

## 2. Methods

In this chapter, we first review the current methods for incorporating non-concurrent controls into the analysis of platform trials in Section 2.1. We discuss the differences and similarities, as well as their limitations. These methods include model-based frequentist approaches, the “Time Machine” Bayesian approach and the network meta-analysis approach. In Section 2.2, we extend the frequentist model approaches for treatment-control comparisons to more complex platform trials than those considered in the literature so far. Furthermore, we explore different variants of the models by considering different definitions of the time variable and propose novel methods that permit further flexibility in the adjustment.

### 2.1. Current Methods for Incorporating Non-concurrent Controls

Consider a platform trial design with  $K$  experimental treatment arms and a shared control group, where the experimental arms enter and leave the trial at different time points. The duration of the trial can then be split into periods, defined as time intervals bounded by any treatment arm either entering or leaving the platform. Alternatively, the time can be divided into equidistant discrete units of calendar time (e.g., weeks or months).

We denote the experimental treatment arms by  $k$  ( $k = 1, \dots, K$ , ordered by entry time) and the control group by  $k = 0$ . Furthermore, we denote the periods the trial consists of by  $s$  ( $s = 1, \dots, S$ ) and the calendar time units by  $c$  ( $c = 1, \dots, C$ ). The total number of patients in the trial is given by  $N$ . The response for patient  $j$  ( $j = 1, \dots, N$ ) is indicated by  $y_j$ , the arm they were allocated to by  $k_j$  and the period and calendar time corresponding to their entry time by  $s_j$  and  $c_j$ , respectively. The observed time of recruitment of patient  $j$  is given by  $t_j$ .

The focus of this thesis is on continuous endpoints, hence the frequentist model-based approaches will only be discussed in this context. As the Bayesian Time Machine, however, has only been proposed for binary endpoints, it will be described for this type of endpoints.

#### 2.1.1. Frequentist Model-based Approaches

Naive pooling of concurrent and non-concurrent controls leads to type I error rate inflation and biased treatment effect estimators, if time trends are present in the trial [16, 17]. Therefore, methodology that accounts for possible time trends is needed in order to perform valid statistical analysis that includes NCC data.

## 2. Methods

Focusing on a simple platform trial with  $K = 2$  experimental treatment arms, Lee and Wason [16] investigated linear regression models that allow to include non-concurrent control data to the analysis. In particular, they assumed a trial, which starts with only one treatment arm and the control group. The second treatment arm is added to the trial later on and both treatment arms finish at the same time. The resulting trial consists of 2 periods, as illustrated in Fig. 2.1.

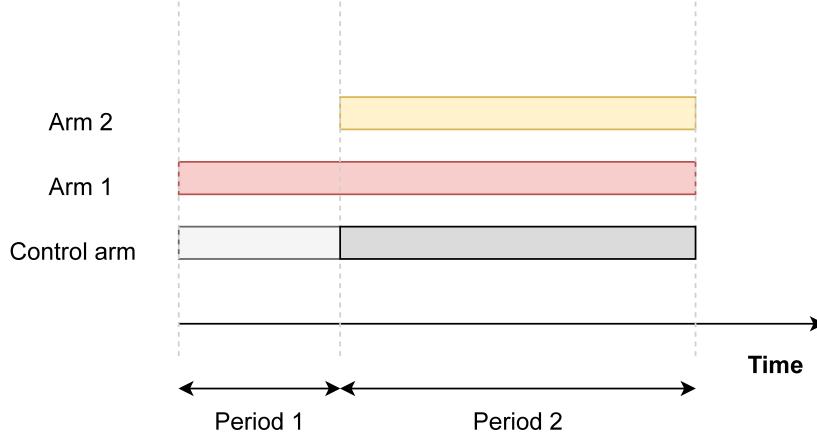


Figure 2.1.: Scheme of the platform trial considered in [16] and [17].

Lee and Wason [16] proposed two linear regression models for evaluating the efficacy of the second treatment arm, which use all data available in the trial (i.e. data from both experimental treatment arms and the control arm). The potential time trends are adjusted for by including the time as a covariate to the model, either in terms of the patient recruitment time (as in (2.1)) or the period indicator (as in (2.2)). The corresponding model equations are as follows:

$$y_j = \eta_0 + \sum_{k=1,2} \theta_k \cdot I(k_j = k) + \gamma \cdot j + \varepsilon_j \quad (2.1)$$

$$y_j = \eta_0 + \sum_{k=1,2} \theta_k \cdot I(k_j = k) + \tau \cdot I(t_j \in s_2) + \varepsilon_j \quad (2.2)$$

where the model intercept  $\eta_0$  is the response in the control group in the first period and  $\theta_k$  is the treatment effect for treatment  $k$ . The time effect is estimated by the parameters  $\gamma$  and  $\tau$ . In (2.1),  $\gamma$  represents the effect of continuous recruitment, while in (2.2)  $\tau$  denotes the time effect between period 1 and 2. The residuals  $\varepsilon_j$  are assumed to be normally distributed with mean 0 and variance  $\sigma^2$  ( $\varepsilon_j \sim \mathcal{N}(0, \sigma^2)$ ). Note that both models implicitly assume that the time trend is equal across all arms. However, the first model (2.1) assumes that the time trend is linear with time, while the second one (2.2) models the time trend as constant in every period.

## 2.1. Current Methods for Incorporating Non-concurrent Controls

Lee and Wason [16] investigated by simulations the operating characteristics of these models when testing the hypothesis  $H_{0,2} : \theta_2 = 0$  against the one-sided alternative  $H_{1,2} : \theta_2 > 0$ , in trials with linear and stepwise time trends that are equal across arms. It was shown that both regression models improve the statistical power as compared to the separate analysis and control the type I error rate, if the functional form of the time trend is correctly specified, the assumption of equal time trends across arms holds and the time effect is additive on the model scale. Moreover, the model with stepwise adjustment (2.2) asymptotically maintains the type I error rate even under misspecification of the functional form of the time trend (e.g., if this is linear instead of stepwise), provided that block randomization is used.

Bofill Roig et al. [17] further investigated these models in trials with time trends that might vary between arms and stated the conditions under which these lead to unbiased treatment effects and type 1 error control. Simulations in [17], showed, also in the context of trials with 2 treatment arms, that the type I error rate control is lost if the time trend in treatment arm 1 differs, and thus when the assumption of equal time trends is violated.

### 2.1.2. Bayesian Time Machine

A Bayesian method that adjusts for potential time trend in the platform trial, the so-called Bayesian Time Machine, has been introduced by Saville et al. [25]. The Time Machine uses the division of the trial into  $C$  calendar time intervals of equal length (“buckets”), which are indexed backwards in time, so that the most recent time interval is denoted by  $c = 1$  and the time interval corresponding to the beginning of the trial by  $c = C$ . The aim is to estimate the treatment effect of the most recently completed treatment arm, while the analysis is performed as soon as the given arm finishes in the trial.

This method has been discussed only in the context of binary endpoints and can be described in terms of a generalized linear model as follows:

$$g(E(y_j)) = \eta_0 + \theta_{k_j} + \alpha_{c_j} \quad (2.3)$$

where  $y_j$  is the binary response for patient  $j$  and  $g(\cdot)$  is the logit link function, which maps the expected value of the patient response to the linear predictors in the model. The model intercept  $\eta_0$  denotes the response of the control group at time of the analysis,  $\theta_{k_j}$  is the effect of the treatment arm  $k$  that patient  $j$  was enrolled in, relative to control. For the parameters  $\eta_0$  and  $\theta_{k_j}$ , normal prior distributions are assumed, with mean 0 and variances  $\sigma_{\eta_0}^2$  and  $\sigma_{\theta}^2$ , respectively:

$$\begin{aligned} \eta_0 &\sim \mathcal{N}(0, \sigma_{\eta_0}^2) \\ \theta_{k_j} &\sim \mathcal{N}(0, \sigma_{\theta}^2) \end{aligned}$$

In the Time Machine, time trend is represented by  $\alpha_{c_j}$ , which is the change in the response in time bucket  $c_j$  (which denotes the time bucket in which patient  $j$  is enrolled)

## 2. Methods

compared to the most recent time bucket  $c = 1$  and is modelled using a Bayesian second-order normal dynamic linear model. This creates a smoothing over the control response, such that closer time buckets are modelled with more similar response rates:

$$\begin{aligned}\alpha_1 &= 0 \\ \alpha_2 &\sim \mathcal{N}(0, 1/\tau) \\ \alpha_c &\sim \mathcal{N}(2\alpha_{c-1} - \alpha_{c-2}, 1/\tau), \quad 3 \leq c \leq C\end{aligned}$$

where  $\tau$  denotes the drift parameter that controls the degree of smoothing over the time buckets and is assumed to have a Gamma hyperprior distribution:

$$\tau \sim \text{Gamma}(a, b)$$

Saville et al. [25] examined the operating characteristics of the Time Machine in a simulation study considering a scenario with  $K = 5$  treatment arms, focusing only on evaluating the efficacy of the last treatment arm ( $k = 5$ ) against the shared control arm (testing  $H_{0,5} : \theta_5 = 1$  vs.  $H_{1,5} : \theta_5 > 1$ ). They show that the Time Machine approximately controls the type I error rate and can lead to superior performance in terms of the statistical power as compared to the frequentist model with categorical adjustment for time in scenarios with linear time trend. However, in situations with sudden changes in the time trend, the frequentist model-based approach is preferable, as the time effect is modelled independently for each time interval in this case.

Similarly to the frequentist model that adjusts for time by including it as a categorical variable, the Bayesian Time Machine relies on the assumption that the time trend affects all arms in the trial equally and is additive on the model scale. Hence there should be no interaction between treatment and time [26]. The Time Machine may also lead to inferior results as compared to the frequentist model in settings with little to no overlap between the treatment arms, as there is not enough data to provide a link between concurrent and non-concurrent controls. Moreover, the performance of the method also depends on the choice of the time buckets, the prior distributions and the values of their corresponding parameters, which need to be chosen individually for a given endpoint, disease and population.

### 2.1.3. Network Meta-analysis

Marschner and Schou [27] proposed to analyse platform trials using meta-analysis techniques, which allows to conduct both, treatment-control as well as treatment-treatment comparisons for non-concurrent arms. They argued that even though the design of a platform trial is usually adapted over time (e.g. by changing randomization allocations, patient recruitment, by adding and dropping treatment arms, etc.), the periods between these adaptations may still be viewed as separate trials with fixed design. In particular, the randomization and allocation ratio is preserved in each period, hence unbiased direct

## 2.1. Current Methods for Incorporating Non-concurrent Controls

comparisons can be made between concurrent arms. Moreover, indirect comparisons between two arms across multiple stages can be performed by combining two direct comparisons to a common reference arm, that is concurrent to both arms of interest at some point (i.e. there is an overlap of the reference arm and the arms being compared). From this perspective, the platform trial can be viewed as a network of direct (concurrent) and indirect (non-concurrent) comparisons.

Let  $\hat{\boldsymbol{\theta}}$  be a vector containing all direct contrast estimates from all periods, i.e., period-wise estimates of treatment-control comparisons, and also of treatment-treatment comparisons of concurrent treatments. The proposed method linearly combines the estimators from  $\hat{\boldsymbol{\theta}}$  to obtain the network estimates  $\hat{\boldsymbol{\theta}}^{network}$  of the treatment-control comparisons. This effect estimator can then be written as:

$$\hat{\boldsymbol{\theta}}^{network} = (\mathbf{X}^T \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^T \mathbf{W} \hat{\boldsymbol{\theta}} \quad (2.4)$$

where  $\mathbf{X}$  is a design matrix specifying the possible treatment contrasts in all periods the trial and  $\mathbf{W}$  is a weight matrix, chosen based on the treatment effect standard errors in each period. This weight matrix also quantifies the contribution of the direct evidence to the overall evidence.

As all previously described approaches for incorporating non-concurrent controls, the validity of the network meta-analysis relies on the assumption that the underlying difference between the compared arms remains the same for all direct and indirect comparisons over time. This assumption can be assessed using formal tests of inconsistency, originally developed in the context of meta-analysis.

### 2.1.4. Open Questions

Novel methods for incorporating NCC data to the analysis of platform trials make use of all data available in the trial to evaluate the efficacy of a particular treatment arm. In other words, data from other experimental treatment arms is borrowed in order to better estimate the effect of time. To guarantee valid statistical inference when including non-concurrent data, the methods discussed in the literature so far rely on the assumption of equal impact of the time trend on all arms in the platform trial on the model scale [26]. Additionally, the Bayesian method only leads to sound results if the values for the parameters of the prior distributions are chosen appropriately. The discussed methods differ in the way the time variable is defined and adjusted for. While the frequentist regression models and the network meta-analysis approach use the concept of periods defined by adaptations in the trial, the Bayesian approach uses time buckets of equal length, similar to actual calendar times.

In the following section, we propose various extensions to the frequentist regression models with regard to the adjustment for time. In particular, we discuss using calendar time adjustment as an alternative to the period adjustment and consider more flexible adjustment techniques, such as random effects and polynomial splines. In Chapter 3, we evaluate the operating characteristics of the newly proposed methods in a simulation study and state under which conditions they lead to valid statistical inference.

## 2. Methods

### 2.2. Extensions to the Frequentist Models for Treatment-Control Comparisons

We extend the frequentist models from [16] and [17] to platform trials with  $K$  experimental treatment arms ( $K > 2$ ), indexed by entry order, and a shared control group (for illustration see Fig. 2.2). Moreover, we propose models that adjust for time trend by using periods, as in [16] and [17], as well as using calendar time intervals as in [25], and explore further approaches to model the time trend.

