The Stata Journal (2018)
18, Number 1, pp. 3–21

Power and sample-size analysis for the Royston–Parmar combined test in clinical trials with a time-to-event outcome

Patrick Royston
MRC Clinical Trials Unit
University College London
London, UK
j.royston@ucl.ac.uk

Abstract. Randomized controlled trials with a time-to-event outcome are usually designed and analyzed assuming proportional hazards (PH) of the treatment effect. The sample-size calculation is based on a log-rank test or the nearly identical Cox test, henceforth called the Cox/log-rank test. Nonproportional hazards (non-PH) has become more common in trials and is recognized as a potential threat to interpreting the trial treatment effect and the power of the log-rank test—hence to the success of the trial. To address the issue, in 2016, Royston and Parmar (*BMC Medical Research Methodology* 16: 16) proposed a "combined test" of the global null hypothesis of identical survival curves in each trial arm. The Cox/log-rank test is combined with a new test derived from the maximal standardized difference in restricted mean survival time (RMST) between the trial arms. The test statistic is based on evaluations of the between-arm difference in RMST over several preselected time points. The combined test involves the minimum p-value across the Cox/log-rank and RMST-based tests, appropriately standardized to have the correct distribution under the global null hypothesis. In this article, I introduce a new command, `power_ct`, that uses simulation to implement power and sample-size calculations for the combined test. `power_ct` supports designs with PH or non-PH of the treatment effect. I provide examples in which the power of the combined test is compared with that of the Cox/log-rank test under PH and non-PH scenarios. I conclude by offering guidance for sample-size calculations in time-to-event trials to allow for possible non-PH.

Keywords: st0510, power_ct, randomized controlled trial, time-to-event outcome, restricted mean survival time, log-rank test, Cox test, combined test, treatment effect, hypothesis testing, flexible parametric model

1 Introduction

In randomized controlled trials with a time-to-event outcome, nonproportional hazards (non-PH) is increasingly recognized as a potentially important issue. In one investigation, statistically significant non-PH was found in about a quarter of cancer trials (Trinquart et al. 2016). Nonstatistically significant non-PH, still of potential practical importance, is likely to occur in a greater proportion of trials, particularly as trial sample sizes get larger.

Concerns about using the hazard ratio (HR) as a summary measure and as the basis of the Cox/log-rank test of the treatment effect in such trials include poor interpretability and possible loss of power. The difference (or ratio) in restricted mean survival time (RMST) between treatment groups is gaining popularity as a summary measure and as the basis of a possible test of a treatment effect (for example, Dehbi, Royston, and Hackshaw [2017]). RMST at some time point ($t^* > 0$) is the integral of the survival function at t^*, that is, the "area under the survival curve" from 0 to t^*. It is interpreted as the mean of the survival-time distribution truncated at t^*. The difference, ΔRMST, defined as RMST in a research arm minus RMST in the control arm, is the integrated difference between the survival functions, equal to the (signed) area between the survival curves up to t^*. Details and an implementation of RMST and ΔRMST in the community-contributed commands `strmst` and `strmst2` may be found in Royston (2015) and Cronin, Tian, and Uno (2016), respectively.

Briefly, in a two-arm trial, consider testing the "global" null hypothesis $H_0 : S_0(t) = S_1(t)$ for any $t > 0$, where $S_j(t)$ is the survival function in the jth group ($j = 0, 1$) and $j = 0$ denotes the control group. Royston and Parmar (2016) proposed a test of H_0 based on the separation of the integrated survival curves. It involves evaluating the maximal chi-squared statistic $C_{\max} = \max\left(Z^2\right)$ over several time points, where $Z = \Delta\text{RMST}/\text{SE}(\Delta\text{RMST})$ is the standardized difference in RMST at a given time point. Arguing pragmatically, Royston and Parmar (2016) determined $C_{\max}$ over 10 equally spaced values of time t^* between the 30th and 100th centiles of the failure times in the dataset. Starting with $C_{\max}$, they developed an approach to testing H_0 that they called the "combined test". The p-value for the combined test is the smaller of the p-value for the Cox/log-rank test and the multiplicity-corrected p-value for $C_{\max}$. In simulation studies, the combined test was shown to be more powerful than the Cox/log-rank test when an "early" effect of treatment was present and only slightly less powerful under PH. The combined test, as implemented in the community-contributed command `stctest`, is described in Royston (2017b).

To my knowledge, power and sample size for the combined test are not computable in closed form. The purpose of this article is to present a command, `power_ct`, for exploring the power and sample size for the combined test. The command `power_ct` uses simulation of possibly censored survival times to estimate power or sample size based on a given trial design. A related helper command, `stcapture` (Royston 2017a), available on Statistical Software Components, outputs survival functions and time-dependent HRs estimated from a dataset in memory. Results from `stcapture` may be fed directly to `power_ct` in the form of stored macros. This facilitates exploration of the operating characteristics of the combined and Cox/log-rank tests under realistic patterns of survival and HR functions.

This article proceeds as follows: In section 2, I describe estimation of RMST and my simulation-based approach to estimate power and sample size for the combined test. In section 3, I introduce the new command `power_ct`, which implements the power and sample-size computations for trial designs with staggered patient entry and defined timelines for patient accrual and follow-up. In section 4, I provide examples under different scenarios of PH and non-PH of the treatment effect. In section 5, I offer broad

suggestions on how to approach sample-size calculations in such trials. In section 6, I conclude with a brief discussion.

2 Methods

2.1 Estimation of RMST

Estimation of RMST for a sample of time-to-event data at a point $t^* > 0$ in analysis time requires determining the area under the survival curve from 0 to t^*. Two methods are available in Stata: Method 1 involves jackknife estimation of pseudovalues (Andersen, Hansen, and Klein 2004), which is equivalent to integrating the Kaplan–Meier curve. Method 2 involves integrating smooth survival curves predicted from flexible parametric survival models (Royston and Parmar 2002; Lambert and Royston 2009; Royston and Lambert 2011), also known as Royston–Parmar models. These methods are described briefly by Royston (2017b) and are implemented in commands `stctest ps` and `stctest rp`, respectively.

