The Stata Journal (2010)
10, Number 3, pp. 339–358

Comparing the predictive powers of survival models using Harrell's C or Somers' D

Roger B. Newson
National Heart and Lung Institute
Imperial College London
London, UK
r.newson@imperial.ac.uk

Abstract. Medical researchers frequently make statements that one model predicts survival better than another, and they are frequently challenged to provide rigorous statistical justification for those statements. Stata provides the `estat concordance` command to calculate the rank parameters Harrell's C and Somers' D as measures of the ordinal predictive power of a model. However, no confidence limits or p-values are provided to compare the predictive power of distinct models. The `somersd` package, downloadable from Statistical Software Components, can provide such confidence intervals, but they should not be taken seriously if they are calculated in the dataset in which the model was fit. Methods are demonstrated for fitting alternative models to a training set of data, and then measuring and comparing their predictive powers by using out-of-sample prediction and `somersd` in a test set to produce statistically sensible confidence intervals and p-values for the differences between the predictive powers of different models.

Keywords: st0198, somersd, stcox, estat concordance, streg, predict, survival, model validation, prediction, concordance, rank methods, Harrell's C, Somers' D

1 Introduction

Harrell's C and the equivalent parameter Somers' D were proposed as measures of the general predictive power of a general regression model by Harrell et al. (1982) and Harrell, Lee, and Mark (1996), who focused attention on the case of a survival model with a possibly right-censored outcome, which was interpreted as a lifetime. In the case of a Cox proportional hazards regression model, both parameters are output by the Stata postestimation command `estat concordance` (see [ST] **stcox postestimation**).[1] However, because Harrell's C and Somers' D are rank parameters, they are equally valid as measures of the predictive power of any model in which the scalar outcome Y is at least ordinal (with or without censorship), and in which the conditional distribution of the outcome, given the predictor variables, is governed by a scalar function of the predictor variables and the parameters, such as the hazard ratio in a Cox regression or the linear predictor in a generalized linear model. If the assumptions of the model are true, then such a scalar predictive score plays the role of a balancing score as defined by Rosenbaum and Rubin (1983).

1. As of Stata 11.1, `estat concordance` provides two concordance measures: Harrell's C and Gönen and Heller's K. Harrell's C is computed by default or if `harrell` is specified.

Harrell's C and Somers' D are members of the Kendall family of rank parameters. The family history can be summarized as follows: Kendall's τ_a begat Somers' D begat Theil–Sen percentile slopes. This family is implemented in Stata by using the `somersd` package, which can be downloaded from Statistical Software Components. An overview of the parameter family is given in Newson (2002), and the methods and formulas are given in detail in Newson (2006a,b,c).

Parameters in this family are defined by assuming the existence of a population of bivariate data pairs of the form (X_i, Y_i) and a sampling scheme for sampling pairs of pairs $\{(X_i, Y_i), (X_j, Y_j)\}$ from that population. A pair of pairs is said to be concordant if the larger of the X values is paired with the larger of the Y values, and a pair is said to be discordant if the larger of the X values is paired with the smaller of the Y values. Kendall's τ_a is the difference between the probability of concordance and the probability of discordance. Somers' $D(X \,|\, Y)$ is the difference between the corresponding conditional probabilities, assuming that the two Y values can be ordered. Harrell's $C(X \,|\, Y)$ is defined as $\{D(X \,|\, Y) + 1\}/2$ and is equal to the conditional probability of concordance plus half the conditional probability that the data pairs are neither concordant nor discordant, assuming that the two Y values can be ordered. In the case where Y is an outcome to be predicted by a multivariate model with a scalar predictive score, there is an underlying population of multivariate data points $(Y_i, V_{i1}, \ldots, V_{ik})$ where the V_{ih} are predictive covariates and the role of the X_i is played by the scalar predictive score $\eta(V_{i1}, \ldots, V_{ik})$. In this case, the Somers' D and Harrell's C parameters can be denoted as $D\{\eta(V_1, \ldots, V_k) \,|\, Y\}$ and $C\{\eta(V_1, \ldots, V_k) \,|\, Y\}$, respectively. If the model is a survival model, then the Y values are lifetimes, and there is the possibility that one or both of a pair of Y values may be censored, which sometimes implies that they cannot be ordered.

We often want to compare the predictive powers of alternative predictors of the same outcome Y. Newson (2002, 2006b) argues that if there is an underlying population of trivariate data points (W_i, X_i, Y_i) and if any positive association between the Y_i and the X_i is caused by a positive association of both of these variables with the W_i, then we must have the inequality $D(X \,|\, Y) - D(W \,|\, Y) \leq 0$ or, equivalently, $C(X \,|\, Y) - C(W \,|\, Y) = \{D(X \,|\, Y) - D(W \,|\, Y)\}/2 \leq 0$. This inequality still holds if the Y variable may be censored, but not if the W or X variable may be censored. This implies that if we have multiple alternative positive predictors of the same outcome, such as alternative predictive scores from alternative multivariate models, then it may be useful to calculate confidence intervals for the differences between the Somers' D or Harrell's C parameters of these predictors with respect to the outcome, and then to make statements regarding which predictors may or may not be secondary to which other predictors. In Stata, this can be done by using `lincom` after the `somersd` command, as demonstrated in section 4.1 of Newson (2002).

