The Stata Journal (2006)
6, Number 3, pp. 309–334

Confidence intervals for rank statistics: Somers' D and extensions

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

Abstract. Somers' D is an asymmetric measure of association between two variables, which plays a central role as a parameter behind rank or nonparametric statistical methods. Given predictor variable X and outcome variable Y, we may estimate D_{YX} as a measure of the effect of X on Y, or we may estimate D_{XY} as a performance indicator of X as a predictor of Y. The `somersd` package allows the estimation of Somers' D and Kendall's τ_a with confidence limits as well as p-values. The Stata 9 version of `somersd` can estimate extended versions of Somers' D not previously available, including the Gini index, the parameter tested by the sign test, and extensions to left- or right-censored data. It can also estimate stratified versions of Somers' D, restricted to pairs in the same stratum. Therefore, it is possible to define strata by grouping values of a confounder, or of a propensity score based on multiple confounders, and to estimate versions of Somers' D that measure the association between the outcome and the predictor, adjusted for the confounders. The Stata 9 version of `somersd` uses the Mata language for improved computational efficiency with large datasets.

Keywords: snp15_6, somersd, Somers' D, Kendall's τ_a, Harrell's c, ROC area, Gini index, population-attributable risk, rank correlation, rank-sum test, Wilcoxon test, sign test, confidence intervals, nonparametric methods, propensity score

1 Introduction

Many authors have argued that so-called nonparametric methods are based on population parameters and that these parameters should be estimated with sample statistics and confidence limits, instead of following the traditional practice of calculating p-values only for the sample statistic. Examples include Kendall and Gibbons (1990), Wolfe and Hogg (1971), and Kerridge (1975). In a more recent paper (Newson 2002), the package `somersd`, introduced in Newson (2000a), was demonstrated as a way of estimating these parameters in Stata. Its name is derived from the parameter Somers' D, which plays a central role. Somers' D is defined in terms of Kendall's τ_a, is in turn used in defining the Hodges–Lehmann median difference and the Theil median slope, and has many applications and extensions of its own. Not all these extensions were implemented in the then-current version of `somersd`, which at the time was written in Stata 6.

The release of Stata 9 in 2005 included the C-like compilable matrix programming language Mata, which made possible a major upgrade of the `somersd` package, with

© 2006 StataCorp LP

improvements in computational efficiency that were sometimes spectacular. These improvements made it practical to extend the definitions of Somers' D (and Kendall's τ_a) to include left- and right-censored data and within-strata and within-cluster versions of the parameters. Therefore, `somersd` can now estimate the parameters behind the sign test (see [R] **signrank**) and the Gehan–Breslow test for censored outcomes (Gehan 1965; Breslow 1970), Harrell's c index for censored outcomes (Harrell et al. 1982; Harrell, Lee, and Mark 1996), and the Gini index (Cowell 1995; Jenkins 1999), all of which are special cases and/or transformations of Somers' D. We can also now estimate parameters measuring the association between an outcome variable, Y, and an exposure variable, X, adjusted for one or more confounders, by defining strata using these confounders and then stratifying by these strata. The inability to adjust an association for confounders is traditionally viewed as a major weakness of rank methods, as is their perceived inability to generate confidence intervals. Both weaknesses are often cited as reasons for not using rank methods, despite their strengths of robustness to outliers and to modeling and distributional assumptions (Kirkwood and Sterne 2003).

In this article, I first redefine Kendall's τ_a and Somers' D in section 2 and then describe the current version of the program `somersd` in section 3. In section 4, I present in detail, for reference purposes, the methods and formulas that `somersd` now uses. In section 5, I demonstrate a range of examples and applications.

2 What is Somers' D?

Somers' D is defined in terms of Kendall's τ_a (Kendall and Gibbons 1990), whose population value is traditionally defined as

$$\tau_{XY} = E\left\{\mathrm{sign}(X_1 - X_2)\,\mathrm{sign}(Y_1 - Y_2)\right\}$$

where (X_1, Y_1) and (X_2, Y_2) are bivariate random variables sampled independently from the same population and $E[\cdot]$ denotes expectation. This definition can be generalized to possibly left- or right-censored, stratified, clustered, or weighted data as follows. Suppose that 4-variate observations (X_i, R_i, Y_i, S_i) are sampled from an arbitrary population, using an arbitrary sampling scheme. The R_i are censorship indicators for the corresponding X_i, and the S_i are censorship indicators for the corresponding Y_i. These censorship indicators are negative for left censorship (where the true value of the indicated variable is known to be equal to or less than its recorded value), positive for right censorship (in which the true value of the indicated variable is known to be equal to or greater than its recorded value), and zero for noncensorship (in which the true value is known to be equal to the recorded value). We define a censored sign difference for two values, u and v, with respective censorship indicators p and q, as

$$\mathrm{csign}(u, p, v, q) = \begin{cases} 1, & \text{if } u > v \text{ and } p \geq 0 \geq q \\ -1, & \text{if } u < v \text{ and } p \leq 0 \leq q \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Given two observations (X_i, R_i, Y_i, S_i) and (X_j, R_j, Y_j, S_j), we will call the product of $\mathrm{csign}(X_i, R_i, X_j, R_j)$ and $\mathrm{csign}(Y_i, S_i, Y_j, S_j)$ the concordance–discordance difference for the two observations, and we will say that the two observations are concordant if this product is 1, discordant if the product is -1, and neither concordant nor discordant if the product is 0. We can now redefine Kendall's τ_a as

$$\tau_{XY} = E\left\{ \mathrm{csign}(X_i, R_i, X_j, R_j)\, \mathrm{csign}(Y_i, S_i, Y_j, S_j) \right\} \tag{2}$$

or (in words) as the mean concordance–discordance difference. This expectation can be defined by using weights specific to the observations and/or restrictions to subsets of pairs of observations, defined in terms of the sampling scheme.

The population value of Somers' D (Somers 1962) is defined as

$$D_{YX} = \frac{\tau_{XY}}{\tau_{XX}} \tag{3}$$

Therefore, τ_{XY} is the difference between two probabilities, namely, the probability that the larger of the two X values is associated with the larger of the two Y values and the probability that the larger X value is associated with the smaller Y value. D_{YX} is the difference between the two corresponding *conditional* probabilities, given that the two X values are known to be unequal. Somers' D is related to Harrell's c index (see Harrell et al. [1982] and Harrell, Lee, and Mark [1996]) by $D = 2c - 1$.

2.1 Interpretations of Somers' D

Somers' D usually measures an association between a predictor variable, X, and an outcome variable, Y. Applications of Somers' D fall into two classes:

- We may use D_{YX} as an effect size, measuring the effect of X on Y.
- We may also use D_{XY} as a predictor performance indicator, measuring the performance of X as a predictor of Y.

Examples of the first class usually involve a binary X variable, indicating that an individual is a member of Group A instead of Group B. They are usually motivated by the possibility that we can intervene to change the group membership of an individual and thereby possibly to change the outcome. Somers' D can then be interpreted, rightly or wrongly, as the difference between two probabilities, namely, the probability that we will increase the outcome of a Group A individual by transferring it to Group B and the probability that we will increase the outcome of a Group B individual by transferring it to Group A. This interpretation will arguably be more credible if Somers' D is restricted to comparisons within strata of individuals that are similar to others in the same stratum. Typical examples of the first class include applying the Gehan–Breslow–Wilcoxon test (Gehan 1965; Breslow 1970) to survival outcome data from a randomized clinical trial, or the estimation of a difference in two proportions of successful binary outcomes (which is a trivial case of Somers' D) from binary-outcome data from a randomized clinical trial.

Examples of the second class may involve censored or uncensored Y variables, although the X variables are uncensored. They are usually motivated by the aim of comparing the performance of a predictor, X, with the performance of another predictor, W, by comparing D_{XY} with D_{WY}. Typical examples are discussed in Harrell et al. (1982) and Harrell, Lee, and Mark (1996), which use the c-transformation of Somers' D. An important special case of Harrell's c is the area under the receiver operator characteristic (ROC) curve for binary Y variables (see [R] **roc**, Hanley and McNeil [1982], or Newson [2002]).