2.2 Simulation to estimate power of the combined test

A challenging preliminary task is to devise an approach to simulation that allows a range of survival and time-dependent HR functions to be studied. For convenience and generality, I adopt a somewhat simplified version of the ART system of trial design (Barthel, Royston, and Babiker 2005; Barthel et al. 2006). The ART system computes power or sample size for the Cox/log-rank test by exact calculation not requiring simulation.

Trial calendar time is divided into M contiguous periods of equal length. A period is a convenient duration such as a year, quarter, or month, depending on the context. Staggered entry of patients into the trial is assumed to occur at a steady (uniform) rate within each period, while potentially varying between periods. Typically, in real trials, patient recruitment starts slowly and speeds up over calendar time. The survival distribution in the control arm, $S_0(t)$, is defined by its values at the end of each period of analysis time. The instantaneous event rate (hazard) is assumed constant throughout each period. This defines a piecewise exponential distribution with piecewise constant hazards.

Patient accrual and follow-up are assumed to take place over K_1 and K_2 calendar periods, respectively, with $K_1 + K_2 = M$. For example, for a trial with $M = 10$ periods each of length 1 year and with accrual for $K_1 = 6$ years, follow-up of all patients recruited by staggered entry over 6 years would continue for a further $K_2 = 4$ years, after which one would analyze the trial outcome data.

The survival distribution in the research arm, $S_1(t)$, is specified with HRs for periods $1, 2, \ldots$ of analysis time applied to $S_0(t)$ via the cumulative hazard function. Proportional hazards would require the HRs to be equal across periods.

The power of the combined test is estimated by simulation within the above framework. A suitably large number of datasets with the predefined piecewise exponential distributions in the control and research arms are simulated using the command `power_ct`, which is described in detail in section 3. The number of simulations in which the combined test is significant at a given level α is counted. The power of the test [with a binomial-based confidence interval (CI)] is the corresponding proportion.

The software also presents the power of the Cox/log-rank test as calculated by ART. The simulation-based power of the Cox/log-rank test is provided in addition as a "reality check".

2.3 A note on Monte Carlo error

With any simulation scheme, estimated quantities are not exact but come with a degree of "Monte Carlo error", reflecting uncertainty due to randomness. In `power_ct`, Monte Carlo error attaches to the estimated power or sample size. For power, the binomial method computes the reported CIs. For sample size, I use the delta method to create a normal-based CI, as described in section 2.4.

The `simulate()` and `ciwidth()` options govern the number of simulation replicates used by `power_ct`. If `ciwidth()` is specified, then after some simple algebra, the corresponding required value of `simulate()`, rounded to the nearest integer, is found to be

$$\texttt{simulate()} = \text{round}\left\{\texttt{power()} \times (1 - \texttt{power()}) \times \left(\frac{zz}{\texttt{ciwidth()}}\right)^2\right\}$$

where `power()` is the target power, $zz = -2 * \Phi^{-1}\{(100 - \texttt{level()})/200\}$, where $\Phi^{-1}()$ is the inverse standard normal distribution function, and `level()` is the desired confidence level (by default, 95%). For example, if `power()` = 0.9, `ciwidth()` = 0.02, and `level()` = 95, then $zz = 3.9199$ and `simulate()` = 3457. If `power()` = 0.8 and the other parameters are unchanged, then `simulate()` = 6146.

2.4 Sample-size calculation for the combined test

`power_ct` provides a simulation-based method of estimating the sample size for the combined test to achieve a given power at a given significance level. The user must suggest at least three plausible values for the sample size in option `n()`. Write $\omega \in (0, 1)$ for power and n for the total sample size. Using simulation, the program estimates the power of the combined test at the supplied sample sizes. The several powers and sample sizes are assumed to follow the relation

$$\Phi^{-1}(\omega) = b_0 + b_1\sqrt{n} \tag{1}$$

Functional form (1) is suggested by (1) of Royston et al. (2011). Parameters b_0 and b_1 may be estimated by ordinary least squares. The required sample size, say, n_{est}, for the target power ω_0 is determined by inversion and back-transformation of (1), giving $n_{\text{est}} =$

$\left\{\left[\Phi^{-1}(\omega_0) - b_0\right]/b_1\right\}^2$. A delta-method, normal-based CI for n_{est} may be found by using `nlcom`, for example, `nlcom ((invnormal('omega0') - _b[_cons])/_b[sqrtn])^2`.

Finally, `power_ct` checks the power for n_{est} for agreement with ω_0 by performing an additional round of simulation with sample size n_{est}.

To demonstrate the linearity between $\Phi^{-1}(\omega)$ and $\sqrt{n}$ in an example, I applied `power_ct` to estimate power in sets of 2,000 simulated trials for a design with non-PH across a range of sample sizes between 500 and 1,500. The HRs over the 8 equal time-periods of the design were 0.6, 0.6, 1.0, 1.0, 1.2, 1.5, 1.5, and 1.5. The survival probabilities were 0.90, 0.71, 0.60, 0.52, 0.44, 0.38, 0.33, and 0.28. The results are shown in figure 1.

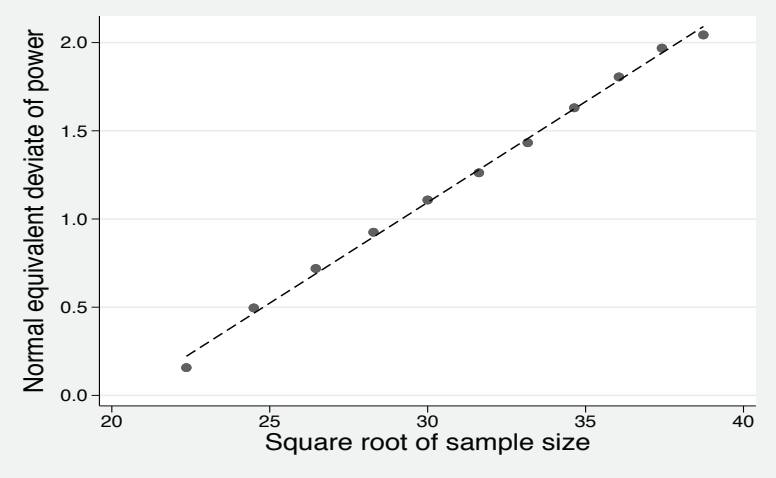

Figure 1. Relationship between normal equivalent deviate of power, $\Phi^{-1}(\omega)$, and square root of sample size in sets of 2,000 simulations of a trial design with non-PH

As can be seen, the relationship between the transformed power and transformed sample size is closely linear across a wide range of powers between about 0.56 and 0.98.