Medical researchers frequently make statements that one model predicts survival better than another. Statistical referees acting for medical journals frequently challenge the researchers to provide rigorous statistical justification for these statements. The Stata postestimation command `estat concordance` provides estimates of Harrell's C and Somers' D but provides no confidence limits for these, nor any confidence limits or

p-values for the differences between the values of these rank parameters from different models. This is the case for good reason: confidence-interval formulas do not protect the user for finding a model in the same data in which its parameters are then estimated. Used sequentially, the `somersd` and `lincom` commands provide confidence limits and p-values for differences between the Somers' D or Harrell's C parameters between different predictors. However, not all medical researchers know how to calculate a confidence interval (CI) when the predictors are scalar predictive scores from models, and fewer still know how to do so in such a way that the confidence limits can be taken seriously. In this article, I aim to explain how medical researchers can calculate CIs and preempt possible queries that may arise in the process.

The remainder of this article is divided into four sections. Section 2 addresses the queries that commonly arise when users try to duplicate the results of `estat concordance` using `somersd`. Section 3 describes the method of splitting the data into a training set (to which models are fit) and a test set (in which their predictive powers are measured). Section 4 describes the extension to non-Cox survival models, such as those described in [ST] **streg**. Finally, section 5 briefly explains how the methods can be extended even further.

2 The Cox model: somersd versus estat concordance

I will demonstrate the principles using the Cox proportional hazards model, which is implemented in Stata using the `stcox` command (see [ST] **stcox**). I also use the Stanford drug-trial dataset, which is used for the examples in [ST] **stcox postestimation**.

Before I raise the issue of confidence limits, we need to see how `somersd` can produce the same estimates as `estat concordance`. This is done using `predict` after the survival estimation command to define the predictive score, and then using `somersd` to measure the association of the predictive score with the lifetime. Users who attempt to use `somersd` to duplicate the estimates of `estat concordance` may face confusion caused by these three issues:

1. The `predict` command, used after `stcox`, by default produces a negative prediction score, in contrast to the positive prediction score produced by using `predict` after most estimation commands.
2. The default coding of a censorship status variable for `stcox` is different from the coding of a censorship status variable for `somersd`.
3. The treatment of tied failure times by `estat concordance` is different from that used by `somersd`.

There are solutions to all of these problems, and I will demonstrate them, enabling users to use `somersd` and `estat concordance` as checks on one another.

Let's start the demonstration by inputting the Stanford drug-trial data, fitting a Cox model, and calling `estat concordance`:

```
. use http://www.stata-press.com/data/r11/drugtr
(Patient Survival in Drug Trial)
. stset
-> stset studytime, failure(died)

     failure event:  died != 0 & died < .
obs. time interval:  (0, studytime]
 exit on or before:  failure
------------------------------------------------------------------------------
       48  total obs.
        0  exclusions
------------------------------------------------------------------------------
       48  obs. remaining, representing
       31  failures in single record/single failure data
      744  total analysis time at risk, at risk from t =         0
                            earliest observed entry t =         0
                                 last observed exit t =        39

. stcox drug age
        failure _d:  died
  analysis time _t:  studytime
Iteration 0:   log likelihood = -99.911448
Iteration 1:   log likelihood = -83.551879
Iteration 2:   log likelihood = -83.324009
Iteration 3:   log likelihood = -83.323546
Refining estimates:
Iteration 0:   log likelihood = -83.323546
Cox regression -- Breslow method for ties
No. of subjects =           48                     Number of obs   =        48
No. of failures =           31
Time at risk    =          744
                                                   LR chi2(2)      =     33.18
Log likelihood  =   -83.323546                     Prob > chi2     =    0.0000
------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        drug |   .1048772   .0477017    -4.96   0.000     .0430057    .2557622
         age |   1.120325   .0417711     3.05   0.002     1.041375     1.20526
------------------------------------------------------------------------------
. estat concordance
  Harrell´s C concordance statistic
        failure _d:  died
   analysis time _t:  studytime
  Number of subjects (N)               =       48
  Number of comparison pairs (P)       =      849
  Number of orderings as expected (E)  =      679
  Number of tied predictions (T)       =       15
         Harrell´s C = (E + T/2) / P =    .8086
                        Somers´ D =    .6172
```

The `stset` command shows us that the input dataset has already been set up as a survival-time dataset that includes one observation per drug-trial subject as well as data on survival time and termination modes, among other things (see [ST] **stset**). The Cox model contains two predictive covariates, `age` (subject age in years) and `drug` (indicating treatment group, with a value of 0 for placebo and a value of 1 for the

drug being tested). We then see that, according to `estat concordance`, Harrell's C is 0.8086 and Somers' D is 0.6172. The Somers' D implies that when one of two subjects is observed to survive another, the model predicts that the survivor is 61.72% more likely to have a lower hazard ratio than the nonsurvivor. The Harrell's C is the probability that the survivor has the lower hazard ratio plus half the (possibly negligible) probability that the two subjects have equal hazard ratios, and this sum is 80.86% on a percentage scale.