The predictor performance indicator D_{XY} has the desirable property that a larger D_{XY} cannot be secondary to a smaller D_{WY}. To understand this point, assume that observations (W_i, X_i, Y_i, S_i) are sampled by some sampling scheme from some population and that the S_i are censorship indicators for the corresponding outcome variables, Y_i. Define the conditional expectation

$$Z(w_i, x_i, w_j, x_j) = E\,\{\,\mathrm{csign}(Y_j, S_j, Y_i, S_i)\,|\,W_i = w_i, X_i = x_i, W_j = w_j, X_j = x_j\,\}$$

for any w_i and w_j in the range of W values and any x_i and x_j in the range of X values. Stating that a positive relationship between X_i and Y_i is caused entirely by a monotonic positive relationship between both variables and W_i is equivalent to stating that

$$Z(w_i, x_i, w_j, x_j) \geq 0 \quad \text{whenever } w_i \leq w_j \text{ and } x_j \leq x_i \tag{4}$$

However, if (4) holds, then $\tau_{WY} - \tau_{XY}$ is nonnegative, and therefore so is $D_{WY} - D_{XY}$. This conclusion follows by an argument similar to (7) and (8) of Newson (2002), which can be generalized trivially to sampling and/or weighting schemes involving nonindependence and/or stratification, as long as the weights are nonnegative. The denominator τ_{YY}, common to D_{WY} and D_{XY}, is simply the proportion of pairs of Y values whose csign, defined by (1), is not set to zero by censored and/or tied Y values. Therefore, D_{XY} is arguably a better indicator of predictor performance than τ_{XY}, because D_{XY} is expressed on a scale from -1 for the best possible negative predictor of Y to $+1$ for the best possible positive predictor of Y, given the level of discreteness and/or censorship existing between the Y values in that particular population.

3 The program somersd

3.1 Syntax

```
somersd [varlist] [if] [in] [weight] [, taua tdist
   transf(transformation_name) cenind(cenind_list) cluster(varname)
   cfweight(expression) funtype(functional_type) wstrata(varlist)
   bstrata(varlist | _n) notree level(#) cimatrix(new_matrix) ]
```

where *transformation_name* is one of

`iden` | `z` | `asin` | `rho` | `zrho` | `c`

and *functional_type* is one of

`wcluster` | `bcluster` | `vonmises`

and *cenind_list* is a list of variable names and/or zeros.

`fweights`, `iweights`, and `pweights` are allowed; see [U] **11.1.6 weight**. They are treated as described in *Interpretation of weights* and *Methods and formulas* below.

`bootstrap`, `by`, `jackknife`, `statsby`, and `svy jackknife` are allowed; see [U] **11.1.10 Prefix commands**.

3.2 Description

`somersd` calculates the rank order statistics Somers' D and Kendall's τ_a, with confidence limits. Somers' D or τ_a is calculated for the first variable of *varlist* as a predictor of each of the other variables in *varlist*, with estimates and jackknife variances and confidence intervals output and saved in `e()` as if for the parameters of a model fit. It is possible to use `lincom` to output confidence limits for differences between the population Somers' D or Kendall's τ_a values.

3.3 Options

`taua` causes `somersd` to calculate Kendall's τ_a. If `taua` is not typed, `somersd` calculates Somers' D.

`tdist` specifies that the estimates are assumed to have a t distribution with $N-1$ degrees of freedom, where N is the number of clusters if `cluster()` is specified or the number of observations if `cluster()` is not specified.

`transf(`*transformation_name*`)` specifies that the estimates are to be transformed, defining estimates for the transformed population value. `iden` (identity or untransformed) is the default. `z` specifies Fisher's z (the hyperbolic arctangent), `asin` specifies Daniels' arcsine, `rho` specifies Greiner's ρ (Pearson correlation estimated using Greiner's relation), `zrho` specifies the z transform of Greiner's ρ, and `c` specifies Harrell's c. If the first variable of *varlist* is a binary indicator of a disease and the other variables are quantitative predictors for that disease, then Harrell's c is the area under the ROC curve. `somersd` recognizes the transformation names `arctanh` and `atanh` as synonyms for `z`, `arcsin` and `arsin` as synonyms for `asin`, `sinph` as a synonym for `rho`, `zsinph` as a synonym for `zrho`, and `roc` and `auroc` as synonyms for `c`. It also recognizes unambiguous abbreviations for transformation names, such as `id` for `iden` or `aur` for `auroc`.

`cenind(`*cenind_list*`)` specifies a list of left- or right-censorship indicators, corresponding to the variables mentioned in the *varlist*. Each censorship indicator is either a variable name or a zero. If the censorship indicator corresponding to a variable is the name of a second variable, then this second variable is used to indicate the censorship status of the first variable. The status is assumed to be left censored (at or below its stated value) in observations in which the second variable is negative, right censored (at or above its stated value) in observations in which the second variable is positive, and uncensored (equal to its stated value) in observations in which the second variable is zero. If the censorship indicator corresponding to a variable is a zero, then the variable is assumed to be uncensored. If `cenind()` is unspecified, then all variables in the *varlist* are assumed to be uncensored. If the list of censorship indicators specified by `cenind()` is shorter than the list of variables specified in the *varlist*, then the list of censorship indicators is completed with the required number of zeros on the right.

`cluster(`*varname*`)` specifies the variable that defines sampling clusters. If `cluster()` is defined, then the variances and confidence limits are calculated assuming that the data represent a sample of clusters from a population of clusters rather than a sample of observations from a population of observations.

`cfweight(`*expression*`)` specifies an expression giving the cluster frequency weights. These cluster frequency weights must have the same value for all observations in a cluster. If `cfweight()` and `cluster()` are both specified, then each cluster in the dataset is assumed to represent a number of identical clusters equal to the cluster frequency weight for that cluster. If `cfweight()` is specified and `cluster()` is unspecified, then each observation in the dataset is treated as a cluster and assumed to represent a number of identical 1-observation clusters equal to the cluster frequency weight. For more details on the interpretation of weights, see *Interpretation of weights* below.

`funtype(`*functional_type*`)` specifies whether the Somers' D or Kendall's τ_a functionals estimated are ratios of between-cluster, within-cluster, or von Mises functionals. These three functional types are specified by the options `funtype(bcluster)`, `funtype(wcluster)`, and `funtype(vonmises)`, respectively. If `funtype()` is not specified, then `funtype(bcluster)` is assumed and between-cluster functionals are estimated. The within-cluster Somers' D is a generalization of the confidence interval corresponding to the sign test (see [R] **signrank**). The Gini coefficient is a special case of the clustered von Mises Somers' D. For more details, see *Methods and formulas*.

`wstrata(`*varlist*`)` specifies a list of variables whose value combinations are the W strata. If `wstrata()` is specified, then `somersd` estimates stratified Somers' D or Kendall's τ_a parameters, applying only to pairs of observations within the same W stratum. These parameters can be used to measure associations within strata, such as associations between an outcome and an exposure within groups defined by values of a confounder or by values of a propensity score based on multiple confounders.

`bstrata(`*varlist* | `_n)` specifies the B strata. If `bstrata()` is specified, then `somersd` estimates Somers' D or Kendall's τ_a parameters specific to pairs of observations from different B strata. These B strata are either combinations of values of a list of variables (if *varlist* is specified) or the individual observations (if `_n` is specified).

B-strata will not often be required. However, if we are estimating the within-cluster Kendall's τ_a (using the options `taua funtype(wcluster)`), then the additional option `bstrata(_n)` will ensure that the within-cluster Kendall's τ_a can take the whole range of values from -1 (for complete discordance within clusters) to $+1$ (for complete concordance within clusters).

`notree` specifies that `somersd` does not use the default search tree algorithm based on Newson (2006) but instead uses a trivial algorithm, which compares every pair of observations and requires much more time with large datasets. This option is rarely used except to compare performance.

