

**AN ADAPTIVE TWO-STAGE DOSE-RESPONSE  
DESIGN METHOD FOR ESTABLISHING PROOF  
OF CONCEPT IN DRUG DEVELOPMENT**

by

**Yoko Tanaka**

B.Sc. in Pharmaceutical Sciences, Kyoto University, Japan, 1997

Submitted to the Graduate Faculty of  
the Graduate School of Public Health in partial fulfillment  
of the requirements for the degree of

**Doctor of Philosophy**

University of Pittsburgh

2010

UNIVERSITY OF PITTSBURGH  
GRADUATE SCHOOL OF PUBLIC HEALTH

This dissertation was presented

by

Yoko Tanaka

It was defended on

September 17, 2010

and approved by

Stewart Anderson, Ph.D., Professor, Department of Biostatistics,

Graduate School of Public Health, University of Pittsburgh

Allan R. Sampson, Ph.D., Professor, Department of Statistics, School of Arts and Sciences,

Department of Biostatistics, Graduate School of Public Health, University of Pittsburgh

Abdus Wahed, Ph.D., Associate Professor, Department of Biostatistics,

Graduate School of Public Health, University of Pittsburgh

Marnie Bertolet, Ph.D., Assistant Professor, Department of Epidemiology,

Graduate School of Public Health, University of Pittsburgh

Dissertation Director: Stewart Anderson, Ph.D., Professor, Department of Biostatistics,

Graduate School of Public Health, University of Pittsburgh

Copyright © by Yoko Tanaka  
2010

# AN ADAPTIVE TWO-STAGE DOSE-RESPONSE DESIGN METHOD FOR ESTABLISHING PROOF OF CONCEPT IN DRUG DEVELOPMENT

Yoko Tanaka, PhD

University of Pittsburgh, 2010

In clinical drug development, searching for the true dose-response curve is ethically and logistically challenging. Establishing evidence of dose-response or Proof of Concept (PoC) is the first step for both determining the best dose-response model and optimizing a treatment dose correctly for clinical use. To overcome these challenges, we employ an adaptive two-stage design where both adding and dropping treatment arms is possible between stages. In the first part of this dissertation, we develop a method extending the Multiple Comparison Procedures and Modeling (MCP-Mod) approach into this adaptive two-stage design. Our goal is to establish "global" PoC across the stages. Between stages, we propose using an Adding and/or Dropping Treatment Adaptation Rule (ADTAR). In the ADTAR method, dose specifications in the second stage depend on the first stage's results. Treating the unobserved doses and imbalanced aggregate sample sizes in the second stage as missing data, we derive weights and adjust the test statistics in the second stage. Specifically, we assume that the missing data mechanism caused by ADTAR is missing at random. At the end of the second stage, we perform the global PoC test combining the test results from both stages. To preserve the family-wise error rate, we use a Conditional Error Function. Using simulation studies, we evaluated our design method and compared it to a conventional (one-stage) study design and different fixed two-stage designs. Our method showed overall robust high power for detecting the global PoC across three forms of true dose-response curves. In the second part of this dissertation, we find constraints for choosing doses in the original and extended MCP-Mod methods. Specifically, we establish lower bounds of the number and

levels of doses for each method using simulation studies. Our proposed method is a viable tool in searching for a dose-response relationship. In accordance with ICH guidelines, our method helps to provide optimal doses of drugs for treating or preventing different diseases. Since drugs are widely used in human populations, such methods have a great Public Health impact in appropriately treating or preventing many types of diseases.

**Keywords:** Adaptive Two-Stage Design, Adding and/or Dropping Treatment Adaptation Rule (ADTAR), Dose-Response Models, Proof of Concept (PoC), Multiple Comparison Procedures - Modeling Approach (MCP-Mod), Clinical Drug Development.

