

# winratiotest: A command for implementing the win ratio and stratified win ratio in Stata

**Short author list:** J. Gregson, J. Ferreira, T. Collier

**Long author list:** John Gregson<sup>1</sup>, João Pedro Ferreira<sup>2,3</sup>, Tim Collier<sup>1</sup>

<sup>1</sup>Department of Medical Statistics, London School of Hygiene and Tropical Medicine, London, United Kingdom

<sup>2</sup> Université de Lorraine, Inserm, Centre d'Investigations Cliniques Plurithématique 1433, and Inserm U1116, CHRU, F-CRIN INI-CRCT (Cardiovascular and Renal Clinical Trialists), Nancy, France.

<sup>3</sup> UnIC@RISE, Department of Surgery and Physiology, Cardiovascular Research and Development Center, University of Porto, Porto, Portugal.

**Corresponding author email:** john.gregson@lshtm.ac.uk

**Short title:** winratiotest: implementing the win ratio in Stata

**Long title:** winratiotest: A command for implementing the win ratio and stratified win ratio in Stata

**Key words:** win ratio, clinical trials

**Abstract.** The win ratio is a statistical method most commonly used for analyzing composite outcomes in clinical trials. Composite outcomes are composed of two or more distinct ‘component’ events (e.g. myocardial infarction, death) and are typically analyzed using time-to-first event methods ignoring the relative importance of the component events. When using the win ratio, component events are instead placed into a hierarchy from most to least important, more important components can then be prioritized over less important outcomes (e.g., death, followed by myocardial infarction). Furthermore, the win ratio enables outcomes of different types (e.g., time-to-event, continuous, binary, ordinal, and repeat events) to be combined. We present `winratiotest`, a program to implement the win ratio approach for hierarchical outcomes in a flexible and user-friendly way.

## 1 Introduction

Randomised clinical trials often compare the time until an event occurs between two treatment groups. Since specific events may be infrequent, two or more distinct types of event are often grouped together to form what is known as a ‘composite outcome’. Such outcomes are typically analyzed using time-to-first event methods. For example, in heart failure trials it is common to use a composite primary outcome with two components: cardiovascular (CV) death and heart failure-related hospitalisation (HFH). The time to the first event is then analyzed using a Cox proportional hazards model. Since one only considers the time until either a HFH or CV death occurs, such an

approach may not align well with patient priorities. Firstly, HFH and CV death are not likely to be of equal importance to patients (most would prefer to have a HFH instead of a fatal event) but both events are weighted equally in the time-to-first analysis outcome i.e., both are simply counted as “an event”. Secondly, if a patient has a HFH and survives they may go on to have further HFHs or a CV death, but these do not contribute to the outcome when analyzed using a time-to-first event approach.

The win ratio addresses this problem by considering outcomes in a hierarchical fashion. The process is best illustrated using a practical example, which follows later. The basic concept is that we consider the component events in turn in order of their importance e.g., CV death first and then HFH (Pocock et al. 2012). The win ratio approach also allows time-to-first event outcomes to be combined with other types of outcomes. For example, in a cardiovascular trial we may wish to consider effects on further outcomes (e.g., quality of life scores, biomarkers, or results from exercise tests) as indicative of treatment benefit. This may be particularly useful in areas where it is not easy to adequately power trials using conventional time-to-event outcomes.

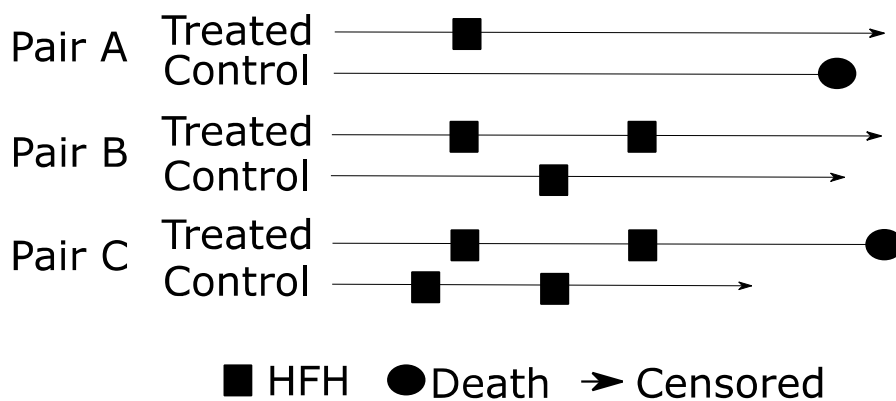
The win ratio has been implemented in several **R** packages (Duarte and Ferreira 2020; Mao and Wang 2020; Luo 2017), but has yet to be formally implemented in Stata. We therefore introduce the `winratiotest` command. The command allows a user to input any number of outcomes and allows for a wide range of outcome types including time-to-failure, time-to-success, repeat events, and continuous outcomes. Binary and ordinal outcomes can also be included by utilizing the similarity between how the win ratio considers continuous, binary, and ordinal outcomes.