`level(`*#*`)` specifies the confidence level, as a percentage, for confidence intervals of the estimates; see [R] **level**.

`cimatrix(`*new_matrix*`)` specifies an output matrix to be created, containing estimates and confidence limits for the untransformed Somers' D, Kendall's τ_a, or Greiner's ρ parameter. If `transf()` is specified, then the confidence limits will be asymmetric and based on symmetric confidence limits for the transformed parameters. This option (like `level()`) may be used in replay mode as well as in nonreplay mode.

If a *varlist* is supplied, then all options are allowed. If not, then `somersd` replays the previous `somersd` estimation (if available), and the only options allowed are `level()` and `cimatrix()`.

3.4 Interpretation of weights

`somersd` inputs up to two weight expressions, which are the ordinary Stata weights given by the *weight* and the cluster frequency weights given by the `cfweight()` option. Internally, `somersd` defines and uses three distinct sets of weights, which are the cluster frequency weights, the observation frequency weights, and the importance weights.

The cluster frequency weights must be the same for different observations in a cluster and imply that each cluster in the input dataset represents a number of identical clusters equal to the cluster frequency weight in that cluster. If `cluster()` is not specified, then the individual observations are clusters, and the cluster frequency weight implies that each 1-observation cluster represents a number of identical 1-observation clusters equal to the cluster frequency weight. The cluster frequency weights are given by `cfweight()` if that option is specified, are set to one if `cfweight()` is unspecified and `cluster()` is specified, are equal to the ordinary Stata weights if neither `cluster()` nor `cfweight()` is specified and the ordinary Stata weights are `fweight`s, and are equal to one otherwise.

The observation frequency weights are summed over all observations in the input dataset to produce the number of observations reported by `somersd` and returned in the estimation result `e(N)` and are not used in any other way. They are set by `cfweight()` if that option is specified and the ordinary Stata weights are not `fweight`s, are equal to the ordinary Stata weights if `cfweight()` is unspecified and the ordinary Stata weights are `fweight`s, are equal to the product of the `cfweight()` expression and the ordinary

Stata weights if `cfweight()` is specified and the ordinary Stata weights are `fweights`, and are equal to one otherwise.

The importance weights are used as described in *Methods and formulas* below. They are equal to the ordinary Stata weights if these are specified and either `cluster()` or `cfweight()` is specified, are equal to the ordinary Stata weights if neither of these two options is specified and the ordinary Stata weights are specified as `pweights` or `iweights`, and are equal to one otherwise.

3.5 Saved results

`somersd` saves the following results in `e()`:

Scalars				
	`e(N)`	number of observations	`e(df_r)`	residual degrees of freedom
	`e(N_clust)`	number of clusters		
Macros				
	`e(cmd)`	`somersd`	`e(param)`	parameter (`somersd` or `taua`)
	`e(parmlab)`	parameter label in output	`e(tdist)`	`tdist` if specified
	`e(depvar)`	name of X variable	`e(clustvar)`	name of cluster variable
	`e(vcetype)`	title used to label Std. Err.	`e(wtype)`	weight type
	`e(wexp)`	weight expression	`e(cfweight)`	`cfweight()` expression
	`e(funtype)`	`funtype()` option	`e(wstrata)`	`wstrata()` option
	`e(bstrata)`	`bstrata()` option	`e(predict)`	program called by `predict`
	`e(transf)`	`transf()` option	`e(tranlab)`	transformation label in output
	`e(properties)`	`b V`		
Matrices				
	`e(b)`	coefficient vector	`e(V)`	variance–covariance matrix
Functions				
	`e(sample)`	marks estimation sample		

`e(depvar)` is (confusingly) the X variable, or predictor variable, in the conventional terminology for defining Somers' D. `somersd` is also different from most estimation commands in that its results are not designed to be used by `predict`.

4 Methods and formulas

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

Somers' D and Kendall's τ_a, in their various forms, can be expressed as ratios of sample means, Hoeffding U statistics, or von Mises V statistics, depending on the functional type specified by the `funtype()` option. `somersd` works by jackknifing the original means, U statistics, and V statistics (Arvesen 1969) and by using Taylor polynomials to derive variances for the ratios. Normalizing and/or variance-stabilizing transformations may then be applied.

We assume the general case where the observations are clustered, which becomes the familiar unclustered case when there is 1 observation per cluster, and that there are N clusters in the sample, sampled from a common population. We assume that there are one or more indexed W strata (defaulting to one all-inclusive W stratum if `wstrata()` is not specified). Two slightly different versions of the notation will be used, depending on whether the user has specified B strata using the `bstrata()` option.

If there are no B strata, then we define w_{fgi}, X_{fgi}, Y_{fgi}, R_{fgi}, and S_{fgi} to be the importance weight, X value, Y value, X-censorship indicator, and Y-censorship indicator, respectively, for the ith observation belonging to the gth W stratum in the fth cluster. Not every possible index combination fgi will correspond to an observation, so all summation over index combinations will be over index combinations corresponding to an observation. For index combinations fgi and jkm corresponding to observations, we can define

$$\begin{aligned} v_{fgi,jkm} &= w_{fgi} w_{jkm} \\ t^{(XY)}_{fgi,jkm} &= v_{fgi,jkm} \, \mathrm{csign}(X_{fgi}, R_{fgi}, X_{jkm}, R_{jkm}) \, \mathrm{csign}(Y_{fgi}, S_{fgi}, Y_{jkm}, S_{jkm}) \end{aligned}$$

We will use the plus-substitution notation to define (for instance)

$$\begin{aligned} v_{fgi,jk+} &= \sum_m v_{fgi,jkm}, & t^{(XY)}_{fgi,jk+} &= \sum_m t^{(XY)}_{fgi,jkm} \\ v_{fgi,j++} &= \sum_k v_{fgi,jk+}, & t^{(XY)}_{fgi,j++} &= \sum_k t^{(XY)}_{fgi,jk+} \end{aligned}$$

and any other sums over any other indices. For clusters f and j, we define

$$\phi^{(V)}_{fj} = \sum_g v_{fg+,jg+}, \qquad \phi^{(XY)}_{fj} = \sum_g t^{(XY)}_{fg+,jg+} \tag{5}$$

That is, $\phi^{(V)}_{fj}$ is the sum of pairwise importance weights, and $\phi^{(XY)}_{fj}$ is the sum of pairwise importance–weighted concordance–discordance differences, belonging to pairs of observations, in the same W stratum, of which the first observation is in cluster f and the second observation is in cluster j. The quantities $\phi^{(V)}_{fj}$ and $\phi^{(XY)}_{fj}$ are known as kernels in the terminology of chapter 5 of Serfling (1980) and are defined for any pair of clusters.

If the user has defined B strata, then we define the kernels $\phi^{(V)}_{fj}$ and $\phi^{(XY)}_{fj}$ by a slightly different formula. We define w_{fghi}, X_{fghi}, Y_{fghi}, R_{fghi}, and S_{fghi} to be the importance weight, X value, Y value, X-censorship indicator, and Y-censorship indicator, respectively, for the ith observation belonging to cluster f, W stratum g, and B stratum h. For index combinations $fghi$ and $jklm$ corresponding to observations, we define

$$\begin{aligned}
v_{fghi,jklm} &= w_{fghi}w_{jklm} \\
t^{(XY)}_{fghi,jklm} &= v_{fghi,jklm}\,\mathrm{csign}(X_{fghi},R_{fghi},X_{jklm},R_{jklm}) \\
&\qquad \mathrm{csign}(Y_{fghi},S_{fghi},Y_{jklm},S_{jklm})
\end{aligned}$$

and for clusters f and j we define

$$\begin{aligned}
\phi^{(V)}_{fj} &= \sum_g v_{fg++,jg++} - \sum_g \sum_h v_{fgh+,jgh+} \\
\phi^{(XY)}_{fj} &= \sum_g t^{(XY)}_{fg++,jg++} - \sum_g \sum_h t^{(XY)}_{fgh+,jgh+}
\end{aligned} \tag{6}$$

This time, $\phi^{(V)}_{fj}$ is the sum of products of importance weights, and $\phi^{(XY)}_{fj}$ is the sum of pairwise importance–weighted concordance–discordance differences, belonging to pairs of observations, in the same W stratum and different B strata, of which the first observation is in cluster f and the second observation is in cluster j. If the user has specified `bstrata(_n)`, then every observation is in its own B stratum, and the second terms in the $\phi^{(V)}_{fj}$ and $\phi^{(XY)}_{fj}$ of (6) will then contain only pairs in which an observation is paired with itself.