The parameters b_0 and b_1 are estimated as 0.115 and -2.345, respectively. For power $\omega_0 = 0.9$, the required n_0 is 1,000 with 95% CI $[989, 1012]$, rounded to the nearest integer. For comparison, using the same parameters, the Cox/log-rank test requires 4,789 patients to achieve power 0.9.

3 The power_ct command

The syntax of `power_ct` is as follows:

```
power_ct [ , alpha(#) aratio(#) at(numlist) ciwidth(#) graphopts(string)
   hr(numlist|#i) level(#) median(#) n(#) n(numlist) nperiod(#)
   onesided(direction) p0(#) plot[ (fn) ] power(#) recruit(#)
   recwt(numlist) saving(fn2 [ , replace ]) simulate(#)
   survival(numlist|#i) timer tscale(#) ]
```

To enable all the features of `power_ct`, you must install packages containing up-to-date versions of the community-contributed commands `artsurv` (Royston and Barthel 2010), `stpm2` (Andersson and Lambert 2012), `stpmean` (Overgaard, Andersen, and Parner 2015), and `stctest` (Royston 2017b). The do-file `power_ct_install.do` is provided for convenience as part of the installation of `power_ct`. It contains the following commands:

```
. ssc install art, replace
. ssc install stpm2, replace
. quietly net sj 15-3 st0202_1
. net install st0202_1, replace
. quietly net sj 17-2 st0479
. net install st0479, replace
```

The `replace` option ensures that out-of-date installed versions of these programs (if any) are smoothly replaced with the most recent versions.

3.1 Description

`power_ct` has two roles for testing for a generalized treatment effect in a two-arm, parallel group clinical trial with a time-to-event outcome.

- Role 1: To explore the power or sample size for the Cox/log-rank test under proportional hazards (PH) or non-PH. This role uses features of the ART package (Barthel, Royston, and Babiker 2005; Barthel et al. 2006).
- Role 2: To use simulation to evaluate the power or sample size for the combined test of Royston and Parmar (2016) under PH or non-PH.

The combined test combines an unweighted Cox/log-rank test, implemented through `stcox`, with a statistic derived from the maximal squared standardized between-arm difference in time-dependent RMST. See Royston and Parmar (2016) for methodological details and Royston (2017b) for a description of `stctest`, an implementation of the combined test.

Note that `power_ct` is an immediate command. It does not use a dataset in memory or disturb data in memory.

3.2 Options

`alpha(#)` defines the significance level for testing for a difference between treatments. The default is `alpha(0.05)` and tests are two sided. See also `onesided()`.

`aratio(#)` defines the allocation ratio, whereas # equals the number of patients allocated to the research arm for each patient allocated to the control arm. For example, `aratio(0.5)` means one research arm patient for every two control-arm patients. The default is `aratio(1)`, meaning equal allocation.

`at(`*numlist*`)` defines time points such that *numlist* lists the periods corresponding to the values of the control-arm survival function in `survival()`. By default, the periods are assumed to be $1, 2, \ldots$ up to the number of elements in `survival()`.

`ciwidth(#)` defines the desired width of the CI on power when working with simulated data using `simulate()`.

`graphopts(`*string*`)` are options of `graph, twoway` that may be applied to enhance the appearance of the plot produced by the `plot` option.

`hr(`*numlist* `|#`*i*`)` defines HRs. Conventionally, it is assumed that HRs < 1 indicate treatment benefit compared with control, and vice versa for HRs > 1. There are two possible syntaxes:

Syntax 1 is `hr(`*numlist*`)`, where *numlist* defines the HRs to be applied to the control-arm survival function during each period. If only the first k ($k <$ `nperiod()`) HRs are supplied, the remaining HRs up to `nperiod()` are set equal to the last-mentioned value. For example, specifying `hr(0.7 0.8)` and `nperiod(4)` would be equivalent to `hr(0.7 0.8 0.8 0.8)`. If `hr(`*numlist*`)` contained just one number, that is, `hr(#)`, PH with HR equal to # would be assumed. The default is `hr(0.75)`.

Syntax 2 is `hr(#`*i*`)` and uses built-in HR pattern number *i*, where `#` denotes the "hash" character and *i* is a positive integer. The program supplies the HRs in each of 10 periods. When the `plot()` option is used to plot survival curves, an impression is given of the relationship between a non-PH HR pattern and the resulting population survival curves in the control and research arms. The 5 10-period HR patterns currently implemented may be described as follows:

`#1`: early positive effect reversing direction in the long term

`#2`: large late effect

`#3`: large early effect reversing direction, then disappearing

`#4`: fairly small early effect slowly reversing direction

`#5`: early effect with crossing survival curves

The five HR patterns currently implemented are as follows:

`#1`: 0.522 0.642 0.722 0.892 1.193 1.571 1.967 2.288 2.478 2.627

`#2`: 1.0 1.0 0.7 0.5 0.5 0.5 0.5 0.5 0.5 0.5

`#3`: 0.3 0.5 1.0 1.4 1.6 1.7 1.0 1.0 1.0 1.0

`#4`: 0.894 0.701 0.768 0.875 1.013 1.185 1.385 1.594 1.775 1.894

`#5`: 0.5 0.5 0.5 0.7 1.0 1.6 2.0 2.0 2.0 2.0

By default, `hr(#`i`)` (syntax 2) assumes built-in survival-function number i. However, if you provide values in `survival()` or you specify `survival(#`j`)` where $j \neq i$, `survival(#`j`)` values are used instead.