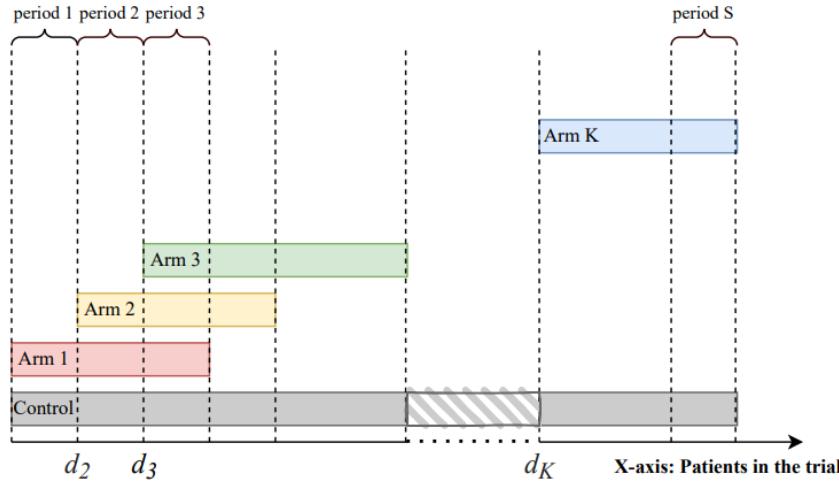


Figure 2.2.: Scheme of the considered general platform trial.

First, we divide the trial into periods, defined as time intervals bounded by adding or dropping experimental treatment arms. For each experimental treatment arm  $k$ ,  $\mathbb{P}(k_j = k)$  is the probability of being allocated to this arm, and is  $> 0$  when the arm is active in the trial (i.e., open for randomization) and 0 before it enters and after it leaves the trial. We denote the entry and exit times of the treatment arm  $k$  by  $t_k^{\text{entry}}$  and  $t_k^{\text{exit}}$ , defined as follows:

$$\forall k \in \{1, \dots, K\}, \forall j \in \{1, \dots, N\} \rightarrow t_k^{\text{entry}} = \min\{t_j \mid \mathbb{P}(k_j = k) > 0\}$$

$$t_k^{\text{exit}} = \max\{t_j \mid \mathbb{P}(k_j = k) > 0\},$$

The vector  $\mathbf{t}^{\text{entry,exit}}$  contains the entry and exit times of all experimental arms.

Denoting by  $\mathbf{t}^{\text{entry,exit}} = \{t_1^{\text{entry}}, t_1^{\text{exit}}, t_2^{\text{entry}}, t_2^{\text{exit}}, \dots, t_K^{\text{entry}}, t_K^{\text{exit}}\}$ , we define the time intervals by considering

$$t_1^{\text{end}} = \min\{t \in \mathbf{t}^{\text{entry,exit}} \mid t > 0\} \text{ for } s = 1$$

$$t_s^{\text{end}} = \min\{t \in \mathbf{t}^{\text{entry,exit}} \mid t > t_{s-1}^{\text{end}}\} \text{ for } s \in \{2, \dots, S\}$$

12 for  $k = 1, \dots, K$ .

④, defined as follows

$$P(k_j = k) = \begin{cases} 0 & \text{if } k \text{ inactive} \\ \frac{1}{\#k_{\text{act}}(t)+1} & \text{if } k \text{ active} \end{cases}$$

where  $k_{\text{act}}(t)$  is the set of active arms at  $t$ .

(#)

(We say that a treatment arm is active in the trial)

$k \in k_{\text{act}}(t)$  if  $t \in [t_k^{\text{entry}}, t_k^{\text{exit}}]$ .

## 2.2. Extensions to the Frequentist Models for Treatment-Control Comparisons

*at before*

where  $S$  is the total number of periods in the trial.

Finally, the time interval for a period  $s$  is given by:

$$T_1^S = [0, t_1^{end}] \text{ for } s = 1$$

$$T_s^S = (t_{s-1}^{end}, t_s^{end}] \text{ for } s \in \{2, \dots, S\}$$

*N*

The trial can also be divided into equidistant units of calendar time. Given the length of the calendar time interval  $c_{length}$ , the trial consists of  $C = \min\{x \in \mathbb{Z} \mid x \geq N/c_{length}\}$  such intervals. The time interval for a calendar time unit  $c$  is then given by:

$$T_c^C = [0, c_{length}]$$

$$T_c^C = ((c-1) \cdot c_{length}, c \cdot c_{length}] \text{ for } c \in \{1, \dots, C-1\}$$

$$T_C^C = ((C-1) \cdot c_{length}, t_K^{exit}] \text{ for } c = C$$

~~We summarize the notation in Table 2.1~~

Notation	Definition
$K$	number of experimental treatment arms
$k = 0, \dots, K$	arm indicator
$N$	total number of patients in the trial
$j = 1, \dots, N$	patient index
$S$	total number of periods
$s = 1, \dots, S$	period indicator
$C$	total number of calendar time units
$c = 1, \dots, C$	calendar time unit indicator
$y_j$	response of patient $j$
$t_j$	patient recruitment time of patient $j$
$t_k^{entry}$	entry time of treatment arm $k$
$t_k^{exit}$	exit time of treatment arm $k$
$t_s^{end}$	end time of period $s$
$T_s^S$	time interval of period $s$
$T_c^C$	time interval of calendar time unit $c$
$\theta_k$	treatment effect of treatment $k$
$\lambda_k$	strength of the time trend in treatment arm $k$
$d$	timing of adding treatment arms
$M$	currently evaluated treatment arm
$S_M$	period, in which arm $M$ left the trial
$C_M$	calendar time unit, in which arm $M$ left the trial
$\mathcal{K}_M$	set of treatment arms active in the trial prior or up to $S_M$ or $C_M$

Table 2.1.: Summary of the used notation.

We focus on evaluating treatment arms that enter when the trial is already ongoing and therefore NCC data is available for these arms. We aim at comparing the efficacy of each

## 2. Methods

treatment against the control as soon as the treatment arm is completed. Consider the one-sided null hypothesis  $H_{0M} : \theta_M \leq 0$  for arm  $M$  under study, where  $\theta_M$  denotes the treatment effect size for treatment  $M$ . To test the null hypothesis  $H_{0M}$ , we propose model-based approaches adjusting for time as fixed or random factor, where time stratified either period or by calendar time. Our proposed models use all data from the trial until treatment  $M$  leaves the platform (i.e., all data in the set  $\mathcal{D}_M = \{(y_j, k_j, t_j), j = 1, \dots, N; t_j \leq t_M^{\text{exit}}\}$ , where  $t_M^{\text{exit}}$  denotes the time in which arm  $M$  finishes). Note that also data from unfinished arms is included to the models. The considered models and corresponding time definitions and adjustments for time trend are summarized in Table 2.2.

Name	Time definition	Time adjustment	Section and reference
<b>Fixed-effect model with period adjustment</b>	period	fixed effect	Section 2.2.1 Equation (2.5)
<b>Fixed-effect model with calendar time adjustment</b>	calendar time	fixed effect	Section 2.2.1 Equation (2.6)
<b>Mixed-effect model with period adjustment and uncorrelated random effects</b>	period	random effect	Section 2.2.2 Equation (2.7)
<b>Mixed-effect model with calendar time adjustment and uncorrelated random effects</b>	calendar time	random effect	Section 2.2.2 Equation (2.8)
<b>Mixed-effect model with period adjustment and autocorrelated random effects</b>	period	random effect	Section 2.2.2 Equation (2.7)
<b>Mixed-effect model with calendar time adjustment and autocorrelated random effects</b>	calendar time	random effect	Section 2.2.2 Equation (2.8)
<b>Spline regression model with knots according to periods</b>	period	polynomial spline	Section 2.2.3 Equation (2.10)
<b>Spline regression model with knots according to calendar times</b>	calendar time	polynomial spline	Section 2.2.3 Equation (2.10)

Table 2.2.: Summary of the proposed models.

### 2.2.1. Fixed-effects Models

Firstly, we consider two linear regression models that estimate the effect of treatments, which were active in the trial prior or up to the time unit  $t_M^{\text{exit}}$  and adjust for potential time trends by including time as a categorical covariate. The two approaches differ in the way time intervals are defined in the model, one using periods and the other the calendar

## 2.2. Extensions to the Frequentist Models for Treatment-Control Comparisons

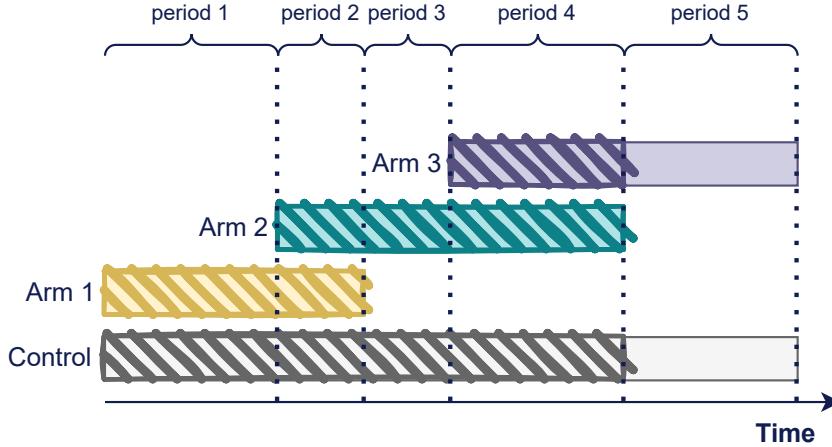


Figure 2.3.: Illustration of the data set  $\mathcal{D}_2$ . Data taken into account for the evaluation of the 2nd treatment arm are highlighted.

time.

### Period Adjustment

In the first model, we adjust for the time effect by including the factor period to the model, i.e., using a step-function:

$$y_j = \eta_0 + \sum_{k \in \mathcal{K}_M} \theta_k \cdot I(k_j = k) + \sum_{s=2}^{S_M} \tau_s \cdot I(t_j \in T_s^S) + \varepsilon_j \quad (2.5)$$

where  $\eta_0$  is the response in the control arm in the first period;  $\theta_k$  represents the effect of treatment  $k$  compared to control for  $k \in \mathcal{K}_M$ , where  $\mathcal{K}_M$  is the set of treatments that were active (i.e., open for randomization) in the trial during periods prior or up to  $t_M^{\text{exit}}$ ;  $\tau_s$  indicates the stepwise period effect between periods 1 and  $s$  ( $s = 2, \dots, S_M$ ), where  $S_M$  denotes the period, in which arm  $M$  left the trial (i.e. the period in which  $t_M^{\text{exit}}$  is included).

This approach is a direct extension of the model (2.2), investigated in [16] and [17], to platform trials with more than two treatment arms and periods. A difference here is that data from treatment arms that entered the trial later than the investigated arm  $M$  and are not finished yet is also used in the analysis. Since we assume that the time effect is equal across arms and model a constant time trend over each period, as in [16] and [17], the model relies on the same assumption, that is, additivity on model scale and equal strength of the time trend across arms.

$$\mathcal{K}_M = \{K_A(t), t \in [0, t_M^{\text{exit}}]\}$$

$$\mathcal{S}_M = \{s : t_M^{\text{exit}} \in T_s\}$$

## 2. Methods

### Calendar Time Adjustment

In the second approach, we consider a regression model adjusting for the time effect by using calendar time intervals, and thus estimating the calendar time effect rather than period effect:

$$y_j = \eta_0 + \sum_{k \in \mathcal{K}_M} \theta_k \cdot I(k_j = k) + \sum_{c=2}^{C_M} \tau_c \cdot I(t_j \in T_c^C) + \varepsilon_j \quad (2.6)$$

Here  $\eta_0$  represents the control response in the first calendar time unit;  $\theta_k$  denotes the treatment effect of treatment  $k$  compared to control for  $k \in \mathcal{K}_M$ , and  $\mathcal{K}_M$  is the set of treatments that were active up until the arm  $M$  left the trial.  $\tau_c$  is the effect between calendar time units 1 and  $c$  ( $c = 2, \dots, C_M$ ), where  $C_M$  indicates the calendar time unit, in which arm  $M$  left the trial.

The division of the trial into calendar times is depicted in Fig. 2.3. The length of these units is given in terms of the number of enrolled patients and can be specified arbitrarily.

**\*\*PENDING: Adapt Fig. 2.3 to also illustrate division into calendar times!\*\***

↘  
 I would only  
use one:  
 - bins  
 - time buckets  
 - calendar time  
 int.

Note that this model, similarly to the Bayesian Time Machine, divides time into bins of equal length. These bins are, unlike the period adjustment considered in [16] and [17], and in the extended model (2.5), independent of alternations to the trial design, i.e. adding and dropping arms is not taken into account.

As the adjustment in this case is done for each calendar time unit, the time effect is modelled as constant in each unit and assumed equal across all arms. Moreover, the length of the time intervals poses an additional design parameter that allows to adjust also for shorter time intervals than are given by the periods.

#### 2.2.2. Mixed Models

*unrelated?*

In models (2.5) and (2.6), time is treated as a fixed factor. Alternatively, patients within different periods or calendar time units could be considered as different clusters, having a period- or calendar time-specific random intercepts. In what follows, we include the time variable to the models as a random factor. Under such models, the potential correlation between the random effects associated with different periods or calendar times can also be taken into account.