We will now see how to duplicate these estimates by using `predict` and `somersd`. We start by defining a negative predictor of lifetime by using `predict` to calculate a hazard ratio. We then derive an inverse hazard ratio, which we expect to be a positive predictor of lifetime:

```
. predict hr
(option hr assumed; relative hazard)
. generate invhr=1/hr
```

This strategy addresses the first of the three sources of confusion mentioned before.

Addressing the second source of confusion, we need to define a censorship indicator for input to the `somersd` command. The `somersd` command has a `cenind()` option that requires a list of censorship indicators. These censorship indicators are allocated one-to-one to the corresponding variables of the variable list input to `somersd` and must be either variable names or zeros (implying a censorship indicator variable whose values are all zero). Censorship indicator variables for `somersd` are positive in observations where the corresponding input variable value is right-censored (or known to be equal to or greater than its stated value), are negative in observations where the corresponding input variable value is left-censored (or known to be equal to or less than its stated value), and are zero in observations where the corresponding input variable value is uncensored (or known to be equal to its stated value). If the list of censorship indicators is shorter than the input variable list, then the list of censorship indicators is extended on the right with zeros, implying that the variables without censorship indicators are uncensored.

This coding scheme is not the same as that for the censorship indicator variable `_d` that is created by the `stset` command, which is 1 in observations where the corresponding lifetime is uncensored and is 0 in observations where the corresponding lifetime is right-censored.

To convert an `stset` censorship indicator variable to a `somersd` censorship indicator variable, we use the command

```
. generate censind=1-_d if _st==1
```

This command creates a new variable, `censind`, which assumes the following values: missing in observations excluded from the survival sample, as indicated by the variable `_st` created by `stset`; 1 in observations with right-censored lifetimes (where `_d` is 0); and 0 in observations with uncensored lifetimes (where `_d` is 1).

We can now use `somersd` to calculate Harrell's C and Somers' D, using the `transf(c)` option for Harrell's C and the `transf(z)` option (indicating the normalizing and variance-stabilizing Fisher's z or hyperbolic arctangent transformation) for Somers' D:

```
. somersd _t invhr if _st==1, cenind(censind) tdist transf(c)
Somers´ D with variable: _t
Transformation: Harrell´s c
Valid observations: 48
Degrees of freedom: 47
Symmetric 95% CI for Harrell´s c
```

_t	Coef.	Jackknife Std. Err.	t	P>|t|	[95% Conf.	Interval]
invhr	.8106332	.0423076	19.16	0.000	.7255213	.8957451

```
. somersd _t invhr if _st==1, cenind(censind) tdist transf(z)
Somers´ D with variable: _t
Transformation: Fisher´s z
Valid observations: 48
Degrees of freedom: 47
Symmetric 95% CI for transformed Somers´ D
```

_t	Coef.	Jackknife Std. Err.	t	P>|t|	[95% Conf.	Interval]
invhr	.7270649	.1378034	5.28	0.000	.4498402	1.00429

```
Asymmetric 95% CI for untransformed Somers´ D
              Somers_D     Minimum     Maximum
       invhr   .62126643   .42176765   .76338983
```

In both cases, we use the survival-time variable `_t`, the survival sample indicator `_st` (created by `stset`), and the inverse hazard rate `invhr` (created using `predict`) to estimate rank parameters of the inverse hazard ratio with respect to survival time (censored by censorship status). In the case of Harrell's C, the estimated parameter is on a scale from 0 to 1 and is expected to be at least 0.5 for a positive predictor of lifetime, such as an inverse hazard ratio. In the case of Somers' D, the untransformed parameter is on a scale from -1 to 1 and is expected to be at least 0 for a positive predictor of lifetime.

However, we now encounter the third source of confusion mentioned before. If we compare the estimates here to those produced earlier by `estat concordance`, we find that the estimates for Harrell's C and Somers' D are similar but not exactly the same. The estimates are 0.8106 and 0.6213, respectively, when computed by `somersd`, and 0.8086 and 0.6172, respectively, when computed by `estat concordance`. The reason for this discrepancy is that `somersd` and `estat concordance` have different policies for comparing two lifetimes that terminate simultaneously when one lifetime is right-censored and the other is uncensored. The `estat concordance` policy assumes that the owner of the right-censored lifetime survived the owner of the uncensored lifetime, whereas the `somersd` policy assumes that neither of the two owners can be said to have survived the other. In the case of a drug trial, one subject might be known to have

died in a certain month, whereas another might be known to have left the country in the same month and has therefore become lost to follow-up. The `estat concordance` policy assumes that the second subject must have survived the first, which might be probable, given that this second subject seems to have been in a fit state to travel out of the country. The `somersd` policy, more cautiously, allows the possibility that the second subject may have left the country early in the month and died unexpectedly of a venous thromboembolism on the outbound plane, whereas the first subject may have died under observation of the trial organizers later in the same month.