The values of `hr()` and `survival()` that are actually used can be inspected after running `power_ct` by typing `return list` and looking at stored quantities (`r()` macros) `r(hr)`, `r(survival0)`, and `r(survival1)`.

`level(`#`)` sets the confidence level for CIs to #. The default # is `c(level)`, initially 95%. See also `help set level`.

`median(`#`)` specifies the median survival time in the control arm. The survival probability in the control arm at period 1, *s0*, is calculated as $e^{-\ln(2)/\#}$. Options `survival()` and `at()` are posted internally as `survival(`*s0*`)` and `at(1)`, respectively, and may not be used.

`n(`#`)` specifies the sample size at which the power of the Cox/log-rank and combined tests is to be determined. By default, sample size for power given in `power(`#`)` is determined according to the control-arm survival probabilities in `survival()`, HRs in `hr()`, number of periods in `nperiod()`, and accrual time in `recruit()`. See also `hr()` and `n(`*numlist*`)`.

`n(`*numlist*`)` estimates the sample sizes for the combined test to achieve power given by `power()`. `power_ct` first uses simulation (see `simulate()`) to estimate the power of the combined test for each sample size in *numlist*, of which there must be at least three. The values in *numlist* are user-supplied "educated guesses" at the required sample size. `power_ct` then regresses the normal equivalent deviates of the estimated power values on the square roots of the corresponding sample sizes, enabling back-calculation of the sample size for the combined test corresponding to the required power. A normal-based CI for the sample size, computed by the delta method, is presented. Finally, the power with the estimated sample size is determined by a further round of simulation. The CI on the reported power should usually enclose the target power specified in `power()`. More precise estimates of the sample size may be obtained by increasing `simulation()`, reducing `ciwidth()`, or increasing the length of *numlist*.

`nperiod(`#`)` specifies the number of "periods" at the end points of which survival probabilities and HRs apply. Periods are integer numbers of time intervals whose lengths are determined by the reciprocal of # in the `tscale()` option. Note that the number of periods of follow-up time for each patient is calculated as `nperiod()` minus `recruit()`. Hence, `recruit()` cannot exceed `nperiod()`. The default is `nperiod(10)`.

`onesided(`*direction*`)` makes all significance tests one sided. If *direction* = `+`, the direction is toward RMST in the research arm exceeding that in the control arm, and HR < 1. If *direction* = `-`, the direction is toward RMST in the research arm being lower than that in the control arm, and HR > 1. The default is *direction* unspecified, meaning two-sided tests.

`p0(`#`)` defines the fraction of patients recruited at time 0. Such patients are followed up from the start of period 1 and through all subsequent periods. The default is `p0(0)`.

`plot` and `plot(`*fn*`)` plot the estimated population survival functions against analysis time using the time-scale factor `tscale()`. If `plot(`*fn*`)` is specified, the plotted values are stored to a file called *fn*`.dta`; otherwise, no data are stored.

`power(`#`)`, when `n()` is omitted, defines the power at which to determine sample size for the Cox/log-rank test. When `n(`*numlist*`)` is specified, `power_ct` estimates the sample size for the combined test to have power #. Note that `power()` and `n()` cannot be specified together.

`recruit(`#`)` defines the number of periods of calendar time over which patients accrue to the trial. By default, accrual is assumed to occur at a uniform rate; see `recwt()` for how to specify varying recruitment rates. The default is `recruit(5)`.

`recwt(`*numlist*`)` defines accrual weights in each period of patient recruitment. The weights must be a constant multiple of the proportions of patients recruited in each period up to `recruit()`. Values in *numlist* must be positive real numbers. The number of values in *numlist* must equal the number defined by `recruit()`. The default is `recwt(1)`, meaning a constant accrual rate across periods.

`saving(`*fn2* [`, replace`]`)` saves simulation estimates at each replicate to a file named *fn2*`.dta`.

`simulate(`#`)` controls the number of simulations to be performed. The default is `simulate(0)`, meaning no simulations are done.

`survival(`*numlist*`|#`*i*`)` defines survival probabilities in the control arm. Unless `hr(#`*i*`)` is specified (see `hr()`), `survival()` is required. In syntax 1, `survival(`*numlist*`)` defines the survival probability at the end of each period. In syntax 2, `survival(#`*i*`)`, where `#` denotes the "hash" character and *i* is a positive integer, the program assumes `nperiod(10)` and supplies the control-arm survival probability in each of 10 periods according to built-in function *i*. The six built-in survival functions are as follows:

`#1`: 0.765 0.516 0.340 0.221 0.161 0.130 0.112 0.100 0.090 0.082

`#2`: 0.765 0.516 0.340 0.221 0.161 0.130 0.112 0.100 0.090 0.082

`#3`: 0.765 0.516 0.340 0.221 0.161 0.130 0.112 0.100 0.090 0.082

`#4`: 0.500 0.265 0.114 0.065 0.046 0.037 0.032 0.029 0.027 0.025

`#5`: 0.984 0.923 0.773 0.644 0.549 0.471 0.424 0.396 0.377 0.363

`#6`: 0.538 0.333 0.248 0.204 0.178 0.160 0.146 0.136 0.127 0.119

Currently, functions 1, 2, and 3 are identical.

`timer` activates a minute timer for simulation runs. Simulations can become lengthy. Elapsed times from `timer` with a smaller number of simulations in `simulate()` can be scaled up to indicate how long a production run will need. (Technical note: `timer` uses timers 99 and 100.)

`tscale(#)` defines the scale factor between analysis-time units and "periods", where one unit of analysis time equals # periods in length. Note that # may be 1, < 1, or > 1, but it is often 1 or > 1 to "magnify" analysis time and give greater detail of the survival function and HRs. The default is `tscale(1)`.