## 2 Methods

The win ratio builds upon the Finkelstein-Schoenfeld test (Finkelstein and Schoenfeld 1999), but goes further by providing an easily interpretable estimate of the effect (win ratio) together with a confidence interval. The principle of the approach is as follows. Consider a randomised controlled trial in which patients are randomised to either an active treatment (treated) or a matching placebo (control). First, if the outcome is a composite, each component is ranked from most to least important. Next, all  $N_t$  treated patients are formed into pairs with all  $N_c$  control patients to form a total of  $N_t N_c$  pairs. Within each of these pairs, one evaluates the components in order of priority starting with the most important until one of the pair is determined to have a better outcome than the other. If the patient on the new treatment has the better outcome it is a ‘win’, if the control patient has a better outcome, it is a ‘loss’, or if the patients cannot be separated it is a ‘tie’.

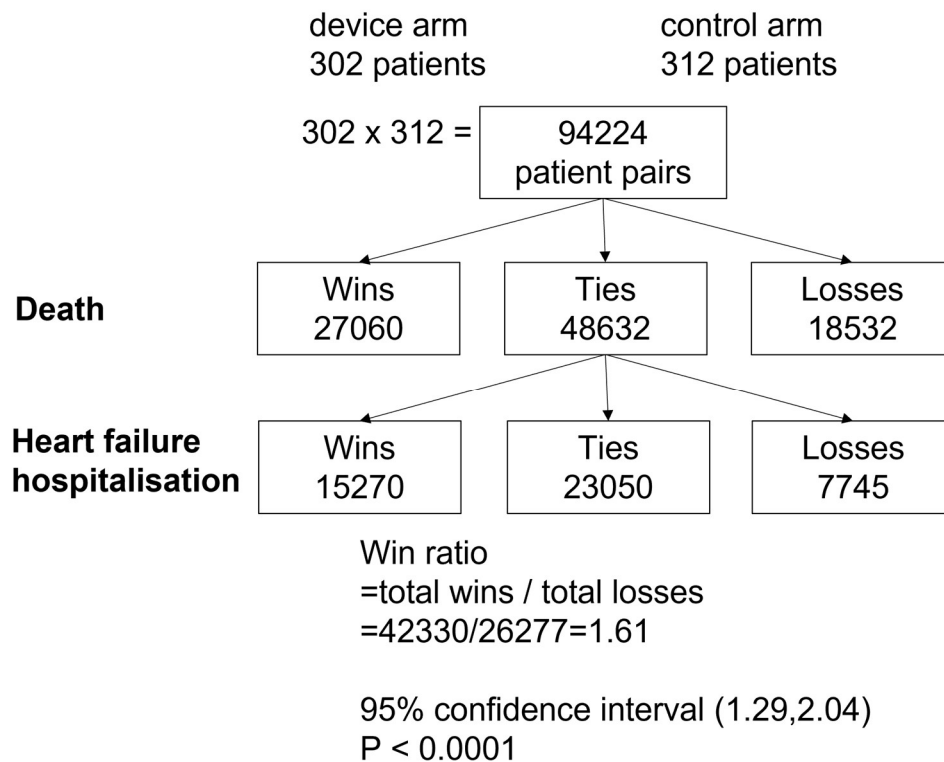
Consider a composite endpoint based on death (the most important outcome) and number of HFH. Such an example is illustrated for 3 pairs of patients in Figure 1. In pair A, the control patient has a

cardiovascular death before the treated patient is censored. The treated patient therefore has a better outcome and this is declared a 'win'. The fact that the treated patient has more HFHs is not considered, because HFH is lower in the hierarchy than death. In pair B, neither patient dies, and so the next level of hierarchy (HFH) is considered. Since the treated patient has more HFHs than the control patient, this is considered a 'loss'. Pair C illustrates the point that for time-to-failure (or time-to-success) outcomes, only outcomes occurring whilst both patients are observed are considered (i.e., before censoring or during 'shared follow-up'). Here the treated patient dies, but we cannot be sure that they died before the control patient who was censored earlier. They are therefore considered a 'tie' on CV death since neither died during their shared follow-up. We next note that both patients had two HFHs during their shared follow-up, and so they are also a tie based on HFH. Pair C is therefore considered a 'tie'.



**Figure 1: Illustrative schematic showing example experience of three pairs of patients in a heart failure trial with two outcomes (death and heart failure hospitalization (HFH)).**

Once all patient pairs have been evaluated the win ratio is calculated as the total number of wins divided by the total number of losses. The win ratio can therefore be readily interpreted as the odds for a randomly chosen pair of patients that the patient in the treated group has a preferable outcome. A corresponding 95% confidence interval and p-value can be calculated, our implementation of which is described in the Appendix. Figure 2 shows a schematic illustrating how the win ratio works considering all patients randomized in the Cardiovascular Outcomes Assessment of the mitralclip Percutaneous Therapy for heart failure patients with functional mitral regurgitation (COAPT) trial using time to death and time to first HFH as the outcome (Stone et al. 2018). Further discussion of the win ratio can be found in Redfors et al. 2020 .