The kernels $\phi^{(V)}_{fj}$ and $\phi^{(XY)}_{fj}$ of (5) or (6) can be "averaged" over their indices to produce parameters denoted as V and T_{XY}, respectively. Kendall's τ_a and Somers' D are defined as ratios of these averages by

$$\tau_{XY} = T_{XY}/V, \quad D_{YX} = T_{XY}/T_{XX} = \tau_{XY}/\tau_{XX} \tag{7}$$

The way in which the kernels are averaged depends on the `funtype()` option. If the user specifies `funtype(wcluster)`, then V and T_{XY} are within-cluster averages. If the user specifies `funtype(bcluster)` (the default), then V and T_{XY} are between-cluster averages. If the user specifies `funtype(vonmises)`, then V and T_{XY} are overall averages. We always estimate the population parameters V and T_{XY} by using sample statistics $\widehat{V}$ and $\widehat{T}_{XY}$ as point estimates, and we estimate the sampling variances of these point estimates by using a jackknife method, with pseudovalues denoted $\psi^{(V)}_j$ and $\psi^{(XY)}_j$ for the jth cluster.

If the user specifies `funtype(wcluster)`, then `somersd` estimates the parameters

$$V = E\left(\phi^{(V)}_{jj}\right), \quad T_{XY} = E\left(\phi^{(XY)}_{jj}\right)$$

These functionals are population means of within-cluster kernels, and their point estimates are the corresponding sample means

$$\widehat{V} = N^{-1}\sum_{j=1}^{N} \phi^{(V)}_{jj}, \quad \widehat{T}_{XY} = N^{-1}\sum_{j=1}^{N} \phi^{(XY)}_{jj} \tag{8}$$

and the jackknife pseudovalues for the jth cluster are given by

$$\psi_j^{(V)} = \phi_{jj}^{(V)}, \quad \psi_j^{(XY)} = \phi_{jj}^{(XY)} \tag{9}$$

If the user has specified `funtype(bcluster)` (the default) or `funtype(vonmises)`, then `somersd` estimates the parameters

$$V = E\left(\phi_{fj}^{(V)}\right), \quad T_{XY} = E\left(\phi_{fj}^{(XY)}\right) \tag{10}$$

for $f \neq j$. These parameters are known as Hoeffding functionals (Hoeffding 1948) if clusters f and j are assumed to be sampled without replacement and as von Mises functionals (von Mises 1947) if clusters f and j are assumed to be sampled with replacement. If the population from which the clusters are sampled is infinite, then the population Hoeffding functional is equal to the corresponding population von Mises functional.

If the user specifies `funtype(bcluster)`, or does not specify a `funtype()` option, then the point estimates of the population Hoeffding functionals are the corresponding sample Hoeffding functionals, or U statistics in the terminology of Hoeffding (1948), Serfling (1980), Serfling (1988), and Lee (1990). They are defined as $\widehat{V} = \widehat{T}_{XY} = 0$ if $N = 1$, and otherwise as

$$\widehat{V} = \frac{\phi_{++}^{(V)} - \sum_{j=1}^N \phi_{jj}^{(V)}}{N(N-1)}, \quad \widehat{T}_{XY} = \frac{\phi_{++}^{(XY)} - \sum_{j=1}^N \phi_{jj}^{(XY)}}{N(N-1)} \tag{11}$$

The jackknife pseudovalues for the jth cluster are given by $\psi_j^{(V)} = \psi_j^{(XY)} = 0$ if $N = 1$, by

$$\psi_j^{(V)} = \phi_{j+}^{(V)} - \phi_{jj}^{(V)}, \quad \psi_j^{(XY)} = \phi_{j+}^{(XY)} - \phi_{jj}^{(XY)} \tag{12}$$

if $N = 2$, and otherwise as

$$\begin{aligned} \psi_j^{(V)} &= (N-1)^{-1}\left(\phi_{++}^{(V)} - \sum_{k=1}^N \phi_{kk}^{(V)}\right) \\ &\quad - (N-2)^{-1}\left\{\phi_{++}^{(V)} - \sum_{k=1}^N \phi_{kk}^{(V)} - 2\left(\phi_{j+}^{(V)} - \phi_{jj}^{(V)}\right)\right\} \\ \psi_j^{(XY)} &= (N-1)^{-1}\left(\phi_{++}^{(XY)} - \sum_{k=1}^N \phi_{kk}^{(XY)}\right) \\ &\quad - (N-2)^{-1}\left\{\phi_{++}^{(XY)} - \sum_{k=1}^N \phi_{kk}^{(XY)} - 2\left(\phi_{j+}^{(XY)} - \phi_{jj}^{(XY)}\right)\right\} \end{aligned} \tag{13}$$

If the user specifies `funtype(vonmises)`, then the point estimates of the population von Mises functionals are the corresponding sample von Mises functionals, or V statistics in the terminology of Riedwyl (1988) and of chapter 5 of Serfling (1980). They are defined as

$$\widehat{V} = N^{-2}\phi_{++}^{(V)}, \quad \widehat{T}_{XY} = N^{-2}\phi_{++}^{(XY)} \tag{14}$$

and the jackknife pseudovalues for the jth cluster are given by

$$\psi_j^{(V)} = \phi_{jj}^{(V)}, \quad \psi_j^{(XY)} = \phi_{jj}^{(XY)} \tag{15}$$

if $N = 1$, and otherwise by

$$\begin{aligned} \psi_j^{(V)} &= N^{-1}\phi_{++}^{(V)} - (N-1)^{-1}\left(\phi_{++}^{(V)} - 2\phi_{j+}^{(V)} + \phi_{jj}^{(V)}\right) \\ \psi_j^{(XY)} &= N^{-1}\phi_{++}^{(XY)} - (N-1)^{-1}\left(\phi_{++}^{(XY)} - 2\phi_{j+}^{(XY)} + \phi_{jj}^{(XY)}\right) \end{aligned} \tag{16}$$

The estimates and jackknife pseudovalues of (8)–(16) can all be expressed in terms of the $\phi_{jj}^{(V)}$, $\phi_{j+}^{(V)}$, $\phi_{jj}^{(XY)}$, and $\phi_{j+}^{(XY)}$. Newson (2006) devised an algorithm to calculate these quantities, using binary search trees, that requires an amount of computation time of order $N_{\text{obs}} \log N_{\text{obs}}$, where N_{obs} is the number of observations. `somersd` uses a version of this algorithm unless the user specifies the `notree` option, in which case `somersd` uses a trivial algorithm, which compares all pairs of observations and requires an amount-of-time quadratic in N_{obs}. The difference in performance can be spectacular in large datasets ($N_{\text{obs}} > 1{,}000$).