Example 1: if analysis time is in years and `tscale(2)` is specified, each period is one half a unit of analysis time (that is, six months) in length.

Example 2: if analysis time is in years and `nperiod(12) tscale(4)` is specified, each period is 3 months; survival probabilities are estimated at $1/4, 2/4, \ldots, 12/4 = 3$ years.

4 Examples

4.1 Overview

In the examples given below, I consider computing the sample size for the combined test under three generic scenarios: a) PH, b) early treatment effect, and c) late treatment effect. The last terms are explained in context in the following sections. I estimated the baseline survival function, $S_0(t)$, in the GOG111 trial in advanced ovarian cancer (McGuire et al. 1996). Table 1 shows $S_0(t)$ and $S_1(t)$, the survival function in the research arm, computed with three time-related patterns of HR.

Table 1. Survival functions for three patterns of a time-dependent HR. See also figure 2.

Period (yr)	$S_0(t)$	PH		Non-PH early		Non-PH late	
		HR	$S_1(t)$	HR	$S_1(t)$	HR	$S_1(t)$
1	0.765	0.750	0.818	0.522	0.870	1.000	0.765
2	0.516	0.750	0.609	0.642	0.675	1.000	0.516
3	0.340	0.750	0.445	0.722	0.500	0.700	0.385
4	0.221	0.750	0.322	0.892	0.340	0.500	0.311
5	0.161	0.750	0.254	1.193	0.233	0.500	0.265
6	0.130	0.750	0.217	1.571	0.167	0.500	0.238
7	0.112	0.750	0.194	1.967	0.124	0.500	0.221
8	0.100	0.750	0.178	2.288	0.096	0.500	0.209
9	0.090	0.750	0.164	2.478	0.074	0.500	0.198
10	0.082	0.750	0.153	2.627	0.058	0.500	0.189

The three population survival functions corresponding to the HR functions in table 1 are shown graphically in figure 2.

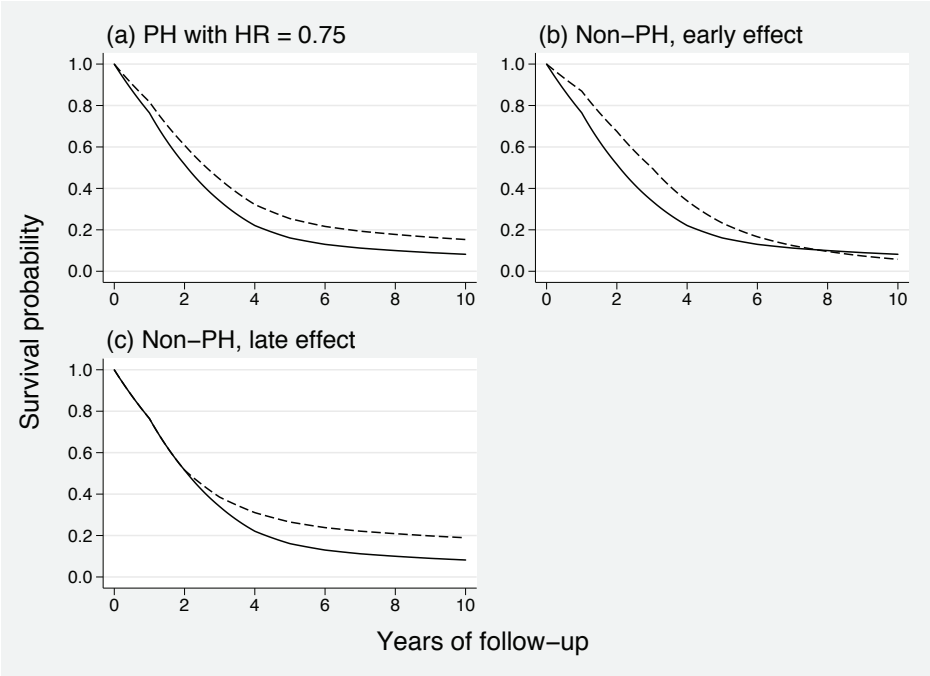

Figure 2. Population survival curves corresponding to PH with HR = 0.75 and non-PH with two different specifications of the time-dependent HR (see table 1). The control-arm survival curve $S_0(t)$ (solid line), also given in table 1, is identical in each panel. The research-arm survival curves are shown by dashed lines.

4.2 Example 1. Sample size under PH

The first port of call for a sample-size calculation with time-to-event data in Stata is usually to assume PH of the treatment effect and work with the Cox/log-rank test. A flexible system to facilitate such exploration is the ART package (Barthel, Royston, and Babiker 2005). The main options of ART (specifically, options of the community-contributed program `artsurv`) are available in simplified form in `power_ct`. If the `simulate()` option of `power_ct` is not specified, `power_ct` calls `artsurv` to calculate the sample size or power according to the Cox/log-rank test but not the combined test.

Previous work suggests that under PH and other things being equal, the combined test requires some 5 to 10% more patients than the Cox/log-rank test to provide a given power (Royston and Parmar 2016). I exemplify the use of `power_ct` in this context.

Suppose the sample size is required to achieve power of 0.9 at a two-sided significance level of 0.05 to detect HR = 0.75 under PH. For illustration, I use the first built-in baseline

survival function in `power_ct` as provided by the option `survival(#1)`. The leftmost four columns of table 1 show the corresponding population survival functions, $S_0(t)$ and $S_1(t)$, at the end of each one-year period up to a maximum of $M = 10$ years.