**Figure 2: Schematic showing how the win ratio worked in the Cardiovascular Outcomes Assessment of the mitraclip Percutaneous Therapy for heart failure patients with functional mitral regurgitation trial (COAPT) of the MitraClip device**

One limitation of the unmatched win ratio is that all patient pairs are compared to one another, even though in some trials, there is a large variation in how frail the patients are. In a conventional time-to-event analysis one can adjust for important prognostic covariates in order to overcome this limitation and increase statistical power. However, the win ratio is a non-parametric method that cannot accommodate covariate adjustment; instead, there are two alternative approaches. The first is to use the matched win-ratio, whereby each treated patient is matched to a control patient on the basis of a set of prognostic covariates (Pocock et al. 2012). Whilst this method tends to increase statistical power, it comes with difficulties. The resultant win ratio can be dependent upon which treated patients are matched to which control patients, which is often difficult to pre-specify in an open and transparent way. Also, if the size of the treated and control group is not identical, some patients end up being unmatched and are consequently excluded from the analysis.

For these reasons, we have not implemented the matched win ratio. Another approach is to use the stratified win ratio whereby patients of similar frailty are divided into strata (Bebu and Lachin 2016). This approach has been used in The Transthyretin Amyloidosis Cardiomyopathy Clinical Trial (ATTR-

ACT) in heart failure (Gora et al. 2009). Patients are then only compared within strata, before the results are pooled across strata. Using strata may increase statistical power in some scenarios, but one needs to be careful that strata are sufficiently large to allow reliable estimation of the win ratio within each stratum. We have implemented the stratified win ratio within the `winratiotest` command.

### 3 The winratiotest command

#### 3.1 Syntax

The syntax for `winratiotest` is:

```
winratiotest idvar trtvar [if] , outcomes(list) [options ]
```

where `idvar` is a unique patient/person identifier variable which may be string or numeric, and `trtvar` is a binary numeric variable, taking values 0 and 1 indicating the control and intervention groups respectively.

`outcomes(list)` is a required option where `list` consists of 3 items each of which relates to an outcome in the hierarchy. Each set of 3 items provides information relating to (i) the outcome variable, (ii) the type of outcome and (iii) either the time variable, if the outcome is a time-to-event or a repeated event, or the direction of comparison (< or >), if the outcome is continuous, ordinal or binary. `outcomes` is a repeated option, where the first instance specifies the most important outcome (usually a fatal event) and subsequent instances specify endpoints in decreasing order of importance. See options and examples below for more details.

#### 3.2 Options

`outcomes(list)` is a required option and can be used as a repeated option. The first instance specifies the most important outcome. If multiple outcome types are used then subsequent instances of `outcomes` specify further endpoints in decreasing order of importance. Each `list` consists of 3 items.

The **first item** in each set relates to the *name* of the outcome variable. For binary, ordinal and continuous outcomes this will be the variable name. In the case of a time-to-event outcome this will be a binary (0/1) variable indicating whether or not the event occurred. In the case of a repeat event outcome this will be the *stub* of the names of a set of binary (0/1) variables indicating the repeat events e.g. `hosp` if `hosp1`, `hosp2`, `hosp3` are three binary variables indicating repeat hospitalisations.

The **second item** in each set will indicate the *type* of outcome. This will be `c` for continuous, ordinal or binary outcomes, `tf` for time-to-event failure outcomes, `ts` for time-to-event success outcomes, and `r#` for repeat events (where # is a number representing the maximum number of variables to consider in the analysis of the repeat events).

The **third item** in each set depends on the type of outcome. For time-to-event outcomes this will be the name of the variable containing the follow-up time. For repeat events this will be the stub of the names of a set of variables containing the follow-up times. For continuous, ordinal or binary outcomes this will be `>` or `<` to indicate the direction of the comparison i.e., whether higher or lower values are better. For continuous outcomes a margin can be added. For example, if a 'win' requires that a patient has a score more than 0.1 units higher, then the command syntax would be `<0.1`.

`strata(varname)` allows computation of the stratified win ratio. `varname` must be a numeric categorical variable.

`stweight(method)` allows specification of the weighting method to be used for the stratified win ratio. `method` can be `unweighted` (the default if no weighting option is specified along with `strata`: test statistics and variance estimators are simply summed across strata), `iv` for inverse-variance weights, and `mh` for Mantel-Haenszel weights (each strata is weighted according to the number of patients in the strata).