Whatever the merits of the two policies, we might still like to show that `somersd` and `estat concordance` can be made to duplicate one another's estimates. This can easily be done if lifetimes are expressed as whole numbers of time units, as they are in the Stanford drug trial data, where lifetimes are expressed in months. In this case, we can add half a unit to right-censored lifetimes only. As a result, right-censored lifetimes become greater than uncensored lifetimes terminating within the same time unit without affecting any other orderings of lifetimes.

In our example, we do this by generating a new lifetime variable, `studytime2`, that is equal to the modified survival time. We then use `stset` to reset the various survival-time variables and characteristics so that the modified survival time is now used. This step is done after using the `assert` command to check that the old `studytime` variable is indeed integer-valued; see [D] **assert** and [D] **functions**. We then proceed as in the previous example:

```
. use http://www.stata-press.com/data/r11/drugtr, clear
(Patient Survival in Drug Trial)
. assert studytime==int(studytime)
. generate studytime2=studytime+0.5*(died==0)
. stset studytime2, failure(died)
     failure event:  died != 0 & died < .
obs. time interval:  (0, studytime2]
 exit on or before:  failure
--------------------------------------------------------------------------------
       48  total obs.
        0  exclusions
--------------------------------------------------------------------------------
       48  obs. remaining, representing
       31  failures in single record/single failure data
    752.5  total analysis time at risk, at risk from t =         0
                              earliest observed entry t =         0
                                   last observed exit t =      39.5
```

(*Continued on next page*)

```
. stcox drug age
        failure _d:  died
  analysis time _t:  studytime2
Iteration 0:   log likelihood = -99.911448
Iteration 1:   log likelihood = -83.551879
Iteration 2:   log likelihood = -83.324009
Iteration 3:   log likelihood = -83.323546
Refining estimates:
Iteration 0:   log likelihood = -83.323546
Cox regression -- Breslow method for ties
No. of subjects =           48                     Number of obs   =        48
No. of failures =           31
Time at risk    =        752.5
                                                   LR chi2(2)      =     33.18
Log likelihood  =   -83.323546                     Prob > chi2     =    0.0000
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
drug	.1048772	.0477017	-4.96	0.000	.0430057	.2557622
age	1.120325	.0417711	3.05	0.002	1.041375	1.20526

```
. estat concordance
  Harrell´s C concordance statistic
        failure _d:  died
  analysis time _t:  studytime2
  Number of subjects (N)              =       48
  Number of comparison pairs (P)      =      849
  Number of orderings as expected (E) =      679
  Number of tied predictions (T)      =       15
         Harrell´s C = (E + T/2) / P =    .8086
                          Somers´ D =    .6172
. predict hr
(option hr assumed; relative hazard)
. generate invhr=1/hr
. generate censind=1-_d if _st==1
. somersd _t invhr if _st==1, cenind(censind) tdist transf(c)
Somers´ D with variable: _t
Transformation: Harrell´s c
Valid observations: 48
Degrees of freedom: 47
Symmetric 95% CI for Harrell´s c
```

_t	Coef.	Jackknife Std. Err.	t	P>|t|	[95% Conf.	Interval]
invhr	.8085984	.0425074	19.02	0.000	.7230845	.8941122

```
. somersd _t invhr if _st==1, cenind(censind) tdist transf(z)
Somers´ D with variable: _t
Transformation: Fisher´s z
Valid observations: 48
Degrees of freedom: 47
Symmetric 95% CI for transformed Somers´ D
```

_t	Coef.	Jackknife Std. Err.	t	P>|t|	[95% Conf.	Interval]
invhr	.7204641	.1373271	5.25	0.000	.4441976	.9967306