#### Mixed Models with Uncorrelated Random Effects

First, we consider simple mixed-effect models, where the effects of the given time intervals (period or calendar time units) are assumed to be uncorrelated with the effects of neighbouring intervals. The mixed-effect model with period adjustment has the following form:

$$y_j = \eta_0 + \sum_{k \in \mathcal{K}_M} \theta_k \cdot I(k_j = k) + \sum_{s=2}^{S_M} u_s \cdot I(t_j \in T_s^S) + \varepsilon_j \quad (2.7)$$

## 2.2. Extensions to the Frequentist Models for Treatment-Control Comparisons

whereas the model adjusting for calendar time units is given by:

$$y_j = \eta_0 + \sum_{k \in \mathcal{K}_M} \theta_k \cdot I(k_j = k) + \sum_{c=2}^{C_M} u_c \cdot I(t_j \in T_c^C) + \varepsilon_j \quad (2.8)$$

where,  $y_j$  and  $\theta_k$  has the same interpretation as in the fixed effect models. The model intercept  $\eta_0$  is in this case given in terms of the control response across the whole trial up until  $S_M$  or  $C_M$ , respectively. Note that this interpretation of the intercept is as in the Time Machine approach.  $u_s$  and  $u_c$  denote the random effect associated with the intercept for period  $s$  or calendar time unit  $c$ .

The period-specific random effects in (2.7) are assumed to be normally distributed with mean 0 and constant variance  $\sigma_{period}^2$ :

$$\mathbf{u} \sim \mathcal{N}(0, \sigma_{period}^2 \cdot I_{S_M \times S_M})$$

The random effects in (2.8) associated with calendar time units are distributed analogously:

$$\mathbf{u} \sim \mathcal{N}(0, \sigma_{calendar}^2 \cdot I_{C_M \times C_M})$$

Note that in this case the correlation between any two period or calendar time effects is 0.

In both models, the distribution of the residuals  $\varepsilon_j$ , associated with the response of individual patient  $j$  is assumed to be the same for all treatments:

$$\varepsilon \sim \mathcal{N}(0, \sigma^2 \cdot I_{N \times N})$$

The random effects,  $u_s$  or  $u_c$ , and the residuals,  $\varepsilon_j$ , are assumed to be independent.

### Mixed Models with Autocorrelated Random Effects

To account for possible correlation of the random effects, we also consider random effects with first-order autoregressive structure (AR(1)), again with period or calendar time adjustments. The model equations are identical to (2.7) and (2.8).

There is a difference, however, with respect to the distribution of the random effects, which are now modelled as autocorrelated. The random effects for individual periods are assumed to be normally distributed with mean 0, constant variance  $\sigma_{period}^2$  and an AR(1) correlation structure:

$$\mathbf{u} \sim \mathcal{N}(0, \sigma_{period}^2 \cdot \Sigma_{S_M \times S_M})$$

## 2. Methods

$$\Sigma_{S_M \times S_M} = \begin{bmatrix} 1 & \phi & \dots & \phi^{S_M-1} \\ \phi & 1 & \dots & \phi^{S_M-2} \\ \vdots & \vdots & \ddots & \vdots \\ \phi^{S_M-1} & \phi^{S_M-2} & \dots & 1 \end{bmatrix}$$

An analogous distribution is assumed for the random effects associated with different calendar times:

$$\mathbf{u} \sim \mathcal{N}(0, \sigma_{calendar}^2 \cdot \Sigma_{C_M \times C_M})$$

$$\Sigma_{C_M \times C_M} = \begin{bmatrix} 1 & \phi & \dots & \phi^{C_M-1} \\ \phi & 1 & \dots & \phi^{C_M-2} \\ \vdots & \vdots & \ddots & \vdots \\ \phi^{C_M-1} & \phi^{C_M-2} & \dots & 1 \end{bmatrix}$$

The parameter  $\phi$  denotes the correlation between two adjacent periods or calendar time units. Note that  $\phi$  can range from -1 to 1 and the correlation of periods that are  $w$  units apart is equal to  $\phi^w$ , such that the correlation is weaker for periods that are further apart.

The residuals are again assumed to be normally distributed (i.i.d.) with mean 0 and constant variance:

$$\varepsilon \sim \mathcal{N}(0, \sigma^2 \cdot I_{N \times N}) \quad (2.9)$$

### 2.2.3. Spline Regression

The models in Sections 2.2.1 and 2.2.2 assumed an underlying linear relationship between the patient response and the time. In order to also capture potential non-linearity of the time trend, we consider estimating the patient response using spline regression. The model is given by:

$$y_j = \eta_0 + \sum_{k \in \mathcal{K}_M} \theta_k \cdot I(k_j = k) + f(t_j) + \varepsilon_j \quad (2.10)$$

where  $y_j$ ,  $\eta_0$  and  $\theta_k$  are defined as in (2.5) and the residuals are  $\varepsilon_j \sim \mathcal{N}(0, \sigma^2)$ . Note that the treatment effect enters the model as a linear predictor. The time trend is modeled via a continuous function  $f(t_j)$  of the patient entry time  $t_j$ , which in our case also indicates their index ( $t_j = j$ ).

In particular, we consider the B-spline function to model the time trend. This function is composed of multiple polynomial functions of a given degree  $q$ , which are joined together at points called *knots*, such that the entire spline is continuously differentiable up to the

## 2.2. Extensions to the Frequentist Models for Treatment-Control Comparisons

$(q - 1)$ th derivative. In our case, the knots are placed within the range of the patient entry times  $t_j$ . To define a B-spline function, we first define the knot sequence

$$\zeta_1 = \dots = \zeta_q = \zeta_{q+1} < \zeta_{q+2} < \dots < \zeta_{q+Z+1} < \zeta_{q+Z+2} = \zeta_{q+Z+3} = \dots = \zeta_{2q+Z+2}$$

where the  $Z$  knots in the set  $\{\zeta_{q+2}, \dots, \zeta_{q+Z+1}\}$  are called *inner knots*, while  $\zeta_{q+1}$  and  $\zeta_{q+Z+2}$  are referred to as *boundary knots*. The additional knots  $\{\zeta_1, \dots, \zeta_q\}$ , as well as  $\{\zeta_{q+Z+3}, \dots, \zeta_{2q+Z+2}\}$  are set equal to the boundary knots, even though their choice is essentially arbitrary and only needed because of the later recursive definition of the B-spline.

The function  $f(t_j)$  can then be represented by a set of basis functions  $B_i^q(t_j)$  as follows:

$$f(t_j) = \sum_{i=1}^{q+Z+1} B_i^q(t_j) \beta_i$$

where  $\beta_i$  are the associated regression coefficients and the functions  $B_i^q(t_j)$  are defined using the Cox-de Boor formula, recursive in  $q$ , as follows:

$$B_i^q(t_j) = \frac{t_j - \zeta_i}{\zeta_{i+q} - \zeta_i} B_i^{q-1}(t_j) + \frac{\zeta_{i+q+1} - t_j}{\zeta_{i+q+1} - \zeta_{i+1}} B_{i+1}^{q-1}(t_j), \quad i = 1, \dots, q + Z + 1$$

with

$$B_i^0 = \begin{cases} 1 & \zeta_i \leq t_j < \zeta_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

and

$$B_i^0 \equiv 0 \text{ if } \zeta_i = \zeta_{i+1}$$

For the positions of the inner knots  $\{\zeta_{q+2}, \dots, \zeta_{q+Z+1}\}$ , we evaluate two strategies, somewhat analogous to the period and calendar time adjustments in Sections 2.2.1 and 2.2.2. Firstly, we place the inner knots to the beginning of each period  $s = 2, \dots, S$ , such that one polynomial of degree  $q$  is always fitted to each period. In this case, the number of inner knots  $Z = S - 1$ .

Moreover, we consider placing the inner knots equidistantly, according to the length of the calendar time unit, and thus fitting a polynomial of degree  $q$  to each calendar time interval, which leads to  $Z = C - 1$ .

The boundary knots  $\zeta_{q+1}$  and  $\zeta_{q+K+2}$  are always set to 1 and  $N$ , respectively, hence to the beginning and end of the trial.

Regarding the degree of the fitted polynomial, we explore linear, quadratic and cubic splines, i.e.  $q \in (1, 2, 3)$ .

Modeling the time using spline functions gives the model additional flexibility as compared to the previously considered approaches, as now also more complex time trend patterns can be modelled more accurately.



### 3. Simulations and Results

The present chapter focuses on assessing the performance of the proposed methods in a simulation study. For each model, we evaluate the type I error rate and statistical power under a wide range of scenarios. In Section 3.1, we describe the general design of the platform trials considered in the study and the chosen design parameters. In Section 3.2, we present the results from the study, discuss the properties of the examined methods and the influence of certain design parameters on the operating characteristics.



#### 3.1. Considered Designs

We consider a platform trial with  $K$  experimental treatment arms that enter the trial sequentially and a control group that is common to all treatment arms. The timing of adding of the treatment arms is given by  $\mathbf{d} = (d_1, \dots, d_K)$ , where  $d_i$  indicates how many patients had already been enrolled to the trial by the time treatment  $i$  entered the platform.  $d_1$  is always set to 0 to ensure that the platform trial starts with at least one experimental treatment (for illustration see Fig. 2.2).

We distinguish four settings in which we vary the design according to the objectives of the simulation study:

*varying*

- **Setting 1:** Trial with  $K = 10$  experimental arms and linear time trend pattern, where we vary the strength of the time trend, as well as the timings of adding new treatment arms. We focus on comparing the efficacy of arm  $M = 5$  to the shared control group, but also discuss how the power of individual treatment-control comparisons depends on the entry order of the treatment arms. In this setting, we aim at evaluating the generalization of the model-based approach with period adjustment to trials with more than two experimental arms.
- **Setting 2:** Trial with  $K = 4$  experimental arms, where we evaluate arm  $M = 3$ , while varying the pattern and strength of the time trend, as well as the size of the calendar time unit. Here, we aim at comparing the definition of time as calendar time intervals to the period definition in fixed effect models.
- **Setting 3:** Trial with  $K = 4$  experimental arms, where we evaluate arm  $M = 3$ , while varying the pattern and strength of the time trend. In this setting, we evaluate the performance of the mixed models.
- **Setting 4:** Trial with  $K = 7$  experimental arms, where we evaluate arm  $M = 3$ , while varying the pattern and strength of the time trend. In this setting, we assess the properties of the spline regression.

### 3. Simulations and Results

To generate the trial data, we assume equal sample sizes in all treatment arms ( $n_k = n = 250, \forall k = 1, \dots, K$ ) and an allocation ratio of  $1 : 1 : \dots : 1$  in each period. Patients are assigned to arms following block randomization with block sizes of  $2 \cdot (\#\text{active arms} + 1)$  in every period. Patients are indexed by entry order, assuming that at each time unit exactly one patient is recruited and the time of recruitment and observation of the response are equal. The continuous outcome  $y_j$  for patient  $j$  is then drawn from a normal distribution according to:

$$y_j \sim \mathcal{N}(\mu, \sigma^2)$$

with

$$\mu = \eta_0 + \sum_{k=1}^K \theta_k \cdot I(k_j = k) + f(j)$$

and

$$\sigma^2 = 1$$

where  $\eta_0$  and  $\theta_k$  are response in the control arm and the effect of treatment  $k$ . Moreover, time trends of various strengths and shapes may be present in the trial. The time trends are denoted by the function  $f(j)$  and their magnitude is given by  $\lambda$ . The following time trend patterns are considered:

*capital letters*

- **linear time trend:**  $f(j) = \lambda \cdot \frac{j-1}{N-1}$ , where  $N$  indicates the total sample size in the trial.
- **stepwise time trend:**  $f(j) = \lambda \cdot (w_j - 1)$ , where  $w_j$  indicates how many treatment arms have already entered the trial at the time patient  $j$  was enrolled.
- **inverted-U trend:**  $f(j) = \lambda \cdot \frac{j-1}{N-1} (I(j \leq N_p) - I(j > N_p))$ , where  $N$  indicates the total sample size in the trial and  $N_p$  is the point at which the trend turns from positive to negative in terms of the sample size.
- **seasonal trend:**  $f(j) = \lambda \cdot \sin(\psi \cdot 2\pi \cdot \frac{j-1}{N-1})$ , where  $N$  indicates the total sample size in the trial and  $\psi$  determines the number of cycles over the whole platform trial.

Under the linear time trend, the mean response linearly increases with the slope  $\lambda$  over time, while under the stepwise time trend, there is a jump in the mean response of size  $\lambda$  every time a new arm is added to the trial. In the case of the inverted-U time trend, the mean response linearly increases (with slope  $\lambda$ ) until the sample size has reached  $N_p$ , and linearly decreases afterwards. The seasonal trend may consist of multiple cycles, where the response increases at first and decreases afterwards, while the respective peaks of the cycles correspond to  $\lambda$  or  $-\lambda$ . Mean responses under the null hypothesis under the considered time trend patterns are illustrated in Fig. 3.1.