`pformat(%fmt)` controls the numeric display format for p-values; for example, for a p-value with a leading 0 and 4 decimal places `%05.4f`.

`wrformat(%fmt)` controls the numeric display format for the estimated win ratio and confidence interval.

`saving(filename [,replace])` saves a dataset containing the number of wins, losses, and ties at each level of the hierarchy and for each strata.

### 3.3 Stored results

`winratiotest` stores the following in `r()`:

#### Scalars

`r(p)` the p-value

`r(se_logwr)` standard error of log win ratio. The standard error of the win ratio is calculated using the method described at the end of the Supplement of Pocock 2012

r(wr)                    win ratio  
r(logwr)                the log win ratio

If the strata() option is used then the following scalars are also stored:

r(p#)                    the p-value in the strata #  
r(se#)                    standard error of log win ratio in strata #  
r(wr#)                    the win ratio in strata #  
r(logwr#)                the log win ratio in strata #

## 4 Example

### The data

We illustrate the use of `winratiotest` with a simulated dataset, representative of a heart failure trial with a maximum of one year of follow-up. The dataset contains information on three outcomes: death (a time-to-event variable), heart failure hospitalization (a repeat event outcome with a maximum of three repeat events) and quality of life (for illustrative purposes it is provided in three formats, as each of a continuous, binary or an ordinal variable) measured at the end of the study where higher values are better. A screenshot of the dataset illustrating some key features is shown in **Figure 3**.

The variables contained in the dataset are as follows:

```
. use win_ratio_example.dta , clear
. describe

win_ratio_example.dta
obs:            800
vars:           16

storage  display  value
variable name  type   format   label      variable label
-----
patid       int     %9.0g    Unique patient identifier
trt         byte    %8.0g    Treatment group (0=control, 1=active)
blvar       byte    %9.0g    bllab      Variable containing strata
fudth       float   %9.0f    Follow-up time for death in years (max follow-up is 1-year)
dth         byte    %9.0g    Binary event indicator for death (0=no, 1=yes)
fuhf1       float   %9.0f    Follow-up time for first heart failure hospitalisation
hf1         byte    %9.0g    Binary event indicator for first heart failure
hospitalisation
fuhf2       float   %9.0f    Follow-up time for second heart failure hospitalisation
hf2         byte    %9.0g    Binary event indicator for second heart failure
hospitalisation
fuhf3       float   %9.0f    Follow-up time for third heart failure hospitalisation
hf3         byte    %9.0g    Binary event indicator for third heart failure
hospitalisation
fuhf4       float   %9.0f    Follow-up time for fourth heart failure hospitalisation
hf4         byte    %9.0g    Binary event indicator for fourth heart failure
hospitalisation
qol         float   %9.0g    Quality-of-life measured at 1-year (higher values better)
qol_bin     byte    %9.0g    Binary quality-of-life variable measured at 1-year (0=poor,
1=good)
```

```
qol_cat          byte      %9.0g          Ordinal quality-of-life variable measured at 1-year (1=very
poor, 2=average, 3=v
```

The data structure for repeat events, which can be seen in Figure 3, requires some explanation. In our simulated dataset, a maximum of three HFHs were experienced by any patient in the trial. Four event indicator variables and corresponding follow-up variables are required: three to cover each of the HFH events and an additional column so that the censoring time is known in patients who have three HFHs. The data are in 'wide' format, with each patient having one row with the same number of variables irrespective of the number of events that they have. The event indicators and corresponding follow-up variables must have the same stubs (e.g. `hfh` and `fuhfh`) with suffixes 1 to 4 to indicate the event number, i.e. `hfh1` and `fuhfh1` are the event indicator and follow-up time for the first event.



Treatment code. **winratiotest** always assumes 1=active, 0=comparator

Stratification variable for calculating the stratified win ratio if the syntax `strata(blvar)` is used

For repeat events one specifies the variable stub for the event indicator (here 'hf') and for the time-to-event (here 'fuhf'). The first event and its time are then given in variable hf1, fuhf1. For the second event it is hf2 and fuhf2 etc.

patid	trt	blvar	fudth	dth	fuhf1	hf1	fuhf2	hf2	fuhf3	hf3	fuhf4	hf4	qol	qol_bin	qol_cat
103	0	Level 3	365	0	365	0	365	0	365	0	365	0	9615819	1	3
186	1	Level 3	309	1	309	0	309	0	309	0	309	0	.	.	.
187	0	Level 3	365	0	195	1	365	0	365	0	365	0	3245676	0	1
677	0	Level 2	365	0	37	1	274	1	353	1	365	0	4152547	0	2

Unique patient ID. Must be unique and non-missing

This is a time-to-failure outcome. Also possible to specify time-to-success if the outcome is good (e.g. hospital discharge)

Continuous variable. Note that when **winratiotest** evaluates continuous variables it only considers direction. Here a large value of qol indicates that quality of life is good.