The parameters we really want to estimate are Kendall's τ_a and/or Somers' D, defined by (7). These formulas are equivalent to the familiar formulas (2) and (3). To estimate them, we use the jackknife method on V and T_{XY} and use appropriate Taylor polynomials. `somersd` calculates correlation measures for one variable X with a set of Y variates $(Y^{(1)}, \ldots, Y^{(p)})$. (The X variate may have a censorship indicator R, and the Y variates may have censorship indicators $(S^{(1)}, \ldots, S^{(p)})$.) It calculates, in the first instance, the covariance matrix for $\widehat{V}$, $\widehat{T}_{XX}$, and $\widehat{T}_{XY^{(i)}}$ for $1 \le i \le p$ by using the jackknife influence matrix Υ, which has N rows labeled by the cluster subscripts, and $p+2$ columns labeled (in Stata fashion) by the names V, X, and $Y^{(i)}$ for $1 \le i \le p$. This matrix is defined by

$$\Upsilon(j,V) = \psi_j^{(V)} - \bar{\psi}^{(V)}, \quad \Upsilon(j,X) = \psi_j^{(XX)} - \bar{\psi}^{(XX)}, \quad \Upsilon\left(j, Y^{(i)}\right) = \psi_j^{(XY^{(i)})} - \bar{\psi}^{(XY^{(i)})}$$

where the quantities

$$\bar{\psi}^{(V)} = N^{-1}\sum_{k=1}^{N}\psi_k^{(V)}, \quad \bar{\psi}^{(XX)} = N^{-1}\sum_{k=1}^{N}\psi_k^{(XX)}, \quad \bar{\psi}^{(XY^{(i)})} = N^{-1}\sum_{k=1}^{N}\psi_k^{(XY^{(i)})}$$

are the mean pseudovalues. (These mean pseudovalues are equal to the corresponding point estimates unless `funtype(vonmises)` is specified, in which case the mean pseudovalue is equal to the corresponding Hoeffding U statistic.) The jackknife covariance matrix is equal to

$$\widehat{C} = \{N(N-1)\}^{-1}\,\Upsilon'\Upsilon$$

The estimates for Kendall's τ_a and Somers' D, for variables Y and X, are defined by

$$\widehat{\tau}_{XY} = \widehat{T}_{XY}/\widehat{V}, \quad \widehat{D}_{YX} = \widehat{T}_{XY}/\widehat{T}_{XX}$$

unless the denominators of these expressions are zero, in which case the numerators must also be zero, and `somersd` therefore sets the estimates and their covariances to zero. If the denominator is nonzero, then the covariance matrix is defined with Taylor polynomials. For Somers' D, we define the $p \times (p+2)$ matrix of estimated derivatives $\widehat{\Gamma}^{(D)}$, whose rows are labeled by the names $Y^{(1)}, \ldots, Y^{(p)}$, and whose columns are labeled by $V, X, Y^{(1)}, \ldots, Y^{(p)}$. This matrix is defined by

$$\widehat{\Gamma}^{(D)}\left(Y^{(i)}, X\right) = \partial\widehat{D}_{Y^{(i)}X}/\partial\widehat{T}_{XX} = -\widehat{T}_{XY^{(i)}}/\widehat{T}^2_{XX}$$

$$\widehat{\Gamma}^{(D)}\left(Y^{(i)}, Y^{(i)}\right) = \partial\widehat{D}_{Y^{(i)}X}/\partial\widehat{T}_{XY^{(i)}} \ = 1/\widehat{T}_{XX}$$

all other entries being 0. For Kendall's τ_a, we define a $(p+1)\times(p+2)$ matrix of estimated derivatives $\widehat{\Gamma}^{(\tau)}$, whose rows are labeled by $X, Y^{(1)}, \ldots, Y^{(p)}$, and whose columns are labeled by $V, X, Y^{(1)}, \ldots, Y^{(p)}$. This matrix is defined by

$$\widehat{\Gamma}^{(\tau)}(X, V) = \partial\widehat{\tau}_{XX}/\partial\widehat{V} = -\widehat{T}_{XX}/\widehat{V}^2$$

$$\widehat{\Gamma}^{(\tau)}(X, X) = \partial\widehat{\tau}_{XX}/\partial\widehat{T}_{XX} = 1/\widehat{V}$$

$$\widehat{\Gamma}^{(\tau)}\left(Y^{(i)}, V\right) = \partial\widehat{\tau}_{XY^{(i)}}/\partial\widehat{V} = -\widehat{T}_{XY^{(i)}}/\widehat{V}^2$$

$$\widehat{\Gamma}^{(\tau)}\left(Y^{(i)}, Y^{(i)}\right) = \partial\widehat{\tau}_{XY^{(i)}}/\partial\widehat{T}_{XY^{(i)}} = 1/\widehat{V}$$

all other entries again being 0. The estimated dispersion matrices of the Somers' D and τ_a estimates are therefore $\widehat{C}^{(D)}$ and $\widehat{C}^{(\tau)}$, respectively, defined by

$$\widehat{C}^{(D)} = \widehat{\Gamma}^{(D)}\,\widehat{C}\,\widehat{\Gamma}^{(D)\,\prime}, \quad \widehat{C}^{(\tau)} = \widehat{\Gamma}^{(\tau)}\,\widehat{C}\,\widehat{\Gamma}^{(\tau)\,\prime} \tag{17}$$

4.1 Transformations

The `transf()` option offers a choice of transformations. Since these are available both for Somers' D and for Kendall's τ_a, we will denote the original estimate as θ (which can stand for D or τ) and the transformed estimate as ζ. They are summarized in table 1, together with their derivatives, $d\zeta/d\theta$.

Table 1: Transformations provided by the `transf()` option of `somersd`

`transf()`	Transform name	$\zeta(\theta)$	$d\zeta/d\theta$
`iden`	Untransformed	θ	1
`z`	Fisher's z	$\text{arctanh}\,\theta =$ $\frac{1}{2}\log\{(1+\theta)/(1-\theta)\}$	$\left(1-\theta^2\right)^{-1}$
`asin`	Daniels' arcsine	$\arcsin\theta$	$\left(1-\theta^2\right)^{-1/2}$
`rho`	Greiner's ρ	$\sin(\frac{\pi}{2}\theta)$	$\frac{\pi}{2}\cos(\frac{\pi}{2}\theta)$
`zrho`	Greiner's ρ (z transformed)	$\text{arctanh}\sin(\frac{\pi}{2}\theta)$	$\frac{\pi}{2}\cos(\frac{\pi}{2}\theta)\{1-\sin(\frac{\pi}{2}\theta)^2\}^{-1}$
`c`	Harrell's c	$(\theta+1)/2$	$1/2$

(All these expressions are defined for $\theta = 0$, but some are undefined for $\theta = 1$ or $\theta = -1$, and in those cases `somersd` enters a substitute θ argument very close to 1 or -1.) If `transf()` is specified, then `somersd` displays and saves the transformed estimates and their estimated covariance, instead of the untransformed versions. If $\widehat{C}^{(\theta)}$ is the covariance matrix for the untransformed estimates given by (17), and $\widehat{\Gamma}^{(\zeta)}$ is the diagonal matrix whose diagonal entries are the $d\zeta/d\theta$ estimates specified in the table, then the transformed parameter and its covariance matrix are

$$\widehat{\zeta} = \zeta(\widehat{\theta}), \quad \widehat{C}^{(\zeta)} = \widehat{\Gamma}^{(\zeta)}\,\widehat{C}^{(\theta)}\,\widehat{\Gamma}^{(\zeta)\,\prime}$$

Fisher's z transform was originally recommended for the Pearson correlation coefficient by Fisher (1921) (see also Gayen 1951), but Edwardes (1995) recommended it specifically for Somers' D on the basis of simulation studies. Daniels and Kendall (1947) suggested Daniels' arcsine as a normalizing transform. If `transf(z)` or `transf(asin)` is specified, then `somersd` prints asymmetric confidence intervals for the untransformed D or τ_a parameters, calculated from symmetric confidence intervals for the transformed parameters using the inverse function $\theta(\zeta)$. (This feature corresponds to the `eform` option of other estimation commands.) Greiner's ρ (Kendall and Gibbons 1990) is designed to estimate the Pearson correlation coefficient corresponding to the measured τ_a. If `transf(zrho)` is specified, then `somersd` prints asymmetric confidence intervals for the untransformed Greiner's ρ, using the inverse z transform on symmetric confidence intervals for the z-transformed Greiner's ρ. Harrell's c is usually a reparameterization of Somers' D and is recommended in Harrell et al. (1982) and Harrell, Lee, and Mark (1996) as a general measure of the predictive power of a prognostic score arising from a medical test.