, for  $\lambda > 0$ ,

### 3.1. Considered Designs

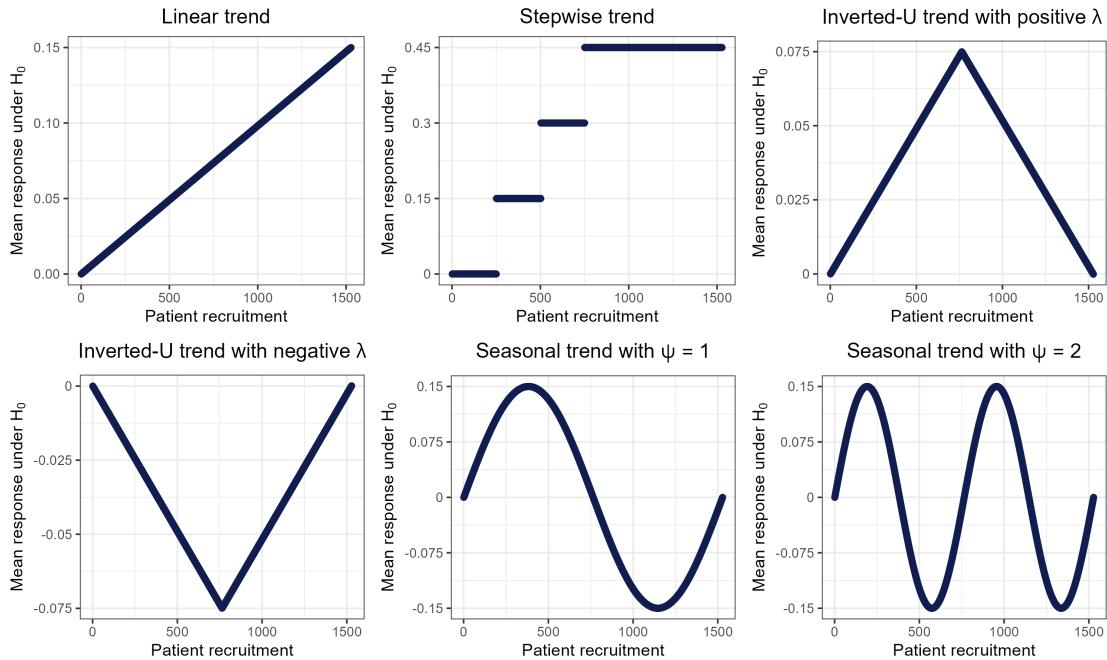


Figure 3.1.: Mean responses under the null hypothesis under time trends of different patterns and strength of  $\lambda = 0.15$ .

*In all scenarios,*

*It is assumed that the equality assumption holds, i.e. the strength of the time trend  $\lambda$  is equal for all arms. Hence, the time trend affects all arms in the same way.*

*pattern fl.) and*

In all cases we assume an underlying response of zero for the control arm ( $\eta_0 = 0$ ) and a treatment effect of  $\theta_k = 0.25$  for treatment arms under the alternative hypothesis, as well as sample sizes of  $n = 250$  for all experimental treatment arms. This treatment effect was chosen such that the separate approach achieves approximately 80% power with the given sample size at a one-sided significance level  $\alpha = 0.025$ . ✓

The strength of the time trend  $\lambda$ , the timings of adding individual treatment arms  $d$ , as well as the length of the calendar time units are varied across the scenarios in order to investigate the impact of these parameters on the considered metrics.

Table 3.1 summarizes the considered simulation settings and parameters. We simulated 10,000 replicates of each scenario to estimate the type I error rate and statistical power. The prediction intervals for an underlying type I error of 0.025 and power of 80% based on 10,000 simulation runs are [0.02194, 0.02806] and [0.79663; 0.80337], respectively.

*I would round them to 3rd dec.*

### 3. Simulations and Results

Setting	$K$	$d$	$\lambda$	Trend pattern	Calendar time unit size	Objective
<b>1</b>	10	$d_i = d \cdot (i - 1)$ with $d \in [0, 500]$ with increments of 25; for $i \in \{1, \dots, 10\}$	[-0.5, 0.5] with increments of 0.125	Linear	-	Evaluate the generalization of the model-based approach with period adjustment
<b>2</b>	4	$d_i = 250 \cdot (i - 1)$ for $i \in \{1, \dots, 4\}$	[-0.5, 0.5] with increments of 0.125	Linear, stepwise, inverted-U, seasonal	[25, 750] with increments of 25	Evaluate the definition of time as calendar time intervals
<b>3</b>	4	$d_i = 250 \cdot (i - 1)$ for $i \in \{1, \dots, 4\}$	[-0.5, 0.5] with increments of 0.125	Linear, stepwise, inverted-U, seasonal	100	Evaluate the mixed models
<b>4</b>	7	$\mathbf{d} = (0, 250, 250,$ $500, 500, 750, 750)$	[-0.5; 0.5] with increments of 0.125	Linear, stepwise, inverted-U, seasonal	100	Evaluate the spline regression

Table 3.1.: Simulation settings and parameters considered in the simulation study.

## 3.2. Results

We present the results in four sections, corresponding to the four different aims of the simulation study, as outlined in Section 3.1. For each aim, we consider a corresponding simulation scenario.

Firstly, in Section 3.2.1, we evaluate the extension the model-based approach with period adjustment, described in Section 2.2.1 in equation (2.5) and assess in particular the impact of overlaps between arms on the operating characteristics of the model.

In Section 3.2.2, we evaluate the calendar time definition of the time covariate in the fixed effect model, given in Section 2.2.1 in equation (2.6). Here, we compare the performance of models with calendar time and period adjustment and discuss the optimal length of the calendar time intervals.

Sections 3.2.3 and 3.2.4 show the performance of the newly proposed flexible modeling approaches for incorporating non-concurrent controls, i.e. linear mixed models from Section 2.2.2 and spline regression described in Section 2.2.3, respectively. The models are evaluated under different patterns and strengths of the time trend.

For all plots showing the statistical power, we restricted the range of the y-axis to [0.7, 1] for better legibility. If the power is not visible for some values on the x-axis, it means that the estimated power was below 0.7. In all plots showing the estimated type I error rate, we include a dashed reference line for the nominal significance level of 0.025 and the prediction interval around this value as a grey box.

### 3.2. Results

#### 3.2.1. Setting 1: Extension of Regression Model to Trials with Multiple Arms

Consider a platform trial with 10 experimental treatment arms and a shared control arm, where the experimental arm  $i$  enters the trial after  $d_i = d \cdot (i - 1)$  patients have been recruited to the platform, as illustrated in Fig. 3.2. Note that  $d$  determines the amount of overlapping sample size between the treatment arms. If  $d = 0$ , all arms join and leave the trial simultaneously, resulting in a standard multi-arm trial and a total overlap between the arms. If  $d = 2n$ , a new treatment arm enters the trial once the previous one finishes, so that there is only one active experimental arm at a time, and hence no overlap between them.

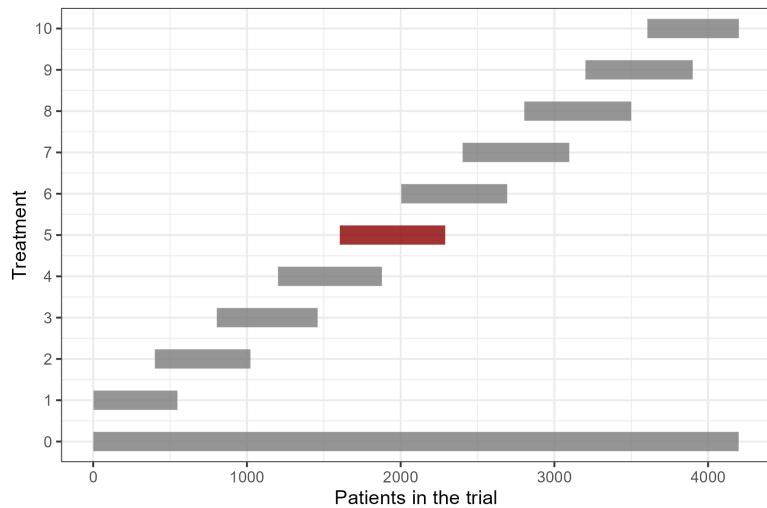


Figure 3.2.: Illustration of the scenario with 10 experimental arms. In this case, arm  $i$  enters after  $d_i = 400 \cdot (i - 1)$  patients have been recruited to the trial. We focus on evaluating the efficacy of the 5th experimental treatment arm, highlighted in the figure, compared to the shared control group. Results for other experimental arms can be found in the Appendix A.1.

We vary the time trend and timing of adding arms as indicated in Table 3.1 in order to evaluate the effect of the overlap and the strength and pattern of the time trend on the type I error rate and statistical power. We compare the linear regression model with period adjustment to the pooled and separate analysis approaches.

Figure 3.3 shows the impact of the strength of the time trend on the operating characteristics when evaluating the 5th experimental arm. The regression model, as well as the separate analysis asymptotically control the type I error rate, regardless of the strength of the time trend. The pooled analysis leads to inflation of the type I error in the presence of positive time trends and deflation in case of negative time trends. Although only results for linear time trends are shown, the regression model maintains the type I error rate under arbitrary time trend pattern. Additionally, the model leads to gains in

for unblinding, maybe:  
→ "treatment arms."

### 3. Simulations and Results

power as compared to the separate analysis.

The effect of the amount of overlapping sample size is shown in Figure 3.4. The overlaps have no effect on the type I error rate control by the regression model and separate approach, which is guaranteed (asymptotically) in all the cases. The inflation of the pooled analysis gets stronger with increasing  $d$ . This is because larger  $d$ 's result in longer platform trials and larger size of the NCC data. The power of the regression model, however, depends on the overlap between treatment arms. In the extreme cases with  $d = 0$  and  $d = 2n$ , the regression model leads to identical power as the separate analysis. If  $d = 0$ , there is no NCC data, as all the arms join the trial at the beginning. Thus, the control group used for the treatment-control comparison is the same for both, the separate analysis and the regression model. In case of no overlap between the arms ( $d = 2n = 500$ ), there is insufficient amount of data to estimate the period effect. Hence, simultaneous presence of the experimental arms in the trial is crucial for a reliable estimation of the period effect and resulting power gains when using the regression model. The maximal power is reached for  $d = 175$ , which in this case leads to the optimal trade-off between the amount of overlapping sample size between treatment arms and the size of the non-concurrent data.

Figure 3.5 illustrates how the type I error rate and power for individual treatment-control comparisons depends on the order of entry in the platform trial. We observe that the power of the regression model increases for arms that were added to the trial later. This is due to larger sample size of the NCC data. On the other hand, this also leads to a higher inflation of the type I error with the pooled analysis.

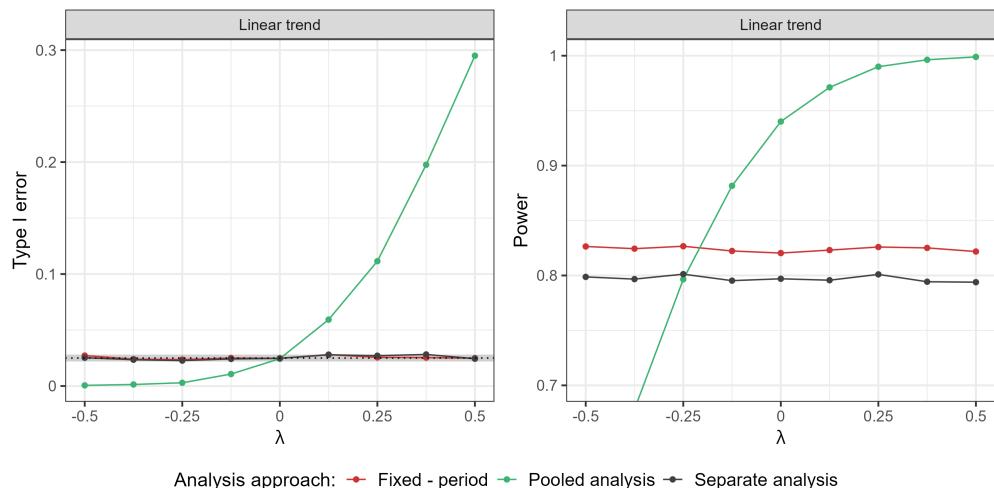


Figure 3.3.: Type I error rate and power of the fixed effect regression model with period adjustment compared to the pooled and separate analyses with respect to the strength of the time trend  $\lambda$ . In this example,  $d = 400$ ,  $n = 250$  in each experimental arm and a linear shape of the time trend are used and the 5th experimental arm is being evaluated.

### 3.2. Results

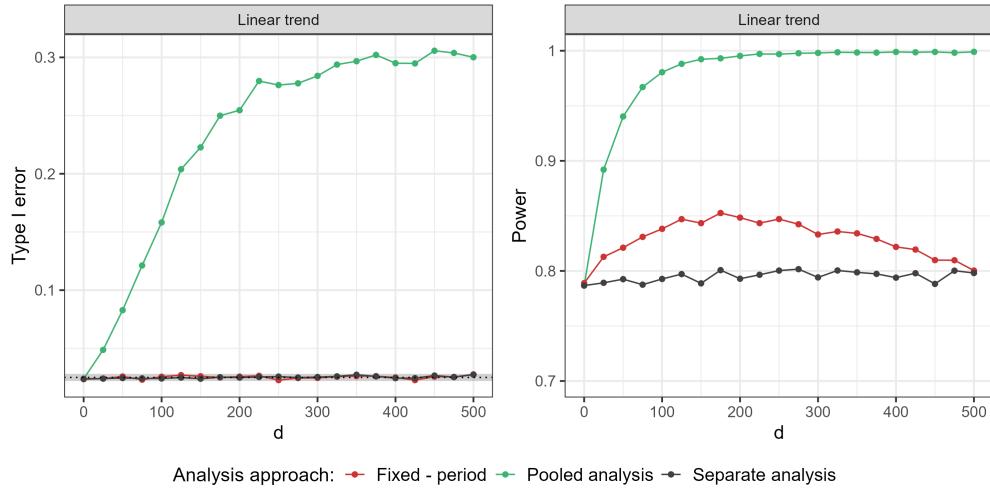


Figure 3.4.: Type I error rate and power of the fixed effect regression model with period adjustment compared to the pooled and separate analyses with respect to the timing of adding the treatment arms. In this example,  $n = 250$  in each experimental arm and a linear time trend with strength  $\lambda = 0.5$  are considered and the 5th experimental arm is being evaluated.