## TABLE OF CONTENTS

<b>PREFACE</b> . . . . .	x
<b>1.0 INTRODUCTION</b> . . . . .	1
<b>2.0 KEY LITERATURE REVIEW</b> . . . . .	4
2.1 DOSE-RESPONSE MODELS . . . . .	4
2.2 CLASSICAL DOSE-RESPONSE STUDY DESIGNS . . . . .	5
2.3 ADAPTIVE DOSE-RESPONSE STUDY DESIGNS . . . . .	8
2.3.1 Adaptive Designs . . . . .	8
2.3.2 Adaptive Dose-Response Designs . . . . .	11
2.4 MULTIPLE COMPARISON PROCEDURES-MODELING METHOD . . . . .	13
2.5 MODIFICATION OF ADDING AND/OR DROPPING TREATMENT ARMS IN ADAPTIVE DESIGNS . . . . .	15
2.6 CONDITIONAL ERROR FUNCTION . . . . .	16
<b>3.0 EXTENDED MULTIPLE COMPARISON PROCEDURES AND MODELING APPROACH</b> . . . . .	18
3.1 METHODOLOGY . . . . .	18
3.1.1 Overview . . . . .	18
3.1.2 Adding and/or Dropping Treatment Adaptation Rule (ADTAR) . . . . .	19
3.1.3 Modeling Stage 2 through Missing Data Analysis . . . . .	21
3.1.4 Combining Stage 1 and Stage 2 . . . . .	22
3.2 DERIVATION OF WEIGHTS . . . . .	22
3.3 CONTROLLING THE FWER CLOSE TO A PRE-CHOSEN $\alpha$ -LEVEL . . . . .	25
3.4 SIMULATION STUDY: EVALUATING EX-MCP-MOD . . . . .	27

3.4.1	Simulation Setup . . . . .	28
3.4.2	Simulation Results . . . . .	33
3.4.3	Effect of Sample Size Imbalance across Groups . . . . .	33
3.4.4	Controlling the FWER . . . . .	36
3.5	DISCUSSION . . . . .	39
<b>4.0</b>	<b>CONSTRAINTS OF DOSE CHOICES IN THE MULTIPLE COM-</b>	
	<b>PARISON PROCEDURES AND MODELING APPROACH . . . . .</b>	<b>40</b>
4.1	CONSTRAINTS TO DOSE CHOICES IN THE MCP-MOD . . . . .	40
4.2	SIMULATION STUDY . . . . .	43
4.2.1	Simulation Method . . . . .	43
4.2.2	Optimal Number of Doses . . . . .	45
4.2.3	Optimal Levels of Doses . . . . .	46
4.2.4	The Best Configurations of Doses . . . . .	49
4.2.5	The Weighting Effect on Positive Definiteness . . . . .	51
4.3	DISCUSSION . . . . .	54
<b>5.0</b>	<b>CONCLUSION . . . . .</b>	<b>55</b>
	<b>APPENDIX. R PROGRAMS FOR SIMULATION STUDIES . . . . .</b>	<b>57</b>
A.1	Proof of Activity (PoA) of <i>ADTAR I</i> (Chapter 3) . . . . .	57
A.2	Proof of Activity (PoA) of <i>One-Stage Design</i> (Chapter 3) . . . . .	74
A.3	Proof of Activity (PoA) of <i>Fixed Two-Stage Designs</i> (Chapter 3) . . . . .	80
A.4	Dose Choices for Positive Definite Matrix (Chapter 4) . . . . .	91
A.5	Weighted Correlation Matrices using Different Numbers of Doses (Chapter 4)	102
	<b>BIBLIOGRAPHY . . . . .</b>	<b>112</b>

## LIST OF TABLES

1	Original and Standardized Model Formula . . . . .	30
2	ADTAR I . . . . .	32
3	Weights ( $\omega_{ij}$ ) Under The Adaptation Choices in ADTAR I . . . . .	32
4	PoAs of <i>ADTARI</i> , <i>Non-weighting ADTARI</i> , <i>One-Stage MCP-Mod</i> , and four two-stage fixed designs under varying D-R shape $\times$ sample size (1) . . . . .	34
5	PoAs of <i>ADTARI</i> , <i>Non-weighting ADTARI</i> , <i>One-Stage MCP-Mod</i> , and four two-stage fixed designs under varying D-R shape $\times$ sample size (2) . . . . .	34
6	PoAs of <i>Random Two-Stage</i> design under varying D-R shape $\times$ sample size . . . . .	38
7	The Minimum and Maximum of The Smallest Eivgenvalues of The Correlation Matrices For Different Number of Doses Simulated From Equal Dose Intervals in $[0,1]$ . . . . .	46
8	The Minimum of The Smallest Eivgenvalues of The Correlation Matrices Against Various Dose Intervals For Different Number of Doses . . . . .	49