```
Asymmetric 95% CI for untransformed Somers´ D
               Somers_D     Minimum     Maximum
       invhr    .6171967   .41711782   .76021766
```

This time, the model fit produces the same output as before, and the command `estat concordance` produces the same estimates as it did before of 0.8086 and 0.6172 for Harrell's C and Somers' D, respectively. But now the same estimates of 0.8086 and 0.6172 are also produced by `somersd`, at least after rounding to four decimal places.

It should be stressed that Harrell's C and Somers' D, computed as above either by `somersd` or by `estat concordance`, are valid measures of the predictive power of a survival model only if there are no time-dependent covariates or lifetimes with delayed entries. However, if `somersd` (instead of `estat concordance`) is used, then sensible estimates can still be produced with weighted data, so long as those weights are explicitly supplied to `somersd`.

3 Comparing predictive powers with training and test sets

Another caution about the results of the previous section is that the confidence intervals generated by `somersd` should not really be taken seriously. This is because, in general, confidence intervals do not protect the user against the consequences of finding a model in a dataset and then estimating its parameters in the same dataset. In the case of Harrell's C and Somers' D of inverse hazard ratios with respect to lifetime, we would expect this incorrect practice to lead to overly optimistic estimates of predictive power because we are measuring the predictive power of a model that is optimized for the dataset in which the predictive power is measured.

We really should be finding models in a training set of data and testing the models' predictive powers, both absolute and relative to each other, in a test set of data that is independent of the training set. If we have only one set of data, we might divide its primary sampling units (randomly or semirandomly) into two subsets, and use the first subset as the training set and the second subset as the test set. Sections 3.1 and 3.2 below demonstrate this practice by splitting the Stanford drug-trial data into a training set and a test set of similar sizes, using random subsets and semirandom

stratified subsets, respectively. We will use the `somersd` policy, rather than the `estat concordance` policy, regarding tied censored and noncensored lifetimes.

3.1 Completely random training and test sets

We will first demonstrate the relatively simple practice of splitting the sampling units, completely at random, into a training set and a test set. We will fit three models to the training set: model 1, containing the variables `drug` and `age`; model 2, containing `drug` only; and model 3, containing `age` only. Next we will use out-of-sample prediction and `somersd` to estimate the predictive powers of these three models in the test set. We will then use `lincom` to compare their predictive powers, in the manner of section 5.2 of Newson (2006b).

We start by inputting the data and then splitting them, completely at random, into a training set and a test set. We use the `runiform()` function to create a uniformly distributed pseudorandom variable, `sort` to sort the dataset by this variable, and the `mod()` function to allocate alternate observations to the training and test sets (see [D] **sort** and [D] **functions**). We then re-sort the data back to their old order using the generated variable `oldord`.

```
. use http://www.stata-press.com/data/r11/drugtr, clear
(Patient Survival in Drug Trial)
. set seed 987654321
. generate ranord=runiform()
. generate long oldord=_n
. sort ranord, stable
. generate testset=mod(_n,2)
. sort oldord
. tabulate testset, m
```

testset	Freq.	Percent	Cum.
0	24	50.00	50.00
1	24	50.00	100.00
Total	48	100.00	

We see that there are 24 patient lifetimes in the training set (where `testset==0`) and 24 in the test set (where `testset==1`). We then fit the three Cox models to the training set and create the inverse hazard-rate variables `invhr1`, `invhr2`, and `invhr3` for models 1, 2 and 3, respectively:

```
. stcox drug age if testset==0
        failure _d:  died
  analysis time _t:  studytime
Iteration 0:   log likelihood = -36.900079
Iteration 1:   log likelihood = -30.207704
Iteration 2:   log likelihood = -30.075862
Iteration 3:   log likelihood = -30.075741
Refining estimates:
Iteration 0:   log likelihood = -30.075741
Cox regression -- Breslow method for ties
No. of subjects =           24                     Number of obs   =        24
No. of failures =           14
Time at risk    =          370
                                                   LR chi2(2)      =     13.65
Log likelihood  =   -30.075741                     Prob > chi2     =    0.0011
------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        drug |   .1302894    .085747    -3.10   0.002     .0358683     .473269
         age |   1.139011   .0678588     2.18   0.029     1.013482    1.280089
------------------------------------------------------------------------------

. predict hr1
(option hr assumed; relative hazard)
. generate invhr1=1/hr1
. stcox drug if testset==0
        failure _d:  died
  analysis time _t:  studytime
Iteration 0:   log likelihood = -36.900079
Iteration 1:   log likelihood = -32.692209
Iteration 2:   log likelihood = -32.647379
Iteration 3:   log likelihood = -32.647309
Refining estimates:
Iteration 0:   log likelihood = -32.647309
Cox regression -- Breslow method for ties
No. of subjects =           24                     Number of obs   =        24
No. of failures =           14
Time at risk    =          370
                                                   LR chi2(1)      =      8.51
Log likelihood  =   -32.647309                     Prob > chi2     =    0.0035
------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        drug |   .1843768    .112761    -2.76   0.006     .0556069     .611341
------------------------------------------------------------------------------

. predict hr2
(option hr assumed; relative hazard)
. generate invhr2=1/hr2
```

(*Continued on next page*)

```
. stcox age if testset==0
        failure _d:  died
  analysis time _t:  studytime
Iteration 0:   log likelihood = -36.900079
Iteration 1:   log likelihood = -35.587135
Iteration 2:   log likelihood =  -35.58462
Refining estimates:
Iteration 0:   log likelihood =  -35.58462
Cox regression -- Breslow method for ties
No. of subjects =           24                     Number of obs   =        24
No. of failures =           14
Time at risk    =          370
                                                   LR chi2(1)      =      2.63
Log likelihood  =    -35.58462                     Prob > chi2     =    0.1048
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age	1.082178	.0526849	1.62	0.105	.9836912	1.190526

```
. predict hr3
(option hr assumed; relative hazard)
. generate invhr3=1/hr3
```