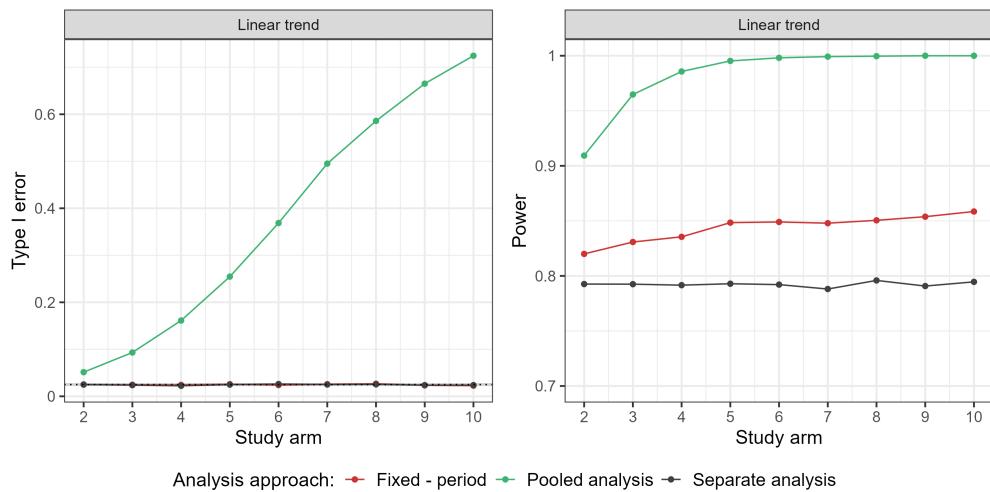


Figure 3.5.: Type I error rate and power of the regression model with period adjustment compared to the pooled and separate analyses with respect to the index of the evaluated arm. In this example,  $d = 200$ ,  $n = 250$  in each experimental arm and a linear time trend with strength  $\lambda = 0.5$  are considered.

### 3. Simulations and Results

#### 3.2.2. Setting 2: Alternative Definition of the Time Covariate

To examine the regression model with calendar time adjustment, we consider a platform trial with 4 experimental treatment arms, where arm  $i$  enters after  $d_i = 250 \cdot (i - 1)$ , leading to a total sample size of 1528 patients. The considered scenario is shown in Figure 3.6. We assess how the model performance depends on the pattern and strength of the time trend, as well as on the chosen calendar time length.

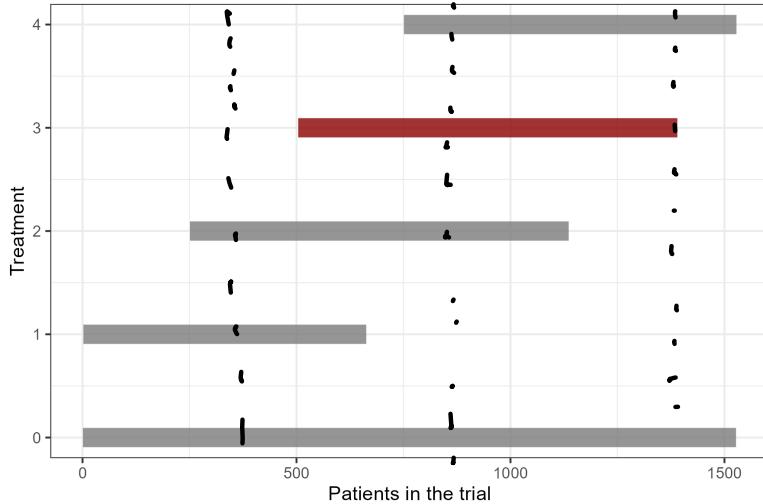


Figure 3.6.: Illustration of the scenario with 4 experimental arms, where arm  $i$  enters after  $d_i = 250 \cdot (i - 1)$  patients have been recruited to the trial. We focus on evaluating the efficacy of the 3rd experimental treatment arm, highlighted in the figure, compared to the shared control group. Results for other experimental arms can be found in the Appendix A.1.

In Figures 3.7 and 3.8 the type I error and power of the model with the calendar time adjustment is compared to the period adjustment and the separate analysis under varying unit size and  $\lambda$ , respectively. For Figure 3.7, unit sizes in the range [25, 750] with increments of 25 were considered. The type I error rate control for different time unit lengths is dependent on the time trend pattern. In case of linear and inverted-U trend, the type I error rate is maintained for moderately sized calendar time units, and slightly inflated for units larger than 600 patients. Under the seasonal time trend, we observe deflation of the type I error for unit sizes  $\geq 200$ . In case of the stepwise trend, the type I error rate control is only given for very small units ( $< 50$  patients). Depending on the unit size, the model with calendar time adjustment can lead to power improvements as compared to the period adjustment, especially in the setting with inverted-U or linear time trend. Nevertheless, as the type I error rate is not maintained in all cases, the choice of the interval length and the resulting trade-off between type I and type II errors needs to be carefully assessed.

→ can we explain further this pattern?

? → More or in the discussion, mention a design in which treat. arms are only allowed to enter when a calendar time int. starts.

### 3.2. Results

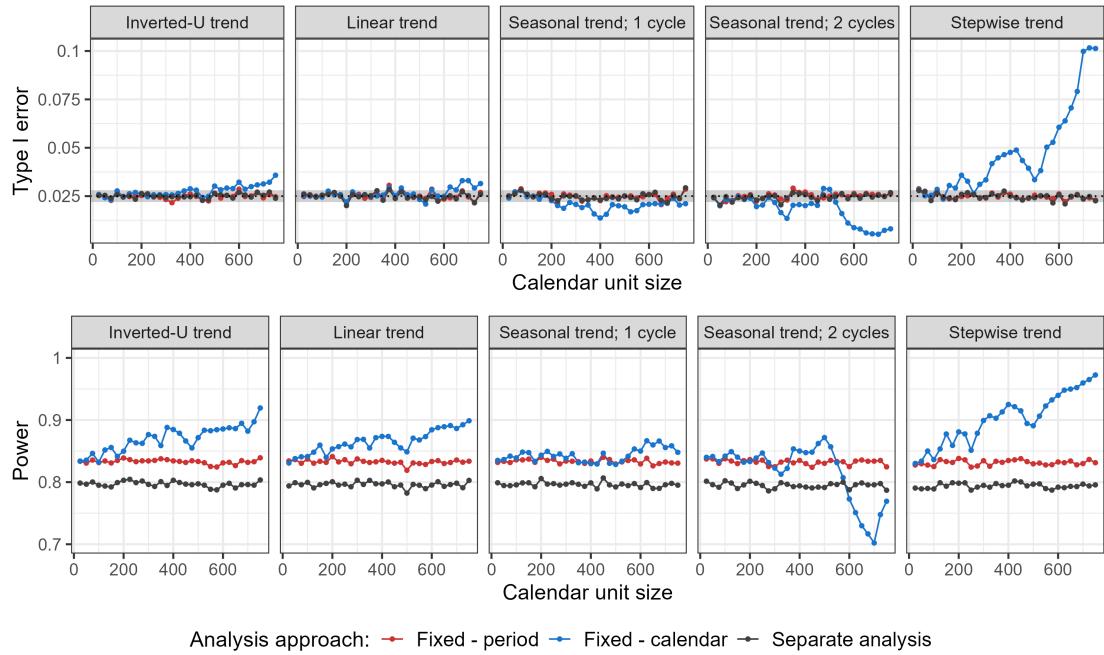


Figure 3.7.: Type I error rate and power of the regression model with calendar time adjustment compared to the regression model with period adjustment and separate analysis with respect to the size of the calendar time unit under different time trend patterns. In this example,  $n = 250$  in each experimental arm,  $\lambda = 0.125$  and  $N_p = 750$  in case of inverted-U trend (corresponding approximately to the middle of the trial) are considered and the 3rd experimental arm is being evaluated.

### 3. Simulations and Results

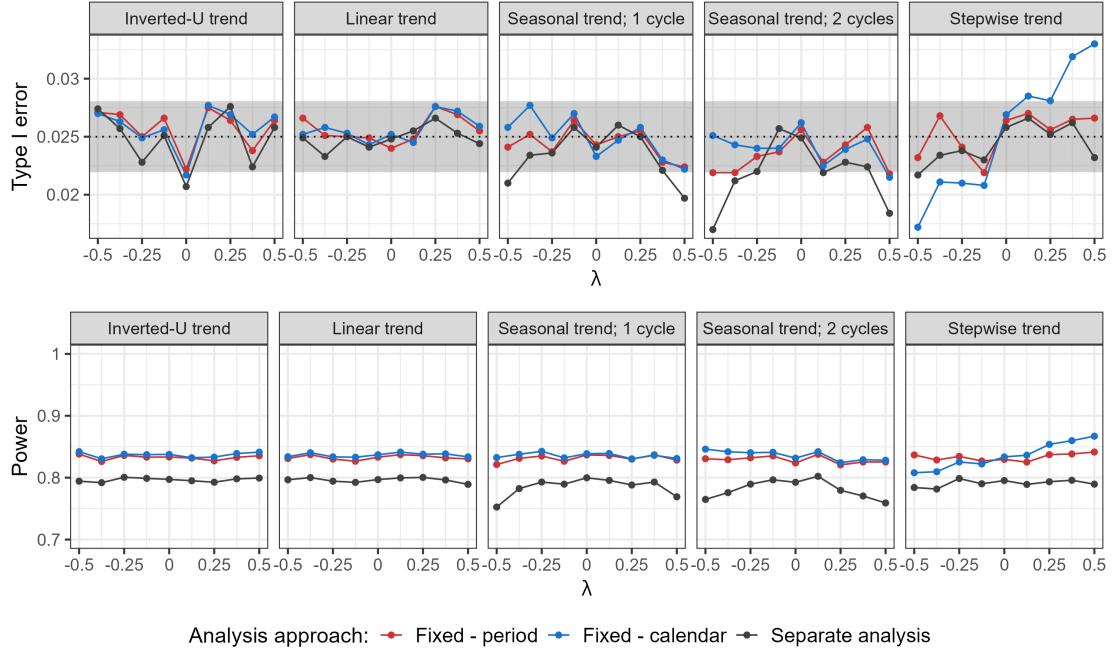


Figure 3.8.: Type I error rate and power of the regression model with calendar time adjustment compared to the regression model with period adjustment and separate analysis with respect to the strength of the time trend  $\lambda$  under different time trend patterns. In this example,  $n = 250$  in each experimental arm,  $N_p = 750$  in case of inverted-U trend (corresponding approximately to the middle of the trial) and calendar time unit length of 100 are used and the 3rd experimental arm is being evaluated.

#### 3.2.3. Setting 3: More Flexible Modelling Approaches: Mixed Models

Also for a setting with four experimental treatment arms (for illustration see Figure 3.6), we analyze the performance of the proposed variants of linear mixed models that include the time covariate as a random effect. We consider varying time trend patterns and strengths and present the resulting type I error rate and power of the mixed models in Figure 3.9, along with the reference model - fixed effect model with period adjustment - and the separate analysis.

We observe that the mixed models only maintain the type I error rate if no time trends are present in the trial (i.e.,  $\lambda = 0$ ). Under this assumption, they also lead to power improvement compared to the fixed effect regression model. However, in case of time trends, the type I error rate control is lost. The inflation (or deflation) of the type I error is most pronounced in the mixed model that adjusts for calendar time intervals as uncorrelated random effects. This is because the variance of the random effects is underestimated, since less observations are available in each calendar time interval, and

### 3.2. Results

the observations within one interval are more similar to each other than it is the case for the periods. Moreover, especially in settings with seasonal and stepwise trend, the maximum inflation seems to be achieved for moderately strong time trends. The reason for this is that the mixed models shrink the effect of the time trend, since it is modelled as a random effect. If, however, the trend is large enough, there is less shrinkage and the time effect is preserved and better adjusted for.

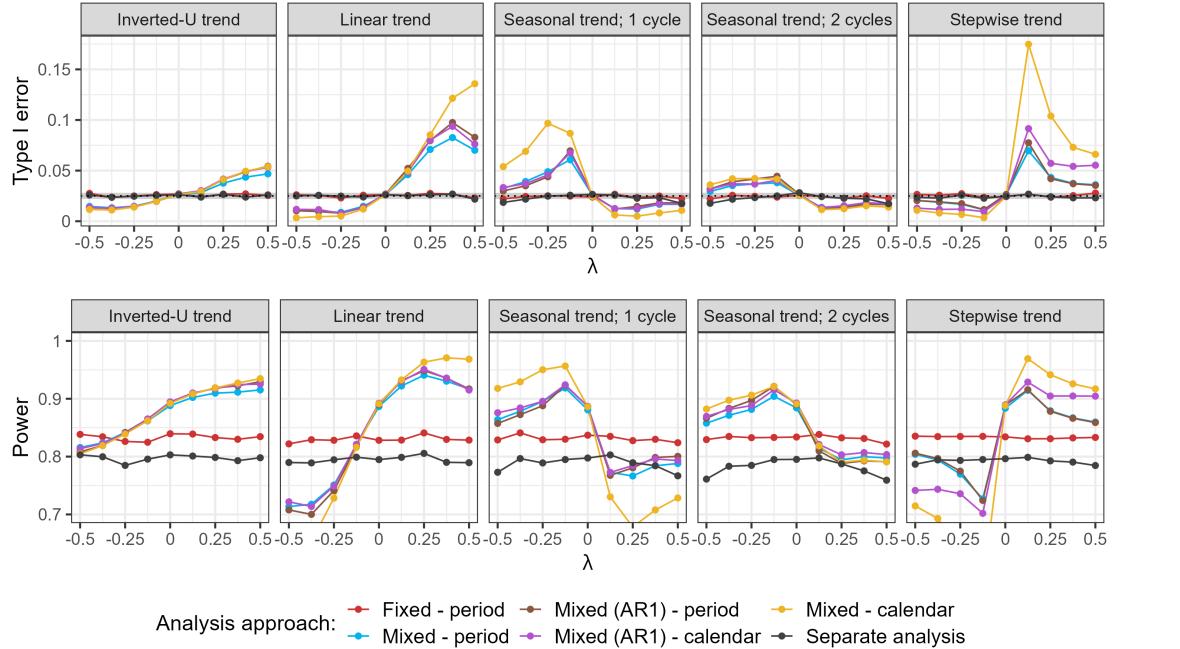


Figure 3.9.: Type I error rate and power of the mixed model with period and calendar time adjustments as uncorrelated and autocorrelated random effects, compared to the fixed effect regression model with period adjustment with respect to the pattern and strength of the time trend. Sample sizes of  $n = 250$  in each experimental arm and  $N_p = 750$  in case of inverted-U trend (corresponding approximately to the middle of the trial) are assumed. In case of calendar time adjustment, unit size of 100 patients is considered. The 3rd experimental arm is being evaluated.