So the direction is specified as '>'.

Binary/ordinal outcomes can also be included in the winratio. Since only the direction is considered one simply specifies them as a continuous ('c') with the appropriate indicator of which direction is favourable, e.g. 'qol\_bin c >'

Figure 3: Screenshot of example data with illustration of some key features

First, consider a patient without any HFH and who was censored alive at 1 year (see patid=103). For this patient `hf1` to `hf4` will all take value 0, with each corresponding follow-up variable (`fuhf1` to `fuhf4`) taking the value 365 (expressed as days). Second, consider a patient who has a single HFH just after six months of follow-up and is then censored alive at one year with no further hospitalizations (see patid=107). For this patient `hf1` will take the value 1 and `fuhf1` takes the value 195. The variables `hf2` to `hf4` will each take the value 0 and `fuhf2` to `fuhf4` will each take the value 365. Third, consider a patient who experienced three HFHs, the variables `hf1` to `hf3` will all take value 1, with each corresponding follow-up variable (`fuhf1` to `fuhf3`) taking the value equal to the three event times (see patid=677). This patient requires a censoring record for the analysis of HFH. Therefore the variable `hf4` takes on the value 0 and `fuhf4` takes the value 365. Fourth, consider a patient who dies during follow-up (see patid=106). The variable `dth` takes the value 1 and the variable `fudth` takes on the number of days from randomization to death, here 309 days. Since the patients experienced no HFHs the variables `hf1`–`hf4` each take the value 0, and the variables `fuhf1`–`fuhf4` each take the value 309. We note that for this patient an evaluation of quality of life at one year would not be possible. Hence the values `qol`, `qol_bin`, `qol_cat` are all set to missing. Patient pairs with missing data are evaluated on the basis of outcomes with complete data only; if there are no outcomes with complete data for both patients then they are considered a tie. Note that missing data is not permitted for time to event or repeat event outcomes and will cause the program to exit execution. This is because patients should instead be censored at their last date of contact.

*Example 1: Win ratio with a time-to-failure and a repeat event outcome*

Suppose we wish to perform an analysis using `winratiotest` where time-to-death is considered the most important outcome, and total number of HFHs is considered as the next most important outcome, with no further outcomes considered. The required syntax is

```
winratiotest patid trt , outcomes(dth tf fudth) outcomes(hf r4 fuhf)
```

Note here that each instance of the `outcomes()` option contains a set of 3 items. The first instance relates to the first outcome in the hierarchy – `dth` is the event indicator, `tf` identifies the outcome as a time to failure outcome and `fudth` is the follow-up time for death. The second instance relates to the second (and final) outcome in the hierarchy – `hf` is the stub of the indicator variables, `r4` identifies the outcome as a repeat event and indicates that variables with suffixes 1-4 are to be used in the analysis (i.e. `hf1`,`hf2`,`hf3`,`hf4`), and `fuhf` is the stub of the follow-up-time variables for each event, again with suffixes 1-4 used in the analysis.

The output using the simulated dataset is

```

. winratiotest patid trt , outcomes(dth tf fudth)      outcomes(hf r4 fuhf)

Total number of patients:      800
Number in control group:      400
Number in active group:      400
Number of comparisons:      160000

                Wins      Losses      Ties
Outcome 1      23182      19870
Outcome 2      29132      17538

Total          52314      37408      70278

Win Ratio: 1.40, 95% CI(1.09, 1.79) P=0.0077

```

The output shows that there are 800 patients in the dataset; 400 in each of the active and control groups. The unmatched win ratio compares all 400 patients in the control group to all 400 patients in the treatment group, yielding a total of  $400 \times 400 = 160,000$  comparisons. The number of wins and losses are displayed at each level in the hierarchy starting with the most important (Outcome 1) and overall, along with the remaining ties. The win ratio, 95% confidence interval and p-value are displayed. The log-win ratio, its standard error and p-value are also available using `return list`. Several methods exist for calculating the standard error of the log-win ratio (Pocock et al. 2012; Dong et al. 2016). For avoidance of doubt, the implementation adopted here is shown in the Appendix.

*Example 2: Win ratio with a time-to-failure, a repeat event, and a continuous outcome*

Suppose we now wish to add the quality-of-life variable (`qol`) as a third level to the hierarchy described in Example 1. To do so we need to add another set of 3 items. The first item is the variable name `qol`. The second item is the variable type, and since `qol` is continuous the second item will be `c`. Finally, since higher values of `qol` represent a better quality of life, i.e. if a treated patient has a higher value of `qol` then it is a 'win', the third item will be the direction `>`. Note that had a lower `qol` represented a better quality of life then the third item would have been `<`.