The variables `invhr1`, `invhr2`, and `invhr3` are defined for all observations, both in the training set and in the test set. We then define the censorship indicator, as before, and estimate the Harrell's C indexes in the test set for all three models fit to the training set:

```
. generate censind=1-_d if _st==1
. somersd _t invhr1 invhr2 invhr3 if _st==1 & testset==1, cenind(censind) tdist
> transf(c)
Somers´ D with variable: _t
Transformation: Harrell´s c
Valid observations: 24
Degrees of freedom: 23
Symmetric 95% CI for Harrell´s c
```

_t	Coef.	Jackknife Std. Err.	t	P>|t|	[95% Conf.	Interval]
invhr1	.8819444	.0490633	17.98	0.000	.7804493	.9834396
invhr2	.7916667	.0330999	23.92	0.000	.7231944	.860139
invhr3	.6365741	.0831046	7.66	0.000	.4646592	.808489

We see that Harrell's C of inverse hazard ratio with respect to lifetime is 0.8819 for model 1 (using both drug treatment and age), 0.7917 for model 2 (using drug treatment only), and 0.6366 for model 3 (using age only). All of these estimates have confidence limits, which are probably less unreliable than the ones we saw in the previous section. However, the sample Harrell's C is likely to have a skewed distribution in the presence of such strong positive associations, for the same reasons as Kendall's τ_a (see Daniels and Kendall [1947]). Differences between Harrell's C indexes are likely to have

a less-skewed sampling distribution and are also what we probably really wanted to know. We estimate these differences with `lincom`, as follows:

```
. lincom invhr1-invhr2
 ( 1)  invhr1 - invhr2 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.0902778	.0350965	2.57	0.017	.0176751	.1628804

```
. lincom invhr1-invhr3
 ( 1)  invhr1 - invhr3 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.2453704	.0736766	3.33	0.003	.0929586	.3977821

```
. lincom invhr2-invhr3
 ( 1)  invhr2 - invhr3 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.1550926	.0823647	1.88	0.072	-.0152917	.3254769

Model 1 seems to have a slightly higher predictive power than model 2 or (especially) model 3, while the difference between model 2 and model 3 is slightly less convincing. We can also do the same comparison using Somers' D rather than Harrell's C, by using the normalizing and variance-stabilizing z transform, recommended by Edwardes (1995) and implemented using the `somersd` option `transf(z)`. In that case, the differences between the predictive powers of the different models will be expressed in z units (not shown).

3.2 Stratified semirandom training and test sets

Completely random training and test sets may have the disadvantage that, by chance, important predictor variables may have different sample distributions in the training and test sets, making both the training set and the test set less representative of the sample as a whole and of the total population from which the training and test sets were sampled. We might feel safer if we chose the training and test sets semirandomly, with the constraint that the two sets have similar distributions of key predictor variables in the various models.

In our case, we might want to ensure that both the training set and the test set contain their "fair share" of drug-treated older subjects, drug-treated younger subjects, placebo-treated older subjects, and placebo-treated younger subjects. To ensure this, we might start by defining sampling strata that are combinations of treatment status and age group, and split each of these strata as evenly as possible between the training set and the test set. Again, this requires the dataset to be sorted, and we will afterward

sort it back to its original order. We sort as follows, using the `xtile` command to define age groups (see [D] **pctile**):

```
. use http://www.stata-press.com/data/r11/drugtr, clear
(Patient Survival in Drug Trial)
. set seed 987654321
. generate ranord=runiform()
. generate long oldord=_n
. xtile agegp=age, nquantiles(2)
. tabulate drug agegp, m
```

Drug type (0=placebo)	2 quantiles of age 1	2	Total
0	11	9	20
1	16	12	28
Total	27	21	48

```
. sort drug agegp ranord, stable
. by drug agegp: generate testset=mod(_n,2)
. sort oldord
. table testset drug agegp, row col scol
```

	2 quantiles of age and Drug type (0=placebo)								
	1			2			Total		
testset	0	1	Total	0	1	Total	0	1	Total
0	5	8	13	4	6	10	9	14	23
1	6	8	14	5	6	11	11	14	25
Total	11	16	27	9	12	21	20	28	48

This time, the training set is slightly smaller than the test set because of odd total numbers of subjects in sampling strata. We then carry out the model fitting in the training set and the calculation of inverse hazard ratios in both sets using the same command sequence as with the completely random training and test sets, producing mostly similar results, which are not shown. Finally, we estimate the Harrell's C indexes in the test set:

```
. generate censind=1-_d if _st==1
. somersd _t invhr1 invhr2 invhr3 if _st==1 & testset==1, cenind(censind) tdist
> transf(c)
Somers´ D with variable: _t
Transformation: Harrell´s c
Valid observations: 25
Degrees of freedom: 24
Symmetric 95% CI for Harrell´s c
```

_t	Coef.	Jackknife Std. Err.	t	P>|t|	[95% Conf.	Interval]
invhr1	.7911392	.0674598	11.73	0.000	.6519091	.9303694
invhr2	.7257384	.049801	14.57	0.000	.6229542	.8285226
invhr3	.5780591	.0972101	5.95	0.000	.3774274	.7786908

The C estimates for the three models are not dissimilar to the previous ones with completely random training and test sets. Their pairwise differences are as follows:

```
. lincom invhr1-invhr2
 ( 1)  invhr1 - invhr2 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.0654008	.0491405	1.33	0.196	-.0360202	.1668219

```
. lincom invhr1-invhr3
 ( 1)  invhr1 - invhr3 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.2130802	.0763467	2.79	0.010	.0555084	.3706519

```
. lincom invhr2-invhr3
 ( 1)  invhr2 - invhr3 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.1476793	.1080388	1.37	0.184	-.0753017	.3706603

Model 1 (with drug treatment and age) still seems to predict better than model 3 (with age alone). This conclusion is similar if we compare the z-transformed Somers' D values, which are not shown.

4 Extensions to non-Cox survival models

Measuring predictive power using Harrell's C and Somers' D is not restricted to Cox models, but can be applied to any model with a positive or negative ordinal predictor. The **streg** command (see [ST] **streg**) fits a wide range of survival models, each of which has a wide choice of predictive output variables, which can be computed using

`predict` (see [ST] **streg postestimation**). These output variables may predict survival times positively or negatively on an ordinal scale and may include median survival times, mean survival times, median log survival times, mean log survival times, hazards, hazard ratios, or linear predictors.

We will briefly demonstrate the principles involved by fitting Gompertz models to the survival dataset that we used in previous sections. The Gompertz model assumes an exponentially increasing (or decreasing) hazard rate, and the linear predictor is the log of the zero-time baseline hazard rate, whereas the rate of increase (or decrease) in hazard rate, after time zero, is a nuisance parameter. Therefore, if the Gompertz model is true, then so is the Cox model. However, the argument of Fisher (1935) presumably implies that if the Gompertz model is true, then we can be no less efficient, asymptotically, by fitting a Gompertz model instead of a Cox model. We will use the predicted median lifetime as the positive predictor, whose predictive power will be assessed using `somersd`.

We start by inputting the cancer trial dataset and defining the stratified, semirandom training and test sets, exactly as we did in section 3.2. We then fit to the training set Gompertz models 1, 2, and 3, containing, respectively, both drug treatment and age, drug treatment only, and age only. After fitting each of the three models, we use `predict` to compute the predicted median survival time for the whole sample, deriving the alternative positive lifetime predictors `medsurv1`, `medsurv2`, and `medsurv3` for models 1, 2, and 3, respectively:

```
. streg drug age if testset==0, distribution(gompertz) nolog
        failure _d:  died
  analysis time _t:  studytime
Gompertz regression -- log relative-hazard form
No. of subjects =           23                     Number of obs   =        23
No. of failures =           15
Time at risk    =          338
                                                   LR chi2(2)      =     20.62
Log likelihood  =   -14.076214                     Prob > chi2     =    0.0000
------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        drug |   .0948331   .0594575    -3.76   0.000     .0277512    .3240694
         age |   1.172588   .0616365     3.03   0.002     1.057798    1.299836
-------------+----------------------------------------------------------------
      /gamma |   .1553139   .0430892     3.60   0.000     .0708605    .2397672
------------------------------------------------------------------------------

. predict medsurv1
(option median time assumed; predicted median time)
```

```
. streg drug if testset==0, distribution(gompertz) nolog
        failure _d:  died
  analysis time _t:  studytime
Gompertz regression -- log relative-hazard form
No. of subjects =           23                     Number of obs   =        23
No. of failures =           15
Time at risk    =          338
                                                   LR chi2(1)      =     11.02
Log likelihood  =   -18.873214                     Prob > chi2     =    0.0009
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
drug	.153411	.0877048	-3.28	0.001	.0500295	.4704213
/gamma	.1063648	.0361612	2.94	0.003	.0354901	.1772394

```
. predict medsurv2
(option median time assumed; predicted median time)
. streg age if testset==0, distribution(gompertz) nolog
        failure _d:  died
  analysis time _t:  studytime
Gompertz regression -- log relative-hazard form
No. of subjects =           23                     Number of obs   =        23
No. of failures =           15
Time at risk    =          338
                                                   LR chi2(1)      =      5.56
Log likelihood  =   -21.606438                     Prob > chi2     =    0.0184
```

_t	Haz. Ratio	Std. Err.	z	P>|z|	[95% Conf.	Interval]
age	1.117255	.0516156	2.40	0.016	1.020536	1.223142
/gamma	.088458	.0341184	2.59	0.010	.0215871	.1553288