Suppose that patient accrual takes place at a uniform rate over 5 of the 10 periods. The sample size for the Cox/log-rank test is calculated without requiring simulation:

```
. power_ct, alpha(.05) power(0.9) hr(0.75) survival(#1) nperiod(10) recruit(5)
ART sample size calculation for Cox/logrank test
HR:                PH with HR = .75
Alpha:             .05
Power:             0.9000
Events:            509
Sample size:       599
```

A sample of $n = 599$ patients experiencing 509 events is required for the Cox/log-rank test to have power 0.9. I now check the power of the combined test under the same scenario with 599 patients, using simulation of 5,000 trial datasets. I first set the random-number seed arbitrarily to 115 to ensure reproducibility of the simulation results later if needed:

```
. set seed 115
. power_ct, alpha(.05) n(599) hr(0.75) survival(#1) nperiod(10) recruit(5)
> simulate(5000)
ART power calculation for Cox/logrank test
HR:                PH with HR = .75
Alpha:             .05
Sample size:       599
Events:            509
Power:             0.9002
Estimating power of combined test with sample size 599
(....10%....20%....30%....40%....50%....60%....70%....80%....90%....100%)
```

Simul.	n	Power_CT	[95% Conf. Int.]	Power_Cox	[95% Conf. Int.]	Mean HR
5000	599	0.8776	0.8682, 0.8866	0.8972	0.8884, 0.9055	0.7490

The power of the combined test is estimated to be 0.878 [95% CI 0.868 to 0.887], about 0.022 lower than for the Cox/log-rank test. What sample size is needed for the combined test to achieve power 0.9? To determine this, the `n()` option must specify at least three sample sizes with ranges intended to include the correct value. Because the Cox/log-rank test is optimally powerful under PH, more than 599 patients are needed for the combined test to obtain sufficient power. I take 600, 650, and 700 as a reasonable candidate range. If this set does not cover the required power, I can adjust the sample sizes and repeat the procedure. `power_ct` performs the simulations and interpolates (transformed) sample size and (transformed) power values, as described in section 2.4, to obtain a sample size estimate for power 0.9. It also provides a 95% CI for the estimated power. Finally, `power_ct` again uses simulation to evaluate the power with the final sample size:

```
. set seed 117
. power_ct, alpha(.05) power(.9) n(600 650 700) hr(0.75) survival(#1)
> nperiod(10) recruit(5) simulate(5000)
Estimating power of combined test in 3 sets of 5000 replicates ...
       n     power SE(power)
------------------------------
     600    0.8814    0.0046
     650    0.9024    0.0042
     700    0.9220    0.0038
------------------------------
Sample size for power .9 = 643, 95% CI (640,646)
ART power calculation for Cox/logrank test
HR:                 PH with HR = .75
Alpha:              .05
Sample size:        643
Events:             547
Power:              0.9192
Estimating power of combined test with sample size 643
(....10%....20%....30%....40%....50%....60%....70%....80%....90%....100%)
```

Simul.	n	Power_CT	[95% Conf. Int.]	Power_Cox	[95% Conf. Int.]	Mean HR
5000	643	0.8956	0.8868, 0.9039	0.9154	0.9073, 0.9230	0.7506

The sample size for the combined test is 643, some 7.3% higher than the 599 required by the Cox/log-rank test. Although the power of 0.896 for the combined test does not exactly equal the requested 0.9, the value of 0.9 lies within the 95% CI for 0.896 of $[0.887, 0.904]$.

4.3 Example 2. Sample size under non-PH: Early effect

I now describe an investigation of sample size for the combined test under a specified time-dependent pattern of non-PH. The HR values are given in the fifth column of table 1. The pattern represents an "early" effect of treatment featuring crossing hazard functions, with HR < 1 for the first four periods and HR > 1 subsequently. Royston and Parmar (2016) demonstrated that the Cox/log-rank test may have severely reduced power in this situation.

I specify power 0.9 at significance level 0.05. I cannot rely on the ART calculation to inform the sample size needed for the combined test. Instead, I instruct `power_ct` to cover a wide range of sample sizes. As a "sighting shot", I choose $n = 200$, 500, and 1,000. Initially, to save computer time, I use a relatively small number of simulations (500):

```
. set seed 119
. power_ct, alpha(.05) power(.9) n(200 500 1000) hr(#1) survival(#1)
> nperiod(10) recruit(5) simulate(500)
Estimating power of combined test in 3 sets of 500 replicates ...
       n     power SE(power)
------------------------------
     200    0.6240    0.0217
     500    0.9500    0.0097
    1000    1.0000    0.0000
------------------------------
Sample size for power .9 = 405, 95% CI (405,405)
ART power calculation for Cox/logrank test
HR:                 .522 .642 .722 .892 1.193 1.571 1.967 2.288 2.478 2.627
Alpha:              .05
Sample size:        405
Events:             359
Power:              0.6878
Estimating power of combined test with sample size 405
(....10%....20%....30%....40%....50%....60%....70%....80%....90%....100%)
```

Simul.	n	Power_CT	[95% Conf. Int.]	Power_Cox	[95% Conf. Int.]	Mean HR
500	405	0.8880	0.8570, 0.9143	0.6320	0.5880, 0.6744	0.7803

It appears that $n = 405$ is likely to be close to the right answer, enabling the range to be narrowed. I now choose $n = 350$, 400, and 450 and rerun `power_ct` with a larger number of simulations (5,000):

```
. set seed 121
. power_ct, alpha(.05) power(.9) n(350 400 450) hr(#1) survival(#1)
> nperiod(10) recruit(5) simulate(5000)
Estimating power of combined test in 3 sets of 5000 replicates ...
       n     power SE(power)
------------------------------
     350    0.8706    0.0047
     400    0.9140    0.0040
     450    0.9426    0.0033
------------------------------
Sample size for power .9 = 383, 95% CI (381,384)
ART power calculation for Cox/logrank test
HR:                 .522 .642 .722 .892 1.193 1.571 1.967 2.288 2.478 2.627
Alpha:              .05
Sample size:        383
Events:             339
Power:              0.6636
Estimating power of combined test with sample size 383
(....10%....20%....30%....40%....50%....60%....70%....80%....90%....100%)
```

Simul.	n	Power_CT	[95% Conf. Int.]	Power_Cox	[95% Conf. Int.]	Mean HR
5000	383	0.9022	0.8936, 0.9103	0.6708	0.6576, 0.6838	0.7682

The revised sample size conferring power 0.902 [95% CI 0.894 to 0.910] is 383. Notably, the power of the Cox/log-rank test in this setting is 0.671, which is much smaller than that of the combined test.