The required syntax and output is displayed below:

```
. winratiotest patid trt , outcomes(dth tf fudth) outcomes(hf r4 fuhf)
    outcomes(qol c >)
```

```
Total number of patients:      800
Number in control group:       400
Number in active group:       400
Number of comparisons:        160000
```

	Wins	Losses	Ties
Outcome 1	23182	19870	
Outcome 2	29132	17538	
Outcome 3	47132	23146	
Total	99446	60554	0

```
Win Ratio: 1.64, 95% CI(1.39, 1.93) P=0.0000
```

The third outcome of `qol` has now been added to the hierarchy, and the win ratio is now highly statistically significant.

We note here that binary and ordinal outcomes are evaluated in the same way as a continuous outcome, so that `winratiotest` needs to know the name of the variable and which direction represents a preferable outcome. For example, the following syntax could be used to evaluate a hierarchy of death, HFH and a 'good' quality of life assessment (`qol > 0.5` as indicated by `qol_bin`).

```
. winratiotest patid trt , outcomes(dth tf fudth) outcomes(hf r4 fuhf)
    outcomes(qol_bin c >)
```

A further possibility is to introduce a margin into the evaluation of a continuous or ordinal categorical outcome. Suppose we want to have the requirement that the (continuous) quality of life score must be more than 0.1 points higher in order for a patient to be declared a 'winner'. The command syntax would then be:

```
winratiotest patid trt , outcomes(dth tf fudth) outcomes(hf r4 fuhf)
    outcomes(qol c >0.1)
```

Notice that there is no gap between `>` and `0.1`. Also note that if the difference within a pair is exactly `0.1` they will be considered a tie since the inequality is strict.

### *Example 3: Stratified win ratio*

Many trials include patients whose disease at randomization is highly variable in terms of severity. It may sometimes be preferable to only compare patients who are more similar to one another. As previously discussed, one way to do this is by using the stratified win ratio (Dong et al. 2018). Using this approach, patients are first split into strata. In this example `blvar` represents a stratification

variable with 3 levels. In general, we recommend restricting the total number of strata so each stratum contains a reasonable number of patients (i.e. avoid sparse strata). The stratified win ratio then works in two steps. First, within each stratum all control patients are compared to all treated patients and a test statistic and variance are calculated within each stratum. Second, the test statistic and win ratio are combined across strata. The second step can be done in several ways, using either an unweighted approach (test statistics and variance estimators are simply summed across strata: the default), inverse-variance weighting or Mantel-Haenszel type weighting (each stratum weighted according to the number of patients in each strata). Formulae demonstrating technical details of the implementation are given in the Appendix.

In `winratioetest` the stratification variable is declared using the `strata` option, and the weights used to combine estimates across strata are declared using the `stweight` option. For example, to calculate a patient-weighted (Mantel-Haenszel type) stratified win ratio using `blvar` to define the strata the required syntax and output would be:

```
. winratiotest patid trt , outcomes(dth tf fudth) outcomes(hf r4 fuhf)
strata(blvar) stweight(mh)
```

Strata: Level 1

---

Total number of patients:	269		
Number in control group:	139		
Number in active group:	130		
Number of comparisons:	18070		
	Wins	Losses	Ties
Outcome 1	3146	2940	
Outcome 2	3468	1538	
Total	6614	4478	6978
Win Ratio:	1.48, 95% CI(0.99, 2.21) P=0.0579		

---

Strata: Level 2

Total number of patients:	255		
Number in control group:	123		
Number in active group:	132		
Number of comparisons:	16236		
	Wins	Losses	Ties
Outcome 1	1583	2333	
Outcome 2	3080	1973	
Total	4663	4306	7267
Win Ratio:	1.08, 95% CI(0.70, 1.68) P=0.7222		

---

Strata: Level 3

Total number of patients:	276		
Number in control group:	138		
Number in active group:	138		
Number of comparisons:	19044		
	Wins	Losses	Ties
Outcome 1	2963	1349	
Outcome 2	3100	2374	
Total	6063	3723	9258
Win Ratio:	1.63, 95% CI(1.04, 2.55) P=0.0333		
Stratified Win Ratio:	1.38 95% CI (1.08, 1.77) P=0.0114		

---

-----

The expanded output now gives information on the number of wins, losses and ties at each level of the hierarchy for each stratum separately along with the stratum level win ratio, 95% confidence interval and p-value. At the bottom, the overall win ratio, 95% confidence interval and p-value are given.

## 5 Conclusion

We have illustrated how our `winratiotest` command function can be used to analyse a clinical trial using the win ratio.

Our win ratio command is similar in spirit to those available in R (Duarte and Ferreira 2020; Mao and Wang 2020; Luo 2017). It can handle any number of outcome variables, and many different outcome types including time-to-failure, time-to-success, repeated failure, continuous, binary and ordinal outcomes. We are aware of only one R package with similar flexibility in terms of data inputs (Duarte and Ferreira 2020). In addition, our package goes beyond previous software implementations by allowing the user to calculate a stratified estimate of the win ratio, and by allowing the use of a margin for continuous or ordinal variables.