5 Examples

These examples overlap with those in `somersd.pdf`, distributed with the `somersd` package. This selection concentrates on extensions to Somers' D not previously available.

5.1 Extensions to paired data

In [R] **signrank**, the paired Wilcoxon and sign tests are demonstrated on a dataset with 1 observation for each of 12 cars and variables `mpg1` and `mpg2`, representing miles per gallon for the car when tested with untreated and treated fuel, respectively. Here we use `somersd` on the same data to produce confidence intervals corresponding to the two rank tests for paired data, both of which test hypotheses about versions of Somers' D.

For the paired Wilcoxon test carried out by `signrank`, the underlying parameter is D_{YX}, where Y is the absolute difference between miles per gallon observed under the two fuel treatments and X is the sign of the difference. We are therefore testing whether positive differences between `mpg1` and `mpg2` tend to have higher values than negative differences. We first do this with `signrank`, which produces only a p-value, and then do this with `somersd`, which gives a confidence interval:

```
. use http://www.stata-press.com/data/r9/fuel
. signrank mpg2=mpg1
Wilcoxon signed-rank test
        sign |      obs   sum ranks    expected
-------------+---------------------------------
    positive |        8        63.5        38.5
    negative |        3        13.5        38.5
        zero |        1           1           1
-------------+---------------------------------
         all |       12          78          78
unadjusted variance      162.50
adjustment for ties       -1.62
adjustment for zeros      -0.25
                     ----------
adjusted variance        160.62
Ho: mpg2 = mpg1
             z =   1.973
    Prob > |z| =   0.0485
. gen signdiff=sign(mpg2-mpg1)
. gen absdiff=abs(mpg2-mpg1)
```

(*Continued on next page*)

```
. somersd signdiff absdiff if absdiff!=0, transf(z)
Somers' D with variable: signdiff
Transformation: Fisher's z
Valid observations: 11

Symmetric 95% CI for transformed Somers' D
------------------------------------------------------------------------------
             |              Jackknife
    signdiff |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     absdiff |   .7331685   .4568681     1.60   0.109    -.1622765    1.628614
------------------------------------------------------------------------------

Asymmetric 95% CI for untransformed Somers' D
                Somers_D    Minimum    Maximum
     absdiff        .625  -.1608669  .92586389

. drop signdiff absdiff
```

`signrank` gives a p-value of 0.0485. `somersd` produces a (slightly) higher p-value but also produces confidence intervals for the z-transformed and untransformed Somers' D. In our sample of 12 cars, if we choose a positive treated–untreated difference and a negative treated–untreated difference at random, then the positive difference is 62.5% more likely to be the larger of the two than to be the smaller of the two. And, in the population of cars from which this sample was taken, we are 95% confident that this difference is between 16% less likely and 93% more likely.

The sign test is based on a within-cluster Somers' D, where the clusters are cars and the observations are performance tests (two on each car). Here the underlying parameter is D_{YX}, where Y is miles per gallon achieved by that car on that test and X is fuel treatment status (untreated or treated). This time, after calling `signtest`, we expand the dataset with 1 observation per car, using `reshape`, to produce a new dataset with 1 observation per test and therefore 2 observations per car. The dataset also contains variables `carseq` containing the car sequence number, `fueltrea` equal to 1 for untreated fuel and 2 for treated fuel, and `mpg` containing miles per gallon. We use `somersd`, with the options `cluster(carseq) funtype(wcluster)`, to produce a confidence interval for Somers' D:

```
. signtest mpg2=mpg1
Sign test
        sign |    observed    expected
-------------+------------------------
    positive |           8         5.5
    negative |           3         5.5
        zero |           1           1
-------------+------------------------
         all |          12          12
One-sided tests:
  Ho: median of mpg2 - mpg1 = 0 vs.
  Ha: median of mpg2 - mpg1 > 0
      Pr(#positive >= 8) =
         Binomial(n = 11, x >= 8, p = 0.5) =  0.1133
  Ho: median of mpg2 - mpg1 = 0 vs.
  Ha: median of mpg2 - mpg1 < 0
      Pr(#negative >= 3) =
         Binomial(n = 11, x >= 3, p = 0.5) =  0.9673
Two-sided test:
  Ho: median of mpg2 - mpg1 = 0 vs.
  Ha: median of mpg2 - mpg1 != 0
      Pr(#positive >= 8 or #negative >= 8) =
         min(1, 2*Binomial(n = 11, x >= 8, p = 0.5)) =  0.2266
. preserve
. gen carseq=_n
. reshape long mpg, i(carseq) j(fueltrea)
(note: j = 1 2)
Data                               wide   ->   long
-----------------------------------------------------------------------------
Number of obs.                       12   ->      24
Number of variables                   3   ->       3
j variable (2 values)                     ->   fueltrea
xij variables:
                              mpg1 mpg2   ->   mpg
-----------------------------------------------------------------------------
. somersd fueltrea mpg, transf(z) cluster(carseq) funtype(wcluster)
Within-cluster Somers' D with variable: fueltrea
Transformation: Fisher's z
Valid observations: 24
Number of clusters: 12
Symmetric 95% CI for transformed Somers' D
                                (Std. Err. adjusted for 12 clusters in carseq)
------------------------------------------------------------------------------
             |              Jackknife
    fueltrea |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         mpg |   .4436516   .3145066     1.41   0.158    -.1727701    1.060073
------------------------------------------------------------------------------
Asymmetric 95% CI for untransformed Somers' D
                Somers_D     Minimum     Maximum
         mpg   .41666667  -.17107135   .78569191
. restore
```

This time `signtest` produces a p-value of 0.2266. `somersd` produces a p-value of 0.158 and confidence intervals for the z-transformed and untransformed Somers' D. We see that, in this sample of 12 cars with two tests each, if the same car is tested with untreated and treated fuel, then it is 42% more likely to travel more miles per gallon with the treated fuel than with the untreated fuel. And, in the population of cars from which this sample was drawn, a car is between 17% less likely and 79% more likely to travel farther per gallon on the treated fuel than on the untreated fuel. Therefore, the high p-value definitely does not indicate proof of the null hypothesis that a car is equally likely to travel farther on treated or untreated fuel.

The within-cluster Somers' D tested by the sign test can easily be generalized to cases where each car is tested more than one time with each type of fuel.

5.2 Extensions to survival data

Here I demonstrate the `cenind()` option with a simple set of survival data distributed by Stata Press, with 1 observation per subject in a drug trial and data on treatment, age, and survival time. We load the data, tabulate the treatment variable `drug`, and finally define the new variables `youth` (representing number of years to the subject's 100th birthday) and `censind` (a censorship indicator equal to 0 for subjects who died and to 1 for subjects whose survival time is right censored). We also use `xtile` to split the sample into three age tertiles.

```
. use http://www.stata-press.com/data/r9/drugtr, clear
(Patient Survival in Drug Trial)

. tab drug, m

  Drug type |
(0=placebo) |      Freq.     Percent        Cum.
------------+-----------------------------------
          0 |         20       41.67       41.67
          1 |         28       58.33      100.00
------------+-----------------------------------
      Total |         48      100.00

. gen youth=100-age

. gen byte censind=1-died

. tab died censind, m

     1 if |
  patient |        censind
     died |         0          1 |     Total
----------+----------------------+----------
        0 |         0         17 |        17 
        1 |        31          0 |        31 
----------+----------------------+----------
    Total |        31         17 |        48 

. xtile agegp=age, n(3)
. tab agegp, m

3 quantiles |
     of age |      Freq.     Percent        Cum.
------------+-----------------------------------
          1 |         18       37.50       37.50
          2 |         16       33.33       70.83
          3 |         14       29.17      100.00
------------+-----------------------------------
      Total |         48      100.00
```