#### 3.2.4. Setting 4: More Flexible Modelling Approaches: Spline Regression

The performance of the spline regression models is assessed in a scenario with 7 experimental treatment arms, where some of them enter and leave the trial simultaneously (see Figure 3.10 for illustration). The models are examined under different time trend patterns with varying strength, as described in Table 3.1.

In our simulations, we also varied the degree of the B-splines, considering linear,

### 3. Simulations and Results

quadratic and cubic splines. However, since the difference in the resulting operating characteristics were only marginal, here we only present results for the cubic spline regression, where the knots are placed either at the beginning of each period or each calendar time interval. Results for linear and quadratic splines can be found in Appendix A.1. Again, we use the fixed effects model with period adjustment as reference.

The type I error rate and power of the evaluated models under the considered scenarios are presented in Figure 3.11. If the time trend pattern is given by a continuous function, the spline regression maintains the type I error rate. However, in case of the stepwise time trend, we observe an inflation in the type I error rate, particularly pronounced when placing the knots according to the periods. This is because the spline regression estimates the time effect by a smooth function, composed of multiple polynomial functions joined together in the knots. This is not an optimal approach if there are sudden jumps in the time trend. In the considered scenarios, the power of the spline regression models was not improved as compared to the fixed effect model.

The power in cases with continuous time trend function is slightly improved as compared to the fixed effects model, however, the differences are only marginal.

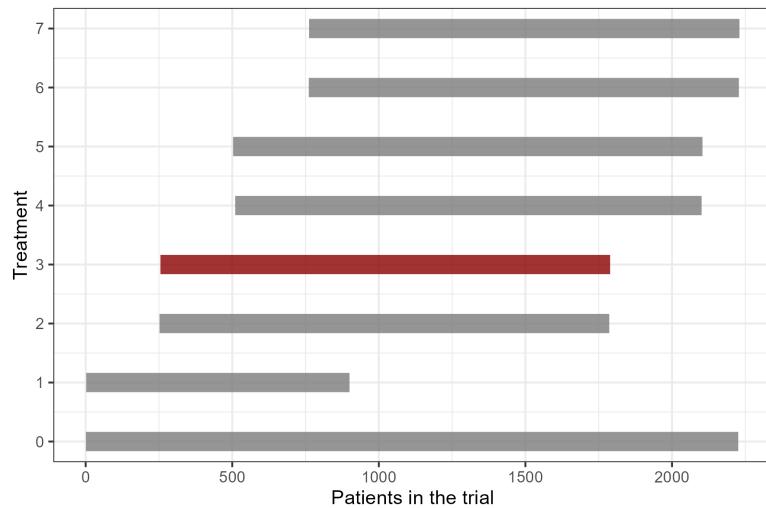


Figure 3.10.: Illustration of the scenario with 7 experimental arms, where the experimental treatment arms enter after  $d = (0, 250, 250, 500, 500, 750, 750)$  patients have been recruited to the trial. We focus on evaluating the efficacy of the 3rd experimental treatment arm, highlighted in the figure, compared to the shared control group. Results for other experimental arms can be found in the Appendix A.1.

### 3.2. Results

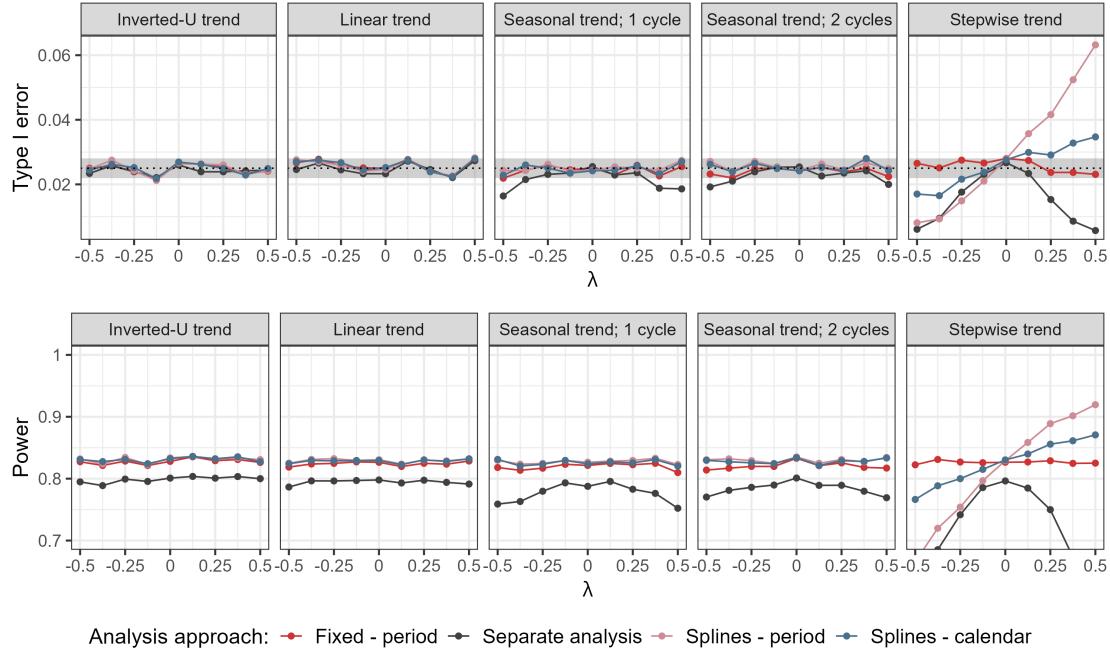


Figure 3.11.: Type I error rate and power for the cubic spline regression model with knots according to periods or calendar time units compared to the regression model with period adjustment with respect to the strength of the time trend  $\lambda$  using different time trend patterns. In this example,  $n = 250$  in each experimental arm and  $N_p = 1115$  in case of inverted-U trend (corresponding to the middle of the trial) are used. In case of calendar time adjustment, unit size of 100 patients is considered. The 3rd experimental arm is being evaluated.



## 4. Software

In this chapter, we introduce the R-package NCC, which allows for simulation and analysis of platform trials with non-concurrent controls using analysis methods introduced in Chapter 2. Especially, this package includes not only the models proposed in Section 2.2, but also the frequentist models (see Section 2.1.1), the Bayesian Time Machine (see Section 2.1.2) and the Meta-Analytic-Predictive (MAP) Prior approach. Moreover, the package contains functions for data generation and wrapper functions for running simulation studies.

The NCC package is available on CRAN and has already been presented in an article, where its functionalities are described. The article entitled “*NCC: An R-package for analysis and simulation of platform trials with non-concurrent controls*” is currently under review in a journal and available on arXiv [28].

In Section 4.1 we describe how the package is structured, which functions it contains and their respective input and output values.

Section 4.2 shows specific examples how to use the NCC R-package to generate and analyze platform trial data and run simulation studies.

### 4.1. Software Description

We present the R-package NCC [29], which was developed for assessing the operating characteristics of analysis methods that utilize non-concurrent controls in the analysis of platform trials. The NCC package provides functions to simulate platform trials with continuous or binary endpoints, as well as functions to analyse the trial data using various approaches, allowing for the incorporation of NCC data. It can be installed either from CRAN (Comprehensive R Archive Network) or Github using the following commands:

```
> # devtools::install_github("pavlakrotka/NCC") # development version  
> install.packages("NCC") # stable version  
> library(NCC)
```

The package comes with an accompanying website with background explanations and short tutorials: <https://pavlakrotka.github.io/NCC/>.

The NCC package focuses on trials with continuous or binary endpoints and consists of 34 functions. Some of the functions are implemented for continuous endpoints and some for binary endpoints. The functions with the suffix `_cont` refer to functions for simulation and analysis of trials with continuous endpoints, while `_bin` refers to binary endpoints. The NCC functions can be grouped into three main categories according to

#### 4. Software

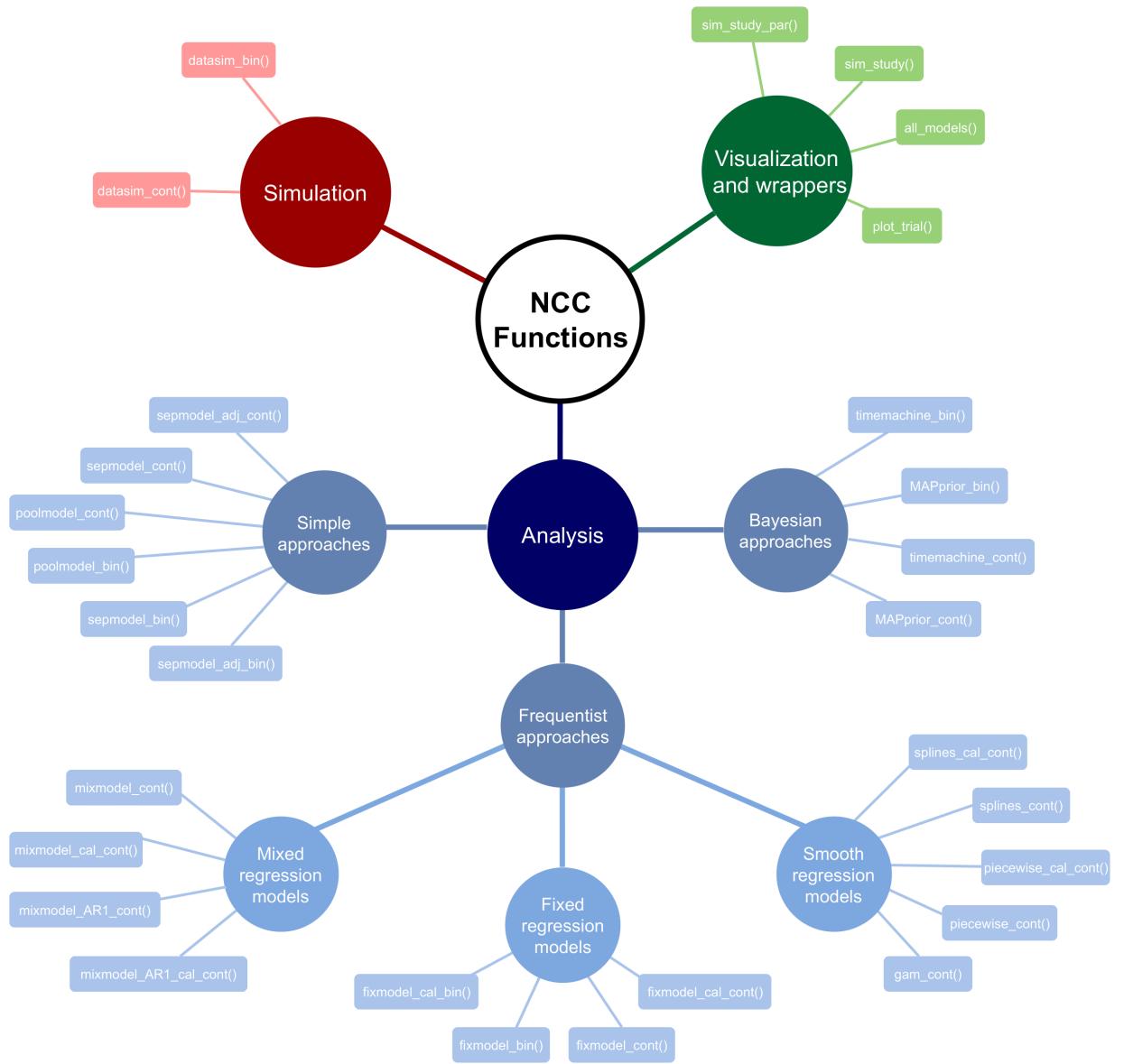


Figure 4.1.: Scheme of the NCC package functions by functionality. Auxiliary functions for data generation are omitted in this figure.