## LIST OF FIGURES

1	Evaluation: Adaptive Design with ADTAR, Fixed Two-stage Designs, and One-stage MCP-Mod Design . . . . .	29
2	Standardized Emax ( $f_1$ ), quadratic ( $f_2$ ), and logistic ( $f_3$ ) models. . . . .	31
3	Proof of Activity in Global Stage for Different Dose-Response Patterns in <i>ADTARI</i> , <i>Non-weighting ADTARI</i> , <i>One-Stage MCP-Mod</i> , <i>Maximum</i> , and <i>Minimum</i> designs . . . . .	35
4	Equal probability of Stage 1's results and sample sizes with and without weighting. . . . .	37
5	<i>Random Two-Stage</i> design . . . . .	38
6	Three-Dimensional Shape of Correlation Elements. . . . .	44
7	The Minimum and Maximum Log 10 of the Smallest Eigenvalues of $\mathbf{R}_3 \times 3$ for a Placebo and Three Treatment Doses ( $d_1$ , $d_2$ , and $d_3$ ). . . . .	47
8	The Minimum and Maximum Log 10 of the Smallest Eigenvalues of $\mathbf{R}_3 \times 3$ for a Placebo and Four Treatment Doses ( $d_1$ , $d_2$ , $d_3$ , and $d_4$ ). . . . .	48
9	Log 10 of The Smallest Eigenvalues of $\mathbf{R}_3 \times 3$ for Varying Dose Intervals in $[0, 1.0]$ . . . . .	50
10	Log 10 of the Smallest Eigenvalues of $\mathbf{R}_3 \times 3$ for a Placebo and Three Treatment Doses ( $d_1$ , $d_2$ , and $d_3$ ). . . . .	52
11	Log 10 of the Smallest Eigenvalues of $\mathbf{R}_3 \times 3$ for a Placebo and Three Treatment Doses ( $d_1$ , $d_2$ , and $d_3$ ). . . . .	53

## **PREFACE**

I am truly grateful to my advisor, Dr. Anderson, for his continuous, patient, and encouraging guidance in completing this thesis. I also thank my committee members, Dr. Sampson, Dr. Wahed, and Dr. Bertolet, and my former committee member, Dr. Vuga, for their help and guidance. Dr. Sampson especially helped to direct my research with his deep knowledge and experiences in this research area. Lastly, I greatly thank all of my friends (especially Shuhei Okumura for many statistical discussions), parents, brother's family, and my husband, Franz, for their help and encouragement through my graduate years.

## 1.0 INTRODUCTION

In recent years, novel designs of clinical trials have played an increasingly important role in the drug approval process. For example, the use of *adaptive* designs generally allow for earlier decisions whether to continue or stop a trial in a combined phase such as Phases I and II or Phases II and III. The FDA recently released a draft guidance on adaptive designs and commented that employing adaptive designs in clinical trials is expected to lead to operational efficiency as well as more informative results given a well-controlled study setting [13]. In the long run, it is expected to benefit patients who need efficacious and safety drugs as early as possible.

The term, ‘adaptive design’ has a broad meaning. Gallo, et al. [14] referred to an adaptive design as being a clinical study design that uses accumulating data to decide how to modify aspects of the study as it continues without undermining the validity and integrity of the trial. Other authors such as Chang, et al. [8] cite many types of adaptations such as adjusting sample size, stopping early due to efficacy or futility, changing the timing and the number of analyses, etc. A more thorough discussion of such adaptations is given in section 2.3.1.

In the FDA’s draft guidance on adaptive designs, clinical experimenters are encouraged to detail *pre*-specified adaptation rules in protocols [13]. The law of Good Clinical Practice (GCP) [22] requires that a clinical protocol be approved by Institutional Review Boards (IRBs) and the regulatory agencies prior to performing the clinical trial. If any protocol amendments are required later on, then that amendment should not be implemented before being reviewed and approved by IRBs and regulatory agencies. Thus, such review procedures often delay the clinical trial. When design adaptations are pre-specified in the initial study protocol, protocol reviewers at the IRBs and regulatory agencies are informed in advance

of what design aspects will be changed in the course of trial. Hence, in such cases, the subsequent review procedures could be expedited by the well-informed protocol reviewers.

In clinical drug development, one of the important factors to consider is the nature of a dose-response (D-R) relationship. A candidate drug is usually brought into the clinical stage with its preliminary pharmacological, toxicological, and absorption/distribution/metabolism/excretion (ADME) drug profile, which has been characterized by data from the pre-clinical stage. However, the physiological mechanisms of the human body are not identical with those of the animals studied in the pre-clinical stages and it is not uncommon that the investigational drug behaves differently than the predicted profile information. Therefore, it is important to characterize the drug profile in the clinical stage so that a dosage is correctly optimized for clinical use. Unlike the pre-clinical stage, it is not easy to characterize the entire ADME profile in the clinical stage due to ethical constraints. For example, in a clinical setting it could be unethical and/or burdensome to observe the relationship between drug concentrations in plasma and responses because of the necessity of obtaining blood samples. In contrast, the relationship between dose administration and responses is much easier to observe in the same setting. Consequently, a D-R relationship is recognized as important information in order to determine a maximum toxicity dose, a minimum efficacious dose, and a therapeutic dose window of a drug in its clinical stage.