Our implementation of the win ratio has limitations. `winratiotest` has a longer computational time than several R packages (Duarte and Ferreira 2020; Mao and Wang 2020; Luo 2017). But it is fast enough for use except in trials with very large numbers of patients or very complex types of events. We have not provided for calculating the paired win ratio which we considered outside the scope of our program. Our implementation of the win ratio estimates the asymptotic standard error at the null value (i.e. assuming the true win ratio=1). This may not work well in small samples and tends to result in conservative confidence intervals when the true win ratio is far from the null value. An alternative is to use asymptotic estimators at the observed win ratio (Dong et al. 2016; 2018). For patients with missing data in continuous covariates we consider them to be tied on that covariate against all other patients. The consequences of this approach are not currently well understood.

Further work on the `winratiotest` is planned. A major difficulty in designing a clinical trial using the win ratio is the complexity of sample size calculations, which until recently have required simulation studies that are difficult to set up. We are currently developing software to calculate sample size, building on the work of Yu and Ganju (2022) and intend to include this software with future releases of `winratiotest`. We are interested in improving the speed of the `winratiotest` function (perhaps via performing calculations in Mata), since our package is slower than some of those available in R. We are also interested in adding additional flexibility to the types of outcomes included.

## 6 Programs and supplemental material

To install the software files as they exist at the time of the publication of this article, type

```
. ssc install winratiotest
```

## 7 References

- Bebu, Ionut, and John M. Lachin. 2016. "Large Sample Inference for a Win Ratio Analysis of a Composite Outcome Based on Prioritized Components." *Biostatistics* 17 (1): 178–87. <https://doi.org/10.1093/biostatistics/kxv032>.
- Dong, Gaohong, Di Li, Steffen Ballerstedt, and Marc Vandemeulebroecke. 2016. "A Generalized Analytic Solution to the Win Ratio to Analyze a Composite Endpoint Considering the Clinical Importance Order among Components." *Pharmaceutical Statistics* 15 (5): 430–37. <https://doi.org/10.1002/pst.1763>.
- Dong, Gaohong, Junshan Qiu, Duolao Wang, and Marc Vandemeulebroecke. 2018. "The Stratified Win Ratio." *Journal of Biopharmaceutical Statistics* 28 (4): 778–96. <https://doi.org/10.1080/10543406.2017.1397007>.
- Duarte, Kevin, and João Pedro Ferreira. 2020. "Win Ratio for Prioritized Outcomes and 95% Confidence Interval." The Comprehensive R Network.
- Finkelstein, Dianne M., and David A. Schoenfeld. 1999. "Combining Mortality and Longitudinal Measures in Clinical Trials." *Statistics in Medicine*. [https://doi.org/10.1002/\(SICI\)1097-0258\(19990615\)18:11<1341::AID-SIM129>3.0.CO;2-7](https://doi.org/10.1002/(SICI)1097-0258(19990615)18:11<1341::AID-SIM129>3.0.CO;2-7).
- Gora, Sarah, Claire Perret, Ikram Jemel, Viviane Nicaud, Gérard Lambeau, François Cambien, Ewa Ninio, Stefan Blankenberg, Laurence Tiret, and Sonia-Athina Karabina. 2009. "Molecular and Functional Characterization of Polymorphisms in the Secreted Phospholipase A2 Group X Gene: Relevance to Coronary Artery Disease." *Journal of Molecular Medicine (Berlin, Germany)* 87 (7): 723–33. <https://doi.org/10.1007/s00109-009-0483-y>.
- Luo, Xiaodong. 2017. "WWR: Weighted Win Loss Statistics and Their Variances. R Package Version 1.2.2." The Comprehensive R Network. <https://cran.r-project.org/package=WWR>.
- Mao, Lu, and Tuo Wang. 2020. "Win Ratio Analysis." The Comprehensive R Network. <https://cran.r-project.org/package=WR>.
- Pocock, Stuart J., Cono A. Ariti, Timothy J. Collier, and Duolao Wang. 2012. "The Win Ratio: A New Approach to the Analysis of Composite Endpoints in Clinical Trials Based on Clinical Priorities." *European Heart Journal* 33 (2): 176–82. <https://doi.org/10.1093/eurheartj/ehr352>.
- Redfors, Björn, John Gregson, Aaron Crowley, Thomas Mcandrew, Ori Ben-Yehuda, Gregg W. Stone, and Stuart J. Pocock. 2020. "The Win Ratio Approach for Composite Endpoints: Practical Guidance Based on Previous Experience." *European Heart Journal* 41 (46): 4391–99. <https://doi.org/10.1093/eurheartj/ehaa665>.
- Stone, Gregg W, JoAnn Lindenfeld, William T Abraham, Saibal Kar, D Scott Lim, Jacob M Mishell, Brian Whisenant, et al. 2018. "Transcatheter Mitral-Valve Repair in Patients with Heart Failure." *The New England Journal of Medicine* 379 (24): 2307–18. <https://doi.org/10.1056/NEJMoa1806640>.
- Yu, Ron Xiaolong, and Jitendra Ganju. 2022. "Sample Size Formula for a Win Ratio Endpoint." *Statistics in Medicine* 41 (6): 950–63. <https://doi.org/10.1002/sim.9297>.