#### 4.1. Software Description

Function	Description	Functionality
<code>datasim_cont()</code>	Simulates trials with continuous endpoints	Data simulation
<code>datasim_bin()</code>	Simulates trials with binary endpoints	Data simulation
<code>get_ss_matrix()</code>	Computes sample sizes per arm and period	Data simulation
<code>linear_trend()</code>	Generates a linear time trend	Data simulation
<code>sw_trend()</code>	Generates a step-wise time trend	Data simulation
<code>inv_u_trend()</code>	Generates a inverted-u time trend	Data simulation
<code>seasonal_trend()</code>	Generates a seasonal time trend	Data simulation
<code>fixmodel_bin()</code>	Performs analysis using a regression model adjusting for periods for binary data	Data analysis
<code>fixmodel_cont()</code>	Performs analysis using a regression model adjusting for periods for continuous data	Data analysis
<code>fixmodel_cal_bin()</code>	Performs analysis using a regression model adjusting for calendar times for binary data	Data analysis
<code>fixmodel_cal_cont()</code>	Performs analysis using a regression model adjusting for calendar times for continuous data	Data analysis
<code>poolmodel_bin()</code>	Performs pooled analysis for binary data	Data analysis
<code>poolmodel_cont()</code>	Performs pooled analysis for continuous data	Data analysis
<code>sepmodel_bin()</code>	Performs separate analysis for binary data	Data analysis
<code>sepmodel_cont()</code>	Performs separate analysis for continuous data	Data analysis
<code>mixmodel_cont()</code>	Performs analysis using a mixed model adjusting for periods as a random factor for continuous data	Data analysis
<code>mixmodel_cal_cont()</code>	Performs analysis using a mixed model adjusting for calendar times as a random factor for continuous data	Data analysis
<code>mixmodel_AR1_cont()</code>	Performs analysis using a mixed model adjusting for periods as a random factor with AR1 correlation structure for continuous data	Data analysis
<code>mixmodel_AR1_cal_cont()</code>	Performs analysis using a mixed model adjusting for calendar times with AR1 correlation structure as a random factor for continuous data	Data analysis
<code>splines_cont()</code>	Performs analysis using regression splines with knots placed according to periods for continuous data	Data analysis
<code>splines_cal_cont()</code>	Performs analysis using regression splines with knots placed according to calendar times for continuous data	Data analysis
<code>plot_trial()</code>	Visualizes the simulated trial over time	Data visualization
<code>sim_study_par()</code>	Performs a simulation study with given scenarios	Wrapper function

Table 4.1.: Main functions of the NCC package with a short description.

their functionality: data simulation, analysis, and visualization and wrappers. See Table 4.1 for a summary of the main functions. Figure 4.1 outlines the package structure.

The functions `datasim_cont()` and `datasim_bin()` refer to the simulation of patient data from a platform trial; functions such as `fixmodel_cont()`, `mixmodel_cont()` or `splines_cont()` are functions devoted to comparing the efficacy of an experimental treatment versus control using concurrent and non-concurrent controls in trials with continuous endpoints. Finally, the functions `plot_trial()` and `sim_study_par()` are intended to visualise the generated trials and perform simulation studies, respectively.

Most functions in the NCC package use common arguments. The main arguments are briefly described in Table 4.2, together with the functions which rely on them.

#### 4. Software

Argument	Description	Functions
<code>num_arms</code>	Number of treatment arms in the trial	<code>datasim_bin()</code> , <code>datasim_cont()</code>
<code>n_arm</code>	Sample size per experimental treatment arm	<code>datasim_bin()</code> , <code>datasim_cont()</code>
<code>d</code>	Timings of adding new arms in terms of number of patients recruited to the trial	<code>datasim_bin()</code> , <code>datasim_cont()</code>
<code>p0</code>	Response in the control arm for platform trials with binary endpoints	<code>datasim_bin()</code>
<code>mu0</code>	Response in the control arm for platform trials with continuous endpoints	<code>datasim_cont()</code>
<code>OR</code>	Odds ratios for each treatment arm compared to control	<code>datasim_bin()</code>
<code>theta</code>	Treatment effects for each treatment arm	<code>datasim_cont()</code>
<code>sigma</code>	Standard deviation of the responses	<code>datasim_cont()</code>
<code>lambda</code>	Strength of time trend in each arm	<code>datasim_bin()</code> , <code>datasim_cont()</code>
<code>trend</code>	Time trend pattern	<code>datasim_bin()</code> , <code>datasim_cont()</code>
<code>data</code>	Trial data, e.g. generated with the <code>datasim_*</code> () functions	<code>fixmodel_bin()</code> , <code>fixmodel_cont()</code> , <code>fixmodel_cal_bin()</code> , <code>fixmodel_cal_cont()</code> , <code>mixmodel_cont()</code> , <code>mixmodel_cal_cont()</code> , <code>mixmodel_AR1_cont()</code> , <code>mixmodel_AR1_cal_cont()</code> , <code>splines_cont()</code> , <code>splines_cal_cont()</code> , <code>sepmodel_bin()</code> , <code>sepmodel_cont()</code> , <code>poolmodel_bin()</code> , <code>poolmodel_cont()</code>
<code>arm</code>	Treatment arm under study to perform inference on	<code>fixmodel_bin()</code> , <code>fixmodel_cont()</code> , <code>fixmodel_cal_bin()</code> , <code>fixmodel_cal_cont()</code> , <code>mixmodel_cont()</code> , <code>mixmodel_cal_cont()</code> , <code>mixmodel_AR1_cont()</code> , <code>mixmodel_AR1_cal_cont()</code> , <code>splines_cont()</code> , <code>splines_cal_cont()</code> , <code>sepmodel_bin()</code> , <code>sepmodel_cont()</code> , <code>poolmodel_bin()</code> , <code>poolmodel_cont()</code>
<code>alpha</code>	Significance level	<code>fixmodel_bin()</code> , <code>fixmodel_cont()</code> , <code>fixmodel_cal_bin()</code> , <code>fixmodel_cal_cont()</code> , <code>mixmodel_cont()</code> , <code>mixmodel_cal_cont()</code> , <code>mixmodel_AR1_cont()</code> , <code>mixmodel_AR1_cal_cont()</code> , <code>splines_cont()</code> , <code>splines_cal_cont()</code> , <code>sepmodel_bin()</code> , <code>sepmodel_cont()</code> , <code>poolmodel_bin()</code> , <code>poolmodel_cont()</code>
<code>ncc</code>	Whether to include NCC data into the analysis	<code>fixmodel_bin()</code> , <code>fixmodel_cont()</code> , <code>fixmodel_cal_bin()</code> , <code>fixmodel_cal_cont()</code> , <code>mixmodel_cont()</code> , <code>mixmodel_cal_cont()</code> , <code>mixmodel_AR1_cont()</code> , <code>mixmodel_AR1_cal_cont()</code> , <code>splines_cont()</code> , <code>splines_cal_cont()</code> , <code>sepmodel_bin()</code> , <code>sepmodel_cont()</code> , <code>poolmodel_bin()</code> , <code>poolmodel_cont()</code>
<code>unit_size</code>	Number of patients per calendar time unit	<code>fixmodel_cal_bin()</code> , <code>fixmodel_cal_cont()</code> , <code>mixmodel_cal_cont()</code> , <code>mixmodel_AR1_cal_cont()</code> , <code>splines_cal_cont()</code>
<code>bs_degree</code>	Degree of the polynomial spline	<code>splines_cont()</code> , <code>splines_cal_cont()</code>

Table 4.2.: Main input arguments together with a short description and functions using these arguments. Detailed explanations can be found at <https://pavlakrotka.github.io/NCC/>.

As this thesis deals with platform trials with continuous endpoints, we will focus on describing the functions for this type of endpoints. Description of functions for binary endpoints can be found in the package manual (see Appendix A.2) and on the package website (<https://pavlakrotka.github.io/NCC/>).

### 4.1.1. Data Simulation

Platform trials with a continuous outcome are simulated using the function `datasim_cont()`, as follows:

```
> datasim_cont(num_arms, n_arm, d, period_blocks = 2, mu0 = 0,
               theta, lambda, sigma, trend, N_peak, n_wave,
               full = FALSE, check = TRUE)
```

The arguments are the number of experimental treatment arms (`num_arms`), as well as their sample size (`n_arm`), timings of entering the trial in terms of patients already enrolled in the trial to this point (`d`) and treatment effects in terms of the difference in means (`theta`). The control mean is given by the argument `mu0` and has a default value of 0. The standard deviation of the responses is specified by the input argument `sigma` and is assumed to be equal for all arms. Moreover, it is assumed that the sample sizes in each experimental arm are equal. Patients are assigned to the arms according to block randomization using an allocation ratio of 1:1....:1 in each period. For each period, the block size equals to `period_blocks`·(number of arms active in that period) is used. For instance, if `period_blocks=2`, in a period with 2 experimental treatment arms and one control arm, the resulting block size would be 6. The function allows to simulate trial data in the presence of time trends of different patterns and strengths, which can also be specified in the input arguments. The time trend pattern can be specified by means of the argument `trend`, choosing from the options linear, stepwise, inverted-U with a peak at time `N_peak` and seasonal with `n_wave` cycles, while the strength of the trend is indicated by the argument `lambda`. Figure 4.2 shows how the mean response changes under different time trend patterns when varying the parameters `lambda`, `N_peak` and `n_wave`. The argument `full` specifies if the output is given solely in the form of a data frame (if `full=FALSE`) with the trial data, or if the full output is provided in the form of a list, including the trial data and additional information (`full=TRUE`). Finally, `check` is an indicator of whether the input parameters are checked to ensure that they are correctly specified. If `check=TRUE`, the function returns helpful error messages in case of wrong input.

By default, the function returns the simulated trial data in the form of a data frame containing the following columns:

- `j` - patient recruitment index
- `response` - response for patient  $j$
- `treatment` - indicator of the treatment patient  $j$  was allocated in
- `period` - indicator of the period in which patient  $j$  was recruited in

Simulation of binary endpoints is performed analogously, using the function `datasim_bin()`, which only differs in the indication of the control response (argument `p0`) and the treatment effects (argument `OR`), which are specified in terms of the odds ratio.

#### 4. Software

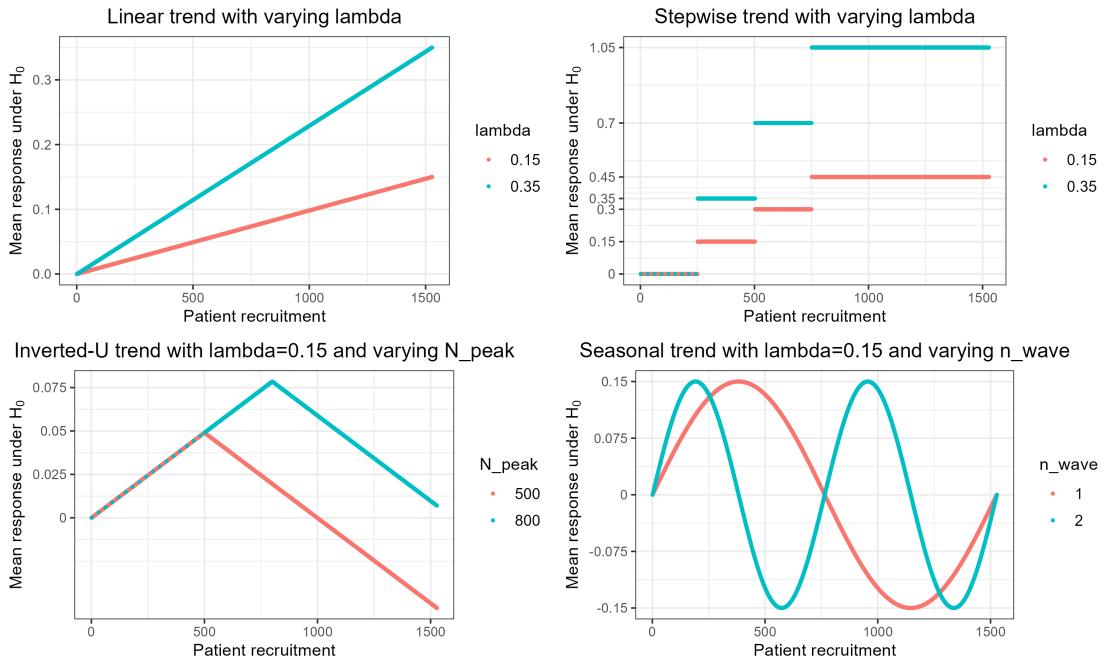


Figure 4.2.: Time trend patterns with varying parameters.

#### 4.1.2. Analysis Approaches

The main frequentist analysis approaches for continuous data implemented in the NCC package are the fixed and mixed effects, as well as spline regression models with adjustment either based on the periods or calendar times. The arguments common to all analysis functions in the NCC package are the data frame with the trial data, consisting of columns named "j", "response", "treatment" and "period" (**data**), the indicator of the experimental treatment arm that should be compared to the control group (**arm**) and the significance level (**alpha**).

To analyze the data using the frequentist fixed effects model from equation (2.5), one can use the **fixmodel\_cont()** function as follows:

```
> fixmodel_cont(data, arm, alpha = 0.025, ...)
```

The function **mixmodel\_cont()** permits to analyse the data using the mixed effects model described by equation (2.7) by means of:

```
> mixmodel_cont(data, arm, alpha = 0.025, ...)
```

To fit the spline regression model from equation (2.10), one can use the function **splines\_cont()** with the following syntax:

```
> splines_cont(data, arm, alpha = 0.025, bs_degree = 3, ...)
```

#### 4.1. Software Description

where the parameter `bs_degree` indicates the degree of used the polynomial spline, with a default value of 3 for a cubic spline.

The above presented functions use the definition of time in terms of periods in the trial. Hence, the fixed and mixed models include the factor period and the spline regression places the inner knots at the beginning of each period. Analogous functions for the calendar time adjustment, indicated by the extension `_cal` (e.g. `fixmodel_cal_cont()`), are also implemented in the package.

The functions perform the respective analysis of the given dataset to compare the efficacy of a specific treatment against control, thus testing the null hypothesis for the treatment effect of the arm under study  $H_0 : \theta_{\text{arm}} = 0$  against the one-sided alternative  $H_1 : \theta_{\text{arm}} > 0$ . By default, to test  $H_0$ , all trial data until the evaluated treatment arm leaves the trial are taken into account (i.e., also including data from unfinished arms that joined the platform up to the final analysis of the given treatment arm).

The output of the analysis functions is in the form of a list, containing the one-sided p-value, estimated treatment effect,  $(1-2\cdot\alpha)\cdot100\%$  confidence interval (i.e., 95% CI if `alpha=0.025`), an indicator of whether the null hypothesis was rejected or not, and the fitted model. Functions for spline regression additionally output the position of the inner knots in terms of patient index.

##### 4.1.3. Trial Data Visualization and Wrapper Functions

The package also includes functions to visualise the platform trial data and wrapper functions for performing simulation studies under different scenarios .