The Wilcoxon–Breslow–Gehan test is demonstrated using Stata in [ST] **sts test**. It tests the hypothesis of a zero value of the Somers' *D* of survival (as the *Y* variable) with respect to membership of a particular group (as the *X* variable). Using `somersd`, we can improve on this test by defining a confidence interval for this Somers' *D* parameter:

```
. sts test drug, wilcoxon

         failure _d:  died
   analysis time _t:  studytime

Wilcoxon (Breslow) test for equality of survivor functions

      |  Events         Events        Sum of
drug  |  observed       expected       ranks
------+--------------------------------------
0     |        19           7.25         385
1     |        12          23.75        -385
------+--------------------------------------
Total |        31          31.00           0

            chi2(1) =      22.61
            Pr>chi2 =     0.0000

. somersd drug studytime, tr(z) cenind(0 censind)
Somers' D with variable: drug
Transformation: Fisher's z
Valid observations: 48

Symmetric 95% CI for transformed Somers' D
------------------------------------------------------------------------------
             |              Jackknife
        drug |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
   studytime |   .8297787   .1935732     4.29   0.000     .4503821    1.209175
------------------------------------------------------------------------------

Asymmetric 95% CI for untransformed Somers' D
               Somers_D    Minimum    Maximum
   studytime  .68035714  .42221306  .83643191
```

We see (from the Wilcoxon test) that the treated group has fewer deaths, and that the placebo group has more deaths, than we would expect by chance, assuming population survival distributions to be the same in the two groups. We also see (from the confidence interval for the untransformed Somers' *D*) that, if we sample a subject at random from each of the two subpopulations (treated and placebo), then the event that the treated subject survives the placebo subject is 42%–84% more probable than the event that the placebo subject survives the treated subject. We can also stratify Somers' *D* by age tertile:

```
. somersd drug studytime, tr(z) cenind(0 censind) wstrata(agegp)
Somers' D with variable: drug
Transformation: Fisher's z
Within strata defined by: agegp
Valid observations: 48

Symmetric 95% CI for transformed Somers' D

                                Jackknife
        drug        Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]

   studytime     .9729551    .2404965    4.05   0.000     .5015905     1.44432

Asymmetric 95% CI for untransformed Somers' D
                Somers_D    Minimum    Maximum
   studytime         .75  .46336709  .89456394
```

We see that, if we sample a subject at random from the same age tertile in both treatment groups (treated and placebo), then it is 46%–89% more likely that the treated subject survives the untreated subject than vice versa.

The Gehan–Breslow–Wilcoxon Somers' D is an example of D_{YX} interpreted as a treatment effect. However, we may also estimate D_{XY} (or the corresponding Harrell's c) as a predictor performance indicator. For instance, we can compare treatment and youth as predictors of survival by using `somersd` and `lincom`:

```
. somersd studytime drug youth, tr(c) cenind(censind)
Somers' D with variable: studytime
Transformation: Harrell's c
Valid observations: 48

Symmetric 95% CI for Harrell's c

                                Jackknife
   studytime        Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]

        drug     .7275986    .0367931   19.78   0.000     .6554855    .7997117
       youth     .6415771    .0528314   12.14   0.000     .5380295    .7451246

. lincom drug-youth
 ( 1)  drug - youth = 0

   studytime        Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]

         (1)     .0860215    .0618354    1.39   0.164    -.0351736    .2072166
```

We see that active drug treatment and youth are both positive survival indicators, as they both have values of Harrell's c greater than 0.5. However, when we use `lincom` to estimate the difference between the two Harrell's c parameters (equal to half the difference between the corresponding Somers' D parameters), we find that the confidence interval for the difference includes zero. From this difference alone, we cannot state that the active treatment is a more or less positive predictor than being young. However, we can use the `wstrata()` option to estimate pooled, stratified Harrell's c values for youth and treatment (based only on comparisons within age tertiles) and their difference:

```
. somersd studytime drug youth, tr(c) cenind(censind) wstrata(agegp)
Somers' D with variable: studytime
Transformation: Harrell's c
Within strata defined by: agegp
Valid observations: 48

Symmetric 95% CI for Harrell's c

------------------------------------------------------------------------------
             |              Jackknife
   studytime |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        drug |   .7630597   .0398266    19.16   0.000      .685001    .8411184
       youth |   .5559701   .0607348     9.15   0.000     .4369321    .6750082
------------------------------------------------------------------------------

. lincom drug-youth

 ( 1)  drug - youth = 0

------------------------------------------------------------------------------
   studytime |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         (1) |   .2070896   .0660029     3.14   0.002     .0777262    .3364529
------------------------------------------------------------------------------
```

This time, we see that youth is a less impressive predictor of survival within age tertiles (as the confidence interval for Harrell's c contains 0.5) and is a worse predictor than treatment when predicting survival between subjects in the same age tertile. We can therefore conclude (strongly) that treatment has an effect that is not entirely caused by confounding by age.

In this analysis, there is only one confounder. There are often many confounders in observational studies in real life, making stratified analyses harder. However, a possible solution might be to define a propensity score, measuring proneness to allocation to a treatment and dependent on all the confounders, and to use `xtile` on the propensity score to define a propensity group variable, which `somersd` can use as the `wstrata()` option. The seminal paper on propensity scores is Rosenbaum and Rubin (1983), but a good place to start a literature search now might be Imai and van Dyk (2004).

5.3 Scenario effects: The Gini coefficient

Econometricians use the Gini coefficient of inequality (Cowell 1995; Jenkins 1999) as a measure of the inequality of a distribution of incomes, wealth, or other assets in a population, on a scale from zero (when everybody has an equal share) to one (when one person has everything). It is traditionally understood by reference to the Lorenz curve, which is the set of (X, Y) points on the unit square such that the richest $100Y$ percent of the population have $100X$ percent of the income (or wealth). The Lorenz curve is therefore an example of a probability–probability plot, as is the ROC curve (Hanley and McNeil 1982). The Gini coefficient is equal to the difference between the area above and below the Lorenz curve. Gini also invented several other coefficients, which are also referred to in various contexts as "the Gini coefficient" and are discussed in Goodman and Kruskal (1959).

The Gini coefficient of inequality is a special case of Somers’ D. Imagine that two lotteries are organized in a population. In the first lottery each member of the population has one ticket, whereas in the second lottery each individual buys a number of tickets proportional to that individual’s income. The first lottery is equivalent to sampling uniformly from the y-axis of the Lorenz plot, whereas the second lottery is equivalent to sampling uniformly from the x-axis of the Lorenz plot. The region above the Lorenz curve corresponds to the event that the second lottery winner is a higher earner than the first, whereas the region below the Lorenz curve corresponds to the event that the first lottery winner is a higher earner than the second. Therefore, the Gini coefficient is a clustered Somers’ D, where the clusters are individuals in the population, the observations are combinations of individual and lottery (first or second), the Y variate is income, the X variate is lottery sequence (1 or 2), and the importance weights are equal for all individuals in the first lottery and proportional to income for all individuals in the second lottery.