As a sensitivity analysis, suppose the same sample size of 383 and the same HRs that define the early effect for the first 4 years are kept, but the subsequent HRs are reduced to 1.0. This is still an early effect lasting four years, but now the treatment effect does not actually "go into reverse" (that is, exhibit crossing hazards) after four years, as it does with `hr(#1)`. What is the power of the Cox/log-rank and combined tests now?

```
. local hr1 0.522 0.642 0.722 0.892 1 1 1 1 1 1
. set seed 123
. power_ct, alpha(.05) n(383) hr(`hr1´) survival(#1) nperiod(10) recruit(5)
> simulate(5000)
ART power calculation for Cox/logrank test
HR:                .522 .642 .722 .892 1 1 1 1 1 1
Alpha:             .05
Sample size:       383
Events:            330
Power:             0.8619
Estimating power of combined test with sample size 383
(....10%....20%....30%....40%....50%....60%....70%....80%....90%....100%)
```

Simul.	n	Power_CT	[95% Conf. Int.]	Power_Cox	[95% Conf. Int.]	Mean HR
5000	383	0.9246	0.9169, 0.9318	0.8600	0.8501, 0.8695	0.7142

The power of the Cox/log-rank test has increased markedly from 0.671 to 0.860. However, it is still somewhat lower than the power of the combined test, which has increased from 0.902 to 0.925.

The example suggests that when an early treatment effect is present, the power of the combined test is superior and quite robust to variations in the HR in the later part of follow-up. The power of the Cox/log-rank test is sensitive to such variations. Typically, relatively few patients who are still event-free contribute data in the late phase.

4.4 Example 3. Sample size under non-PH: Late effect

A second type of non-PH pattern that may be seen, for example, in screening or prevention trials, is the late treatment effect. Here the HR may be close to 1 in the early follow-up phase and decrease later, signifying a late-onset treatment effect. The corresponding survival curves coincide in the early period and separate later. In many trials, most of the events occur in the early follow-up phase. Obtaining sufficient power in such trials can be a challenge.

Earlier simulation work (Royston and Parmar 2016) suggested that with a late effect, the power of the combined test is not far short of that of the Cox/log-rank test. To obtain a sample-size estimate for the combined test, I take an approach similar to the PH case. I obtain a rough indication of the required sample size from the Cox/log-rank

test, then refine it for the combined test using the `n(`*numlist*`)` option of `power_ct`. As an example, I use hypothetical time-dependent HR pattern #2 as provided in `power_ct` through the option `hr(#2)` (see table 1), together with `survival(#1)` as before:

```
. power_ct, alpha(.05) power(.9) hr(#2) survival(#1) nperiod(10) recruit(5)
ART sample size calculation for Cox/logrank test
HR:                 1 1 .7 .5 .5 .5 .5 .5 .5 .5
Alpha:              .05
Power:              0.9000
Events:             811
Sample size:        971
. local n = r(N)
```

I need $n = 971$ patients. Note that `power_ct` stores the required sample size (971) in `r(N)`, which I have stored in local macro `` `n´ `` for further use below. Based on this, I guess a (generous) range of, say, minus 10% to plus 15% of 971, that is, 874 and 1,117, intended to cover power 0.9 for the combined test.

```
. local n1 = round(`n´ - .10*`n´)
. local n2 = round(`n´ + .15*`n´)
. set seed 125
. power_ct, alpha(.05) power(.9) n(`n1´ `n´ `n2´) hr(#2) survival(#1)
> nperiod(10) recruit(5) simulate(5000)
Estimating power of combined test in 3 sets of 5000 replicates ...
       n     power SE(power)
------------------------------
     874    0.8446    0.0051
     971    0.8718    0.0047
    1117    0.9192    0.0039
------------------------------
Sample size for power .9 = 1048, 95% CI (1021,1074)
ART power calculation for Cox/logrank test
HR:                 1 1 .7 .5 .5 .5 .5 .5 .5 .5
Alpha:              .05
Sample size:        1048
Events:             876
Power:              0.9206
Estimating power of combined test with sample size 1048
(....10%....20%....30%....40%....50%....60%....70%....80%....90%....100%)
```

Simul.	n	Power_CT	[95% Conf. Int.]	Power_Cox	[95% Conf. Int.]	Mean HR
5000	1048	0.8990	0.8903, 0.9072	0.9208	0.9130, 0.9281	0.7965

Sample size $n = 1048$ is indicated for the combined test. The corresponding power is 0.899 [95% CI 0.890 to 0.907]. The sample size is 7.9% larger than the 971 needed for the Cox/log-rank test.

5 Sample-size calculation: General recommendations

Based on examples and on previous experience with the combined test (Royston and Parmar 2016), I give tentative recommendations for trial sample-size calculation below. I describe the approach for the most popular power values of 0.9 and 0.8. In principle, any desired power may be targeted.

1. As described in section 4.2, power the trial for the combined test under PH. Power and sample-size assessments under non-PH for various patterns of time-dependent HR may be viewed as sensitivity analyses. They may be informed by subject-matter considerations, for example, hypotheses about the likely modes of action of the treatment regimens under comparison.

2. To achieve power 0.9 for the combined test under PH, initially use the ART methodology implemented in `power_ct` to compute the sample size for the Cox/log-rank test with power 0.92. Simulation is unnecessary. If the target power is 0.8, do the same calculation for Cox/log-rank power 0.83. Call the resulting sample size n_{PH}.

3. To obtain n_{est}, the estimated sample size, run `power_ct` for power 0.9 or 0.8 with three sample sizes covering a sensible range, for example, $n_{\text{PH}}, 0.9n_{\text{PH}}, 1.1n_{\text{PH}}$. The choice of number of simulations may be guided by specifying `ciwidth()` instead of `simulate()`. For example, `ciwidth(0.02)` would give a CI width of about ± 0.01 for the estimated power conditional on n_{est}. The value of n_{est} proposed for the combined test should be close to n_{PH}.

