined by a confounder, or by a propensity score summarizing multiple confounders. The `somersd` package therefore allows users to do more with rank methods than they were probably accustomed to do, although we may still need regression methods to define a propensity score.

Rank parameters of the “Somers’  $D$  family” have the advantage of being robust to distributional assumptions. This is mainly because Somers’  $D$  and Kendall’s  $\tau_a$  have “democratic” influence functions, based on a principle of “one comparison, one vote”, causing the Central Limit Theorem (in most cases) to work faster than it would for comparable regression parameters. (See Hampel (1974) and Hampel et al. (1986) for more about influence functions.) This robustness *must*, to an extent, be purchased at the price of being less robust to small sample numbers. The argument of Fisher (1935) implies that, *if* we know the distributional family *a priori*, *then* an estimate for a median slope or difference based on maximum–likelihood estimators will have a lower asymptotic variance than the corresponding Theil–Sen or Hodges–Lehmann statistics. The contrast in power may be spectacular at tiny sample sizes, when using a  $t$ –test may reduce the minimum detectable difference from infinity to a finite difference. (This is why `censlope` and `cendif` can produce infinite confidence limits.) At larger sample sizes, there is typically a more modest contrast in power, such as a 5% reduction in the minimum detectable difference, and even this may be conditional on guessing the distributional family right in advance. However, more work is needed (and is in progress), in order to find out more about the tradeoffs involved.

## 6 Acknowledgements

We are extremely grateful to all the families who took part in the ALSPAC study, the midwives for their help in recruiting them, and the whole ALSPAC team, which includes interviewers, computer and laboratory technicians (in particular Fay Stratton who was involved in measurement of total IgE), clerical workers, research scientists, volunteers, managers, receptionists and nurses. Particular thanks are due to our colleague Seif Shaheen, of Imperial College London, UK, who had the idea on which the analyses presented in Subsection 4.2 were based. The UK Medical Research Council, the Wellcome Trust and the University of Bristol provide core support for ALSPAC. The Medical Research Council also funded collection of IgE data. Roger Newson’s work at Imperial College London is financed by the UK Department of Health. I would also like to thank my former colleague Nigel Smeeton, of King’s College London, UK, for drawing my attention to the Theil–Sen slope and to his work on rank methods with Peter Sprent (Sprent and Smeeton (2001)), and all at StataCorp for the Mata language,

which made the `censlope` module possible.

## 7 References

- Altman, D. G., D. Machin, T. N. Bryant, and M. J. Gardner. 2000. *Statistics with Confidence*. 2nd ed. London, UK: British Medical Journal Books.
- Campbell, M. J., and M. J. Gardner. 1988. Calculating confidence intervals for some non-parametric analyses. *British Medical Journal* 296: 1454–1456.
- Conover, W. J. 1980. *Practical Nonparametric Statistics*. 2nd ed. New York, NY: John Wiley & Sons.
- Edwardes, M. D. d. B. 1995. A confidence interval for  $\Pr(X < Y) - \Pr(X > Y)$  estimated from simple cluster samples. *Biometrics* 51: 571–578.
- Fisher, R. A. 1935. The logic of inductive inference. *Journal of the Royal Statistical Society* 98(1): 39–82.
- Hampel, F. R. 1974. The influence curve and its role in robust estimation. *Journal of the American Statistical Association* 69: 383–393.
- Hampel, F. R., E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel. 1986. *Robust Statistics. The Approach Based on Influence Functions*. New York, NY: John Wiley & Sons.
- Hodges, J. L., and E. L. Lehmann. 1963. Estimates of location based on rank tests. *The Annals of Mathematical Statistics* 34: 598–611.
- Kendall, M. G., and J. D. Gibbons. 1990. *Rank Correlation Methods*. 5th ed. New York, NY: Oxford University Press.
- Lehmann, E. L. 1963. Nonparametric confidence intervals for a shift parameter. *The Annals of Mathematical Statistics* 34(4): 1507–1512.
- Moser, B. K., and G. R. Stevens. 1992. Homogeneity of variance in the two-sample means test. *The American Statistician* 46(1): 19–21.
- Moser, B. K., G. R. Stevens, and C. L. Watts. 1989. The two-sample  $t$ -test versus Satterthwaite's approximate  $F$ -test. *Communications in Statistics – Theory and Methods* 18(11): 3963–3975.
- Newson, R. 2000a. `snp15: somersd` – Confidence intervals for nonparametric statistics and their differences. *Stata Technical Bulletin* 55: 47–55. In *Stata Technical Bulletin Reprints*, vol. 10, 312–322. College Station, TX: Stata Press.
- . 2000b. `snp16: Robust confidence intervals for median and other percentile differences between two groups`. *Stata Technical Bulletin* 58: 30–35. In *Stata Technical Bulletin Reprints*, vol. 10, 324–331. College Station, TX: Stata Press.