I can illustrate this principle with the `womenwage` dataset, distributed by Stata Press and used in [R] **intreg**. We `preserve` the data and use the `expgen` package (an extended version of `expand` downloadable from SSC) to replace each observation in the original dataset (containing 1 observation per woman) with 2 observations (one per woman per lottery). The new dataset is indexed by the variables `womanid` (denoting sequence number of the woman) and `lotseq` (denoting sequence number of the lottery). We create an importance variable, `pwt`, containing probability weights equal for all women in the first lottery and equal to a woman’s wage (to the nearest kilodollar) in the second lottery. We then use `somersd`, using the normalizing and/or variance-stabilizing z transformation, before restoring the old dataset:

```
. use http://www.stata-press.com/data/r9/womenwage, clear
(Wages of women)

. preserve

. expgen = 2, oldseq(womanid) copyseq(lotseq)

. lab var lotseq "Lottery sequence number"

. gen pwt = (lotseq==1) + wage*(lotseq==2)

. lab var pwt "Probability weight"

. somersd lotseq wage [pwei=pwt], cluster(womanid) funtype(vonmises) tr(z)
Von Mises Somers’ D with variable: lotseq
Transformation: Fisher’s z
Valid observations: 976
Number of clusters: 488

Symmetric 95% CI for transformed Somers’ D
                              (Std. Err. adjusted for 488 clusters in womanid)
------------------------------------------------------------------------------
             |              Jackknife
      lotseq |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        wage |   .2875044   .0114695    25.07   0.000     .2650246    .3099843
------------------------------------------------------------------------------

Asymmetric 95% CI for untransformed Somers’ D
                Somers_D    Minimum    Maximum
        wage   .27983629  .25898919  .30042278

. restore
```

We see that, if the women in this dataset organized two lotteries amongst themselves, and each woman bought one ticket in the first lottery and a number of tickets worth a constant fraction of her wages in the second lottery, then the second lottery winner would be 27.98% more likely than the first lottery winner to be the higher earner. And, if the same lotteries were organized in the population from which these women were sampled, then the difference would probably be between 25.90% and 30.04%. The option `funtype(vonmises)` is necessary because there is a small but nonzero probability that, by chance, the same woman will win both lotteries, although the other women in the sample will probably not believe this if it happens.

The Gini coefficient represents a type of treatment-effect Somers' D, which we might call a scenario-effect Somers' D, where the treatment groups are two scenarios, imagined to happen to the same population. Another example of a scenario-effect Somers' D is the population-attributable risk (Gordis 2000), defined by epidemiologists as the difference between the risk of a disease in the population we can observe and the risk of disease that would be observed in the same population in an alternative scenario. In the alternative scenario, we can eliminate an exposure, which is assumed to have a causal effect on the risk of a disease. To estimate this effect by using `somersd`, we would use `funtype(vonmises)` and expand each individual in a sample into two "scenario individuals", corresponding to the same individual under the two scenarios, and assign a zero importance weight to exposed individuals under the second scenario. Sampling probability weights might be used to standardize the two scenarios to a common distribution of a stratifying variable, defined by age and/or a propensity score for the exposure.

6 Summary

Somers' D is an ordinal association measure. It includes, as special cases, a large family of parameters, which underlie rank or so-called nonparametric methods and are interpretable as differences between proportions. The Stata 9 version of the `somersd` package has added the options `cenind()`, `cfweight()`, `funtype()`, `wstrata()`, and `bstrata()`. These additions allow the user to estimate these special cases, most of which could not be estimated by the previous Stata 6 version. These differences can be adjusted for confounding variables, which is not usually easy using rank-based methods. We may still need to use regression methods to define a propensity score.

Until now, `somersd` has been limited in its ability to calculate confidence intervals for rank statistics not interpreted as differences between proportions. Such rank statistics include the Hodges–Lehmann median difference and the Theil median slope, discussed in section 6 of Newson (2002), which are expressed in units of a Y variable, or in Y units per X unit, and are both defined in terms of Somers' D. The present `somersd` package includes `cendif`, which calculates (albeit inefficiently) a robust confidence interval for the unstratified Hodges–Lehmann median difference and was introduced in Newson (2000b). Work is in progress to address these major limitations of the `somersd` package.

7 References

Arvesen, J. N. 1969. Jackknifing U-statistics. *Annals of Mathematical Statistics* 40: 2076–2100.

Breslow, N. E. 1970. A generalized Kruskal–Wallis test for comparing k samples subject to unequal patterns of censorship. *Biometrika* 57: 579–594.

Cowell, F. A. 1995. *Measuring Inequality*. 2nd ed. Hemel Hempstead, UK: Harvester–Wheatsheaf.

Daniels, H. E., and M. G. Kendall. 1947. The significance of rank correlation where parental correlation exists. *Biometrika* 34: 197–208.

Edwardes, M. D. 1995. A confidence interval for $\Pr(X < Y) - \Pr(X > Y)$ estimated from simple cluster samples. *Biometrics* 51: 571–578.

Fisher, R. A. 1921. On the "probable error" of a coefficient of correlation deduced from a small sample. *Metron* 1: 3–32.

Gayen, A. E. 1951. The frequency distribution of the product-moment correlation coefficient in random samples of any size drawn from non-normal universes. *Biometrika* 38: 219–247.

Gehan, E. A. 1965. A generalized Wilcoxon test for comparing arbitrarily single-censored samples. *Biometrika* 52: 203–223.

Goodman, L. A., and W. H. Kruskal. 1959. Measures of association for cross-classifications. II: Further discussion and references. *Journal of the American Statistical Association* 54: 123–163.

Gordis, L. 2000. *Epidemiology*. 2nd ed. Philadelphia: Saunders.

Hanley, J. A., and B. J. McNeil. 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* 143: 29–36.

Harrell, F. E., Jr., R. M. Califf, D. B. Pryor, K. L. Lee, and R. A. Rosati. 1982. Evaluating the yield of medical tests. *Journal of the American Medical Association* 247: 2543–2546.

Harrell, F. E., Jr., K. L. Lee, and D. B. Mark. 1996. Multivariable prognostic models: Issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors. *Statistics in Medicine* 15: 361–387.

Hoeffding, W. 1948. A class of statistics with asymptotically normal distribution. *Annals of Mathematical Statistics* 19: 293–325.

Imai, K., and D. A. van Dyk. 2004. Causal inference with general treatment regimes: Generalizing the propensity score. *Journal of the American Statistical Association* 99: 854–866.

Jenkins, S. P. 1999. sg104: Analysis of income distributions. *Stata Technical Bulletin* 48: 4–18. Reprinted in *Stata Technical Bulletin Reprints*, vol. 8, pp. 243–260. College Station, TX: Stata Press.

Kendall, M. G., and J. D. Gibbons. 1990. *Rank Correlation Methods*. 5th ed. London: Arnold.

Kerridge, D. 1975. The interpretation of rank correlations. *Applied Statistics* 24: 257–258.

Kirkwood, B. R., and J. A. C. Sterne. 2003. *Essential Medical Statistics*. 2nd ed. Oxford: Blackwell Science.

Lee, A. J. 1990. *U-statistics: Theory and Practice*. New York: Marcel Dekker.

von Mises, R. 1947. On the asymptotic distribution of differentiable statistical functions. *Annals of Mathematical Statistics* 18: 309–348.

Newson, R. 2000a. snp15: somersd—Confidence intervals for nonparametric statistics and their differences. *Stata Technical Bulletin* 55: 47–55. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 312–322. College Station, TX: Stata Press.

———. 2000b. snp16: Robust confidence intervals for median and other percentile differences between groups, *Stata Technical Bulletin* 58: 30–35. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 324–331. College Station, TX: Stata Press.

———. 2002. Parameters behind "nonparametric" statistics: Kendall's tau, Somers' D and median differences. *Stata Journal* 2: 45–64.

———. 2006. Efficient calculation of jackknife confidence intervals for rank statistics. *Journal of Statistical Software* 15: 1–10.

Riedwyl, H. 1988. V-statistics. In *Encyclopedia of Statistical Sciences*, ed. S. Kotz and N. L. Johnson, vol. 9, 509–512. New York: Wiley.

Rosenbaum, P. R., and D. B. Rubin. 1983. The central role of the propensity score in observational studies for causal effects. *Biometrika* 70: 41–55.

Serfling, R. J. 1980. *Approximation Theorems of Mathematical Statistics*. New York: Wiley.

———. 1988. U-statistics. In *Encyclopedia of Statistical Sciences*, ed. S. Kotz and N. L. Johnson, vol. 9, 436–444. New York: Wiley.

Somers, R. H. 1962. A new asymmetric measure of association for ordinal variables. *American Sociological Review* 27: 799–811.

Wolfe, D. A., and R. V. Hogg. 1971. On constructing statistics and reporting data. *American Statistician* 25: 27–30.

About the author

Roger Newson is a lecturer in medical statistics at Imperial College London, London, UK, working principally in asthma research. He wrote the `somersd` and `expgen` packages.