The International Conference on Harmonization Guideline E4 (ICH-E4) [21] raised an issue that practical study designs do not exist to allow for precise determination of these doses. The randomized parallel dose-response study design is extensively used and has had considerable success among the classical clinical study designs. However, in ICH-E4, regulatory agencies and drug developers have been encouraged to employ new approaches in the design and analysis of dose-response data. Accordingly, adaptive dose-response designs have been one such novel approach of recent use in the pharmaceutical industry.

The Pharmaceutical Innovation Steering Committee (PISC) of the Pharmaceutical Research and Manufacturers of America (PhRMA) formed the Dose-Ranging Designs working group in 2005, and the group has summarized their work to evaluate and develop alternative approaches to conventional dose-finding designs including adaptive dose-ranging designs by Bornkamp, et al. [4]. The multiple comparison-modeling approach (MCP-Mod) by Bretz, et

al. [6] and Pinheiro, et al. [29] is one of the evaluated approaches. This hybrid methodology uses both multiple comparison procedures (MCP) and modeling techniques (Mod) within one strategy for dose-ranging studies. Using the MCP-Mod method, one is able to test if a dose-response curve is significant. Hence, if this proof of evidence, or *proof of concept* (PoC) result, is to be significant, the best dose-response model is selected from a set of multiple candidate models, and is followed by estimation of target doses within the selected best model.

In this dissertation, we propose to generalize one previously developed method for two-stage adaptive designs to accommodate situations where both adding and dropping treatment arms are possible between stages. Our procedure will be based on using the data set in its entirety and will control for the familywise error rates. Specifically, we propose the following:

1. to develop an efficient two-stage study design where an adaptation rule adding and/or dropping treatment arms, which we call ADTAR, is pre-specified;
2. to use our adaptive design to establish evidence of a global dose-response relationship or Proof of Concept (PoC) for a treatment; and
3. to show that the two-stage design with the pre-specified adaptation rule can be used to improve statistical power in establishing global PoC.

Our proposed approach extends the MCP-Mod method. We employ the idea of MCP-Mod to use sets of multiple candidate dose-response models in a two-stage adaptive design, but we further allow adding new treatment groups and/or dropping inferior treatment groups in the second stage. At the beginning of the second stage, we prepare the same set of the candidate dose-response models while using possibly different dose levels. In our study design, not only is PoC evaluated at the end of the first stage, but we also consider how to combine data across the two stages to globally establish the PoC while controlling the family-wise error rate (FWER). Since the ICH-E4 advises drug developers to examine the entire database for possible dose-response information, the entire dose-response data is used in our adaptive two-stage study design. We consider detecting a D-R relationship in the special case of adding and/or dropping treatment groups, and evaluate the power of our approach compared to that of a common conventional study.

## 2.0 KEY LITERATURE REVIEW

### 2.1 DOSE-RESPONSE MODELS

Let  $f(d, \boldsymbol{\theta})$  denote a general steady-state D-R model with a parameter vector,  $\boldsymbol{\theta}$ , which is used to describe a mean response at a certain dose level,  $d$ . The  $E_{\max}$  model is a simple empirical dose-response model which is applicable to situations where the test drug does not behave irreversibly and where equilibrium exists between the concentration of the drug at the effect (e.g., organ) site and in the plasma [9]. Its general form is

$$f(d, \boldsymbol{\theta}) = E_0 + \frac{E_{\max}d}{EC_{50} + d}, \quad (2.1)$$

where  $\boldsymbol{\theta} = (E_0, E_{\max}, EC_{50})$ ,  $E_0$  is a baseline effect,  $E_{\max}$  is the maximum effect associated with  $d$ , and  $EC_{50}$  is the dose that produces half of the maximum change [5].

When a pharmacokinetic (PK) state is considered to be steady under the same conditions for the  $E_{\max}$  model, logistic models and log-linear models are also commonly used to study a relationship between a drug concentration and a drug effect [27]. Branson, Pinheiro, and Bretz [5] used the following logistic model in their development of the MCP-Mod approach:

$$f(d, \boldsymbol{\theta}) = E_0 + \frac{E_{\max}}{1 + \exp\left[\frac{EC_{50}-d}{\delta}\right]}, \quad (2.2)$$

where  $E_0$  is the baseline effect,  $E_{\max}$  is the maximum effect size from the baseline effect,  $EC_{50}$  is the dose producing half of  $E_{\max}$ , and  $\delta$  is the parameter related to the rate of response change with dose. In 2.2,  $E_0$  and  $E_{\max}$  are location and scale parameters, respectively. Therefore, its standardized model has