```
. predict medsurv3
(option median time assumed; predicted median time)
```

Unsurprisingly, the fitted parameters are not dissimilar to the corresponding parameters for the Cox regression. We then compute the censorship indicator `censind`, and then the Harrell's C indexes, for the test set:

(*Continued on next page*)

```
. generate censind=1-_d if _st==1
. somersd _t medsurv1 medsurv2 medsurv3 if _st==1 & testset==1, cenind(censind)
> tdist transf(c)
Somers´ D with variable: _t
Transformation: Harrell´s c
Valid observations: 25
Degrees of freedom: 24
Symmetric 95% CI for Harrell´s c
```

_t	Coef.	Jackknife Std. Err.	t	P>|t|	[95% Conf.	Interval]
medsurv1	.7911392	.0674598	11.73	0.000	.6519091	.9303694
medsurv2	.7257384	.049801	14.57	0.000	.6229542	.8285226
medsurv3	.5780591	.0972101	5.95	0.000	.3774274	.7786908

We then compare the Harrell's C parameters for the alternative median survival functions, using `lincom`, just as before:

```
. lincom medsurv1-medsurv2
 ( 1)  medsurv1 - medsurv2 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.0654008	.0491405	1.33	0.196	-.0360202	.1668219

```
. lincom medsurv1-medsurv3
 ( 1)  medsurv1 - medsurv3 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.2130802	.0763467	2.79	0.010	.0555084	.3706519

```
. lincom medsurv2-medsurv3
 ( 1)  medsurv2 - medsurv3 = 0
```

_t	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
(1)	.1476793	.1080388	1.37	0.184	-.0753017	.3706603

Unsurprisingly, the conclusions for the Gompertz model are essentially the same as those for the Cox model.

5 Further extensions

The use of Harrell's C and Somers' D in test sets to compare the power of models fit to training sets can be extended further to nonsurvival regression models. In this case, life is even simpler because we do not have to define a censorship indicator such as `censind` for input to `somersd`. The predictive score is still computed using out-of-sample prediction and can be either the fitted regression value or the linear predictor (if one exists in the model).

The methods presented so far have the limitation that the Harrell's C and Somers' D parameters that we calculated estimate only the ordinal predictive power (in the population from which the training and test sets were sampled) of the precise model that we fit to the training set. We might prefer to estimate the mean predictive power that we can expect (in the whole universe of possible training and test sets) using the same set of alternative models. Bootstrap-like methods for doing this, involving repeated splitting of the same sample into training and test sets, are described in Harrell et al. (1982) and Harrell, Lee, and Mark (1996).

Another limitation of the methods presented here, as mentioned at the end of section 2, is that they should not usually be used with models with time-dependent covariates. This is because the predicted variable input to `somersd`, which the alternative predictive scores are competing to predict, is the length of a lifetime rather than an event of survival or nonsurvival through a minimal time interval, such as a day. A predictor variable for such a lifetime must therefore stay constant, at least through that lifetime, which rules out functions of continuously varying time-dependent covariates.

In Stata, survival-time datasets may have multiple observations for each subject with a lifetime, representing multiple sublifetimes. Discretely varying time-dependent covariates, which remain constant through a sublifetime, can also be included in such datasets. `somersd` can therefore be used when these conditions are met: the model is a Cox regression, the time-dependent covariates vary only discretely, the multiple sublifetimes are the times spent by a subject in an age group, and each subject becomes at risk at the start of each age group to which she or he survives. If the subject identifier variable is named `subid`, and the age group for each sublifetime is represented by a discrete variable `agegp`, then the user may use `somersd` with `cluster(subid) funtype(bcluster) wstrata(agegp)` to calculate Somers' D or Harrell's C estimates restricted to comparisons between sublifetimes of different subjects in the same age group. See Newson (2006b) for details of the options for `somersd`, and see [ST] **stset** for details on survival-time datasets.

If the user has access to sufficient data-storage space, then the age groups can be defined finely (as subject-years or even subject-days), and the discretely time dependent covariates might therefore be very nearly continuously time-dependent. Any training sets or test sets in this case should, of course, be sets of subjects rather than sets of lifetimes.

6 Acknowledgments

I would like to thank Samia Mora, MD, of Partners HealthCare, for sending me the query that prompted me to write this article. I also thank the many other Stata users who have also contacted me over the past few years with essentially similar queries on how to use `somersd` to compare the predictive powers of survival models.

7 References

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. 1935. The logic of inductive inference. *Journal of the Royal Statistical Society* 98: 39–82.

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.

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

———. 2006a. Confidence intervals for rank statistics: Percentile slopes, differences, and ratios. *Stata Journal* 6: 497–520.

———. 2006b. Confidence intervals for rank statistics: Somers' *D* and extensions. *Stata Journal* 6: 309–334.

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

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.

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 `parmest` Stata packages.