6 Discussion

It is apparent that the power of the Cox/log-rank test is vulnerable to scenarios where a treatment effect in the early follow-up phase disappears or reverses later on. This is the phenomenon of crossing hazard functions. An extreme case of crossing hazards is when the survival curves also cross (see, for example, figure 2A of Mok et al. [2009]). Because of its sensitivity to "gaps" between Kaplan–Meier curves, that is, local features rather than the particular pattern of consistent separation implied by PH, the combined test may be able to detect a statistically and clinically significant early treatment effect when the Cox/log-rank test fails to. Therefore, using the combined test in such cases can enhance power.

Because the Cox/log-rank test is optimally powerful under PH, a requirement for the combined test to provide a given power under PH inevitably causes an increase in sample size. The increase is akin to an "insurance premium" to cope with possible failures of the PH assumption (Royston and Parmar 2016). In practice, the premium is modest, less than a 10% increase in sample size in all the instances I have investigated. The benefit of the combined test is one that is more robust to failure of the PH assumption than the Cox/log-rank test in some situations.

Note that the sample-size calculation for the combined test under PH is sufficiently well defined to be fully specifiable in the trial protocol. The combined test may be

applied to the final trial dataset without the need for any data-driven modifications to the analysis strategy. Such prespecification is a key requirement of good clinical practice when designing and running a trial.

7 Acknowledgment

I thank Dr. Tim Morris for helpful comments on the manuscript.

8 References

Andersen, P. K., M. G. Hansen, and J. P. Klein. 2004. Regression analysis of restricted mean survival time based on pseudo-observations. *Lifetime Data Analysis* 10: 335–350.

Andersson, T. M.-L., and P. C. Lambert. 2012. Fitting and modeling cure in population-based cancer studies within the framework of flexible parametric survival models. *Stata Journal* 12: 623–638.

Barthel, F. M.-S., A. Babiker, P. Royston, and M. K. B. Parmar. 2006. Evaluation of sample size and power for multi-arm survival trials allowing for non-uniform accrual, non-proportional hazards, loss to follow-up and cross-over. *Statistics in Medicine* 25: 2521–2542.

Barthel, F. M.-S., P. Royston, and A. Babiker. 2005. A menu-driven facility for complex sample size calculation in randomized controlled trials with a survival or a binary outcome: Update. *Stata Journal* 5: 123–129.

Cronin, A., L. Tian, and H. Uno. 2016. strmst2 and strmst2pw: New commands to compare survival curves using the restricted mean survival time. *Stata Journal* 16: 702–716.

Dehbi, H.-M., P. Royston, and A. Hackshaw. 2017. Life expectancy difference and life expectancy ratio: Two measures of treatment effects in randomised trials with non-proportional hazards. *British Medical Journal* 357: j2250.

Lambert, P. C., and P. Royston. 2009. Further development of flexible parametric models for survival analysis. *Stata Journal* 9: 265–290.

McGuire, W. P., W. J. Hoskins, M. F. Brady, P. R. Kucera, E. E. Partridge, K. Y. Look, D. L. Clarke-Pearson, and M. Davidson. 1996. Cyclophosphamide and cisplatin compared with paclitaxel and cisplatin in patients with stage III and stage IV ovarian cancer. *New England Journal of Medicine* 334: 1–6.

Mok, T. S., Y.-L. Wu, S. Thongprasert, C.-H. Yang, D.-T. Chu, N. Saijo, P. Sunpaweravong, B. Han, B. Margono, Y. Ichinose, Y. Nishiwaki, Y. Ohe, J.-J. Yang, B. Chewaskulyong, H. Jiang, E. L. Duffield, C. L. Watkins, A. A. Armour, and M. Fukuoka. 2009. Gefitinib or carboplatin–paclitaxel in pulmonary adenocarcinoma. *New England Journal of Medicine* 361: 947–957.

Overgaard, M., P. K. Andersen, and E. T. Parner. 2015. Regression analysis of censored data using pseudo-observations: An update. *Stata Journal* 15: 809–821.

Royston, P. 2015. Estimating the treatment effect in a clinical trial using difference in restricted mean survival time. *Stata Journal* 15: 1098–1117.

———. 2017a. stcapture: Stata module to estimate survival functions and hazard ratios. Statistical Software Components S458312, Department of Economics, Boston College. https://ideas.repec.org/c/boc/bocode/s458312.html.

———. 2017b. A combined test for a generalized treatment effect in clinical trials with a time-to-event outcome. *Stata Journal* 17: 405–421.

Royston, P., and F. M.-S. Barthel. 2010. Projection of power and events in clinical trials with a time-to-event outcome. *Stata Journal* 10: 386–394.

Royston, P., F. M.-S. Barthel, M. K. B. Parmar, B. Choodari-Oskooei, and V. Isham. 2011. Designs for clinical trials with time-to-event outcomes based on stopping guidelines for lack of benefit. *Trials* 12: 81.

Royston, P., and P. C. Lambert. 2011. *Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model*. College Station, TX: Stata Press.

Royston, P., and M. K. B. Parmar. 2002. Flexible parametric proportional-hazards and proportional-odds models for censored survival data, with application to prognostic modelling and estimation of treatment effects. *Statistics in Medicine* 21: 2175–2197.

———. 2016. Augmenting the logrank test in the design of clinical trials in which non-proportional hazards of the treatment effect may be anticipated. *BMC Medical Research Methodology* 16: 16.

Trinquart, L., J. Jacot, S. C. Conner, and R. Porcher. 2016. Comparison of treatment effects measured by the hazard ratio and by the ratio of restricted mean survival times in oncology randomized controlled trials. *Journal of Clinical Oncology* 34: 1813–1819.

About the author

Patrick Royston is a medical statistician with 40 years of experience and a strong interest in biostatistical methods and in statistical computing and algorithms. He works largely in methodological issues in the design and analysis of clinical trials and observational studies. He is currently focusing on alternative outcome measures and tests of treatment effects in trials with a time-to-event outcome, on parametric modeling of survival data, and on novel clinical trial designs.