- . 2002. Parameters behind “nonparametric” statistics: Kendall’s tau, Somers’  $D$  and median differences. *Stata Journal* 2(1): 45–64.
- . 2006a. Efficient calculation of jackknife confidence intervals for rank statistics. *Journal of Statistical Software* 15(1): 1–10. Downloadable from <http://www.jstatsoft.org/> as of 18 September 2006.
- . 2006b. Confidence intervals for rank statistics: Somers’  $D$  and extensions. *Stata Journal* 6(3): 308–334.
- Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. 1992. *Numerical Recipes in C: The Art of Scientific Computing*. 2nd ed. Cambridge, UK: Cambridge University Press.
- Ridders, C. J. F. 1979. A new algorithm for computing a single root of a real continuous function. *IEEE Transactions on Circuits and Systems* CAS-26(11): 979–980.
- Rosenbaum, P. R. 2002. *Observational Studies*. 2nd ed. New York, NY: Springer.
- Rosenbaum, P. R., and D. B. Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70(1): 41–55.
- Satterthwaite, F. E. 1946. An approximate distribution of estimates of variance components. *Biometrics* 2(1): 110–114.
- Sen, P. K. 1968. Estimates of the regression coefficient based on Kendall’s tau. *Journal of the American Statistical Association* 63(324): 1379–1389.
- Serfling, R. J. 1980. *Approximation Theorems of Mathematical Statistics*. New York, NY: John Wiley & Sons.
- Shaheen, S. O., R. B. Newson, A. J. Henderson, J. E. Headley, F. D. Stratton, R. W. Jones, D. P. Strachan, and the ALSPAC Study Team. 2005. Prenatal paracetamol exposure and risk of asthma and elevated immunoglobulin E in childhood. *Clinical and Experimental Allergy* 35: 18–25.
- Somers, R. H. 1962. A new asymmetric measure of association for ordinal variables. *American Sociological Review* 27: 799–811.
- Sprent, P., and N. C. Smeeton. 2001. *Applied Nonparametric Statistical Methods*. 3rd ed. Boca Raton, FA: Chapman & Hall.
- Theil, H. 1950. A rank-invariant method of linear and polynomial regression analysis, I, II, III. *Proceedings of the Koninklijke Nederlandse Akademie Wetenschappen, Series A – Mathematical Sciences* 53: 386–392, 521–525, 1397–1412.
- Wang, D. 1999. sg123: Hodges–Lehmann estimation of a shift in location between two populations. *Stata Technical Bulletin* 52: 52–53. In *Stata Technical Bulletin Reprints*, vol. 9, 255–257. College Station, TX: Stata Press.

Wilcox, R. R. 1998. A note on the Theil-Sen regression estimator when the regressor is random and the error term is heteroscedastic. *Biometrical Journal* 40(3): 261–268.

**About the author**

Roger Newson is a Lecturer in Medical Statistics at Imperial College London, UK, working principally in asthma research. He wrote the `somersd` and `parmed` packages.