The visualization function `plot_trial()` uses as argument a vector with treatment indicators ordered by time (`treatments`) and outputs a plot of the trial progress over time, as we will illustrate in Section 4.2. The main wrapper function is `sim_study_par()`, which permits to efficiently run simulation studies using parallel computing. The code is parallelized on replication level, i.e. replications of one scenario are distributed over the available cores. Using this function requires creating a data frame with the desired simulation scenarios beforehand, which is then used as input to the function (argument `scenarios`) as follows:

```
> sim_study_par(nsims, scenarios, arms,
                 models = c("fixmodel", "sepmodel", "poolmodel"),
                 endpoint, perc_cores = 0.9)
```

where the remaining arguments specify how many times each scenario is to be replicated (`nsim`), the treatment arms that will be evaluated (`arms`), the considered analysis approaches (`models`), the type of endpoint (`endpoint`) and the (approximate) percentage of available cores that should be used for the simulations (`perc_cores`). The output of `sim_study_par` is a data frame with all considered scenarios and corresponding results, that is, the probability to reject the null hypothesis, the bias, and the mean squared error (MSE) of the treatment effect estimates for each evaluated treatment arm and each considered analysis method.

#### 4. Software

## 4.2. Examples

### 4.2.1. How to Analyze Platform Trial Data Utilising Non-concurrent Controls

Assume a platform trial with a shared control and three experimental treatment arms entering sequentially, where the second and third treatment arms enter after 100 and 250 patients are recruited to the trial, respectively. Furthermore, assume sample sizes of 100 and treatment effects of 0.25 in each experimental arm, as well as a standard deviation of the responses equal to 1. The mean response for all arms increases by 0.15, whenever a new arm is added to the trial. The data of this hypothetical trial can be simulated using the `datasim_cont()` function:

```
> set.seed(5)
> trial_data <- datasim_cont(num_arms = 3, n_arm = 100, d = c(0, 100, 250),
  theta = rep(0.25, 3), lambda = rep(0.15, 4),
  sigma = 1, trend = "stepwise_2")
```

The generated data is structured as follows:

```
> head(trial_data)
  j      response treatment period
1 1  0.78575816        1       1
2 2 -0.14030110        0       1
3 3 -0.36289414        0       1
4 4 -0.31737256        1       1
5 5  1.41623385        0       1
6 6 -0.04480078        0       1
```

where the patient index is given in the first column, followed by the continuous responses, the treatment arm indicator and finally the period allocation.

In order to illustrate the active treatment arms over time, we use the function `plot_trial()`, whose output is shown in Figure 4.3.

```
> plot_trial(trial_data$treatment)
```

## 4.2. Examples

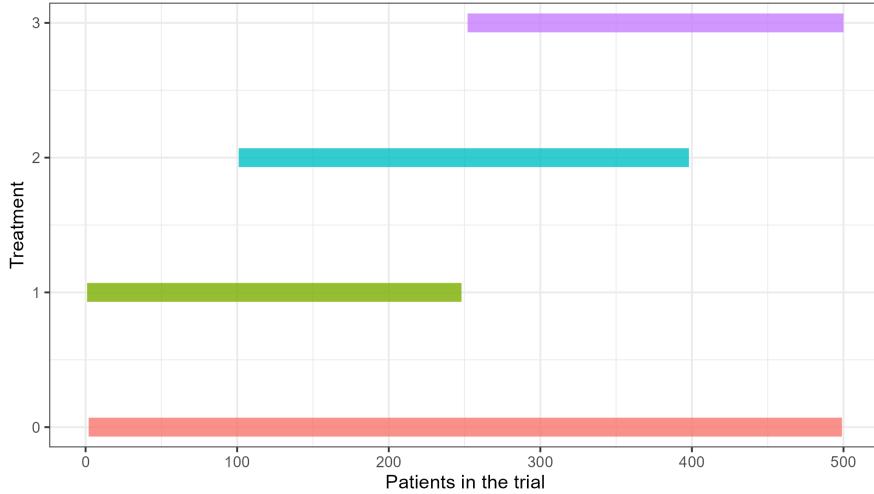


Figure 4.3.: Output of the function `plot_trial()`.

The figure helps to display the entry and exit of arms into and out of the trial over time and to visualize the overlaps between arms more easily.

When the third arm ends, we want to evaluate its efficacy compared to the control using different frequentist model-based approaches. First, we consider a fixed regression model with period adjustment for time trend (as in (2.5)), which is fitted using the `fixmodel_cont()` function:

```
> fixmodel_cont(trial_data, arm = 3, alpha = 0.025)
$p_val
[1] 0.01816827
$treat_effect
[1] 0.2790546
$lower_ci
[1] 0.01782669
$upper_ci
[1] 0.5402824
$reject_h0
[1] TRUE
```

Analysis using a fixed regression model with calendar time adjustment (given by (2.6)) can be performed by means of the `fixmodel_cal_cont()` function. The length of the calendar time interval to adjust for (in terms of the recruited patients) is given by the argument `unit_size`.

```
> fixmodel_cal_cont(trial_data, arm = 3, unit_size = 25, alpha = 0.025)
$p_val
[1] 0.0218659
```

#### 4. Software

```
$treat_effect
[1] 0.2712733
$lower_ci
[1] 0.007656746
$upper_ci
[1] 0.5348899
$reject_h0
[1] TRUE
```

A mixed effect model with period adjustment that assumes uncorrelated random effects (see equation (2.7)) is implemented in the `mixmodel_cont()` function:

```
> mixmodel_cont(trial_data, arm = 3, ci = TRUE, alpha = 0.025)
$p_val
[1] 0.00231647
$treat_effect
[1] 0.3392225
$lower_ci
[1] 0.1059671
$upper_ci
[1] 0.572476
$reject_h0
[1] TRUE
```

Finally, if we want to adjust for time using cubic splines (see equation (2.10)), we can do so by means of the `splines_cont()` function:

```
> splines_cont(trial_data, arm = 3, bs_degree = 3, alpha = 0.025)
$p_val
[1] 0.01686
$treat_effect
[1] 0.2795447
$lower_ci
[1] 0.02160482
$upper_ci
[1] 0.5374847
$reject_h0
[1] TRUE
$knots
[1] 100 250 400
```

In the output of each analysis function, the first element of the list is the p-value (`p_val`) corresponding to testing the null hypothesis  $H_0 : \theta_3 = 0$ , followed by the estimated treatment effect in terms of difference of the mean response of the evaluated

## 4.2. Examples

treatment and the control mean (`treat_effect`) and the respective lower and upper confidence limits (`lower_ci`, `upper_ci`). The list also includes a binary indicator of (`p_val < alpha`), i.e., whether the null hypothesis can be rejected on the specified significance level (`reject_h0`). In the considered case, the null hypothesis is rejected by all models, which implies that treatment arm 3 is efficacious. The output of the spline regression also includes the positions of the inner knots, which are in this case placed to the beginning of each period. Furthermore, each output includes the respective fitted regression model (`model`), which is here omitted for simplicity. However, the fitted model can be further analysed using the conventional R functions for generalized linear models, such as `summary(fixmodel_cont(data = trial_data, arm = 3)$model)`.

### 4.2.2. How to Run a Simulation Study

As the package was used to perform all simulations presented in Chapter 3, we show how to use it to run simulation studies by means of a simple example. Consider a platform trial with four experimental treatment arms entering sequentially. We aim to assess the robustness of analysis methods that utilise non-concurrent controls in the presence of time trends. For this, we first create a data frame with the desired scenarios that contains all the parameters needed for data generation and analysis.

```
> lambda_values <- rep(seq(-0.15, 0.15, length.out = 9), 2)
> sim_scenarios <- data.frame(num_arms = 4,
  n_arm = 250,
  d1 = 250*0,
  d2 = 250*1,
  d3 = 250*2,
  d4 = 250*3,
  period_blocks = 2,
  mu0 = 0,
  sigma = 1,
  theta1 = 0,
  theta2 = 0,
  theta3 = 0,
  theta4 = 0,
  lambda0 = lambda_values,
  lambda1 = lambda_values,
  lambda2 = lambda_values,
  lambda3 = lambda_values,
  lambda4 = lambda_values,
  trend = c(rep("linear", 9), rep("stepwise_2", 9)),
  alpha = 0.025)
```

We assume here that the null hypothesis holds for all experimental arms (`theta1=...=theta4=0`). We vary the strength (`lambda`) and pattern (`trend`) of the time trend, in order to invest-

#### 4. Software

igate their impact on the type I error rate, bias and mean squared error (MSE) of the treatment effect estimates. Note, however, that the time trend is equal across arms.

We use the function `sim_study_par()` to perform a simulation study with the created scenarios. Here, we evaluate the 4th experimental treatment arm using regression models with period adjustment as fixed or uncorrelated random effect, as well as cubic spline regression and compare their operating characteristics to the separate approach, where only CC data is used for the analysis and the pooled analysis, which naively pools CC and NCC data without further adjustments. Each scenario will be replicated 1000 times.

```
> sim_results <- sim_study_par(nsim = 1000,
                                scenarios = sim_scenarios,
                                arms = 4,
                                models = c("fixmodel",
                                           "mixmodel",
                                           "splines",
                                           "sepmodel",
                                           "poolmodel"),
                                endpoint = "cont",
                                verbose = TRUE)
```

By default (if `verbose=TRUE`), the function reports the system time after each scenario finishes in order to track the progress of the simulations. These messages can be suppressed by setting `verbose=FALSE`.

The resulting data frame contains the considered scenarios and simulation results. The results include the probability of rejecting the null hypothesis, bias and MSE of the treatment effect estimates. We can now visualize the performance of the considered analysis methods with respect to the strength and pattern of the time trend, similarly to Section 3.2. Figure 4.4 depicts the type 1 error, bias and MSE with respect to the strength of the time trend and according to the pattern of the time trend. The results show that the pooled analysis and the mixed model lead to inflation of the type I error rate in the presence of positive time trend and its deflation if there are negative time trends. The separate approach and the regression model with period as fixed effect control the type I error rate and yield unbiased treatment effect estimates. As we are using relatively weak time trends in this example, the cubic spline regression control the type I error in case of both, linear and stepwise time trend. However, we can observe slight bias in the treatment effect estimates in the case with stepwise time trend.

## 4.2. Examples

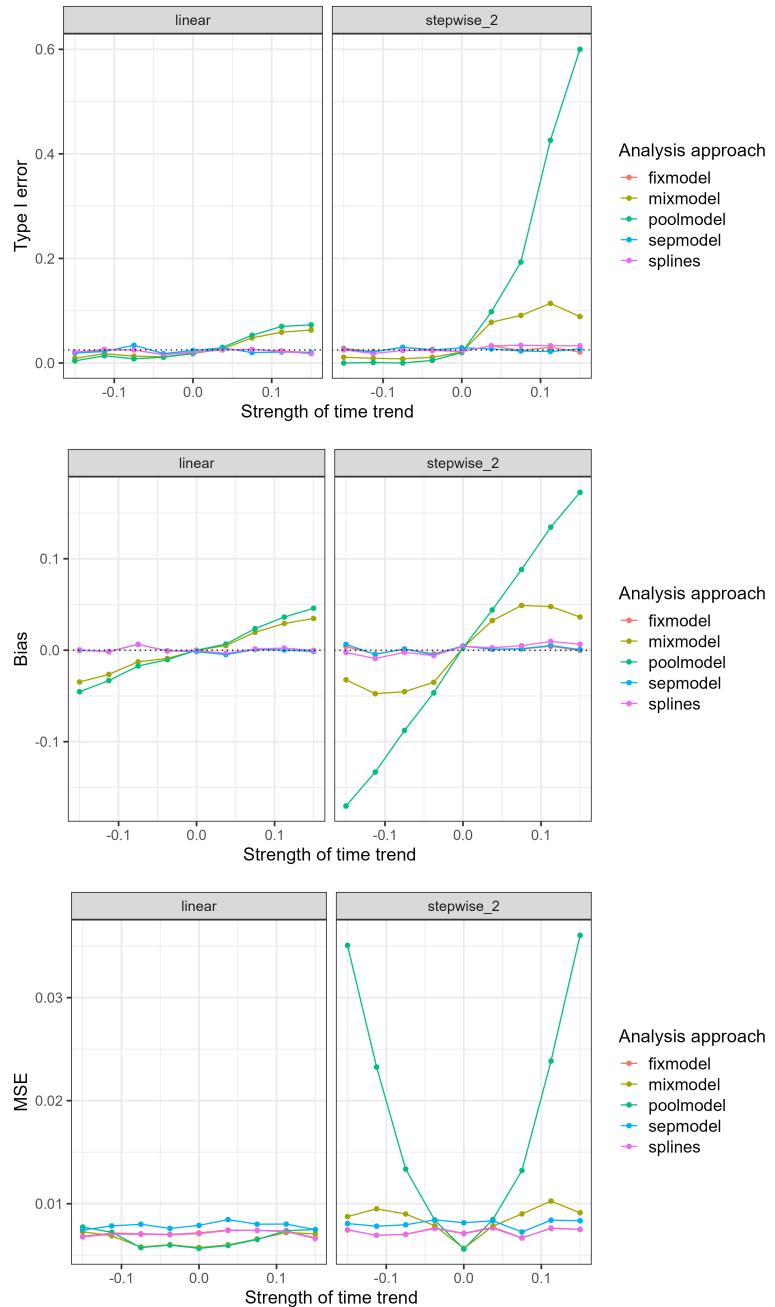


Figure 4.4.: Results of the toy simulation study. Type I error rate, bias and MSE of the treatment effect estimates for treatment arm 4 with respect to the strength of the time trend.