## About the authors

John Gregson is Assistant Professor in Medical Statistics at the London School of Hygiene and Tropical Medicine. His work on this project was supported by a British Heart Foundation grant.

Joao Pedro Ferreira is a research doctor, Associate Professor and cardiologist at the Regional University Hospital of Nancy. He is also an affiliated lecturer at the University of Glasgow.



Tim Collier is Associate Professor in Medical Statistics at the London School of Hygiene and Tropical Medicine.

## Appendix: Technical details regarding the implementation of the winratio test command

### 1. Calculation of standard errors and 95% confidence intervals

The calculation of p-values and standard errors follows the approach of Pocock (2012).

Suppose there are  $N_t$ ,  $N_c$  patients on treatment and control respectively, so total sample size  $N = N_t + N_c$ .

First we compare all treated patients to all control patients, making a total of  $N_t N_c$  comparisons and calculate the number of wins and losses. The win ratio is then the total number of wins divided by the total number of losses.

Next, **irrespective of treatment group** we compare all possible pairs of patients  $i, j$  to determine whether patient  $i$  was the winner, the loser or they tied, making a total of  $N^2$  comparisons. We assign  $U_{ij} = +1, -1$  or  $0$  according to whether patient  $i$  was the winner, the loser or they tied. Then for patient  $i$  we define  $U_i = \sum_{i \neq j} u_{ij}$ . Note  $U_i$  will be a positive integer if patient  $i$

wins more often than he loses compared to all other patients. Then we calculate  $T = \sum_{i=1}^N U_i D_i$  where  $D_i = 1$  if patient  $i$  is

on the new treatment and  $D_i = 0$  if patient  $i$  is on standard treatment. Under the null hypothesis of no true difference

between new and standard treatment,  $T$  has variance  $V$  where  $V = \frac{N_c N_t}{N(N-1)} \sum_{i=1}^n U_i^2$

Then  $z = \frac{T}{\sqrt{V}}$  is a standardised normal deviate from which P can be readily obtained. Specifically,  $z > 1.96$ ,  $z > 2.38$  and  $z >$

$3.29$  correspond to  $P < .05$ ,  $P < .01$  and  $P < .001$  respectively.

Finally we then calculate the standard error of the log win ratio,  $S$ , as  $S = \log(\text{win ratio})/z$ . 95% confidence intervals reported by **winratio test** are calculated as:

$$\exp \left\{ \log \left( \frac{\text{Wins}}{\text{Losses}} \right) - 1.96S, \log \left( \frac{\text{Wins}}{\text{Losses}} \right) + 1.96S \right\}$$

### 2. Weighting schemes for stratified win ratio within the winratio test command

For the stratified win ratio, irrespective of the weighting scheme the analysis first starts by comparing all patient pairs within each stratum. Within each of  $K$  strata the following statistics are calculated within each stratum separately: 1) Wins and losses within each stratum  $W_k$  and  $L_k$  respectively, and the corresponding win ratio  $WR_k$ ; 2) The test statistic described above,  $T_k$ ; 3) The variance of the within-strata test statistic  $V_k$  4) The standard error of the log win ratio  $SE_k$ .

Each stratum is then assigned a weight,  $weight_k$ . The stratified win ratio is then calculated using

$$SWR = \frac{\sum_{k=1}^K W_k weight_k}{\sum_{k=1}^K L_k weight_k}$$

For the **unweighted stratified** win ratio the calculation of Z-statistics and standard errors proceeds as follows:

$$Z = \frac{\sum_{k=1}^K T_k}{\sqrt{\sum_{k=1}^K V_k}}$$

Standard error of log win ratio:  $SE = \log(SWR)/Z$

For the **inverse-variance** or **Mantel-Haenszel** type weighted win ratio the calculation of Z-statistics and standard errors proceeds as follows. The standard error of the log of the stratified win ratio is calculated as

$$SE = \sqrt{\sum_{k=1}^K SE_k^2 weight_k}$$

For inverse-variance the strata weights are defined as:

$$weight_k = \frac{1}{V_k} / \sum_{k=1}^K \frac{1}{V_k}$$

For Mantel-Haenszel type weighting the strata weights are defined as:

$$weight_k = \frac{1}{N_k} / \sum_{k=1}^K \frac{1}{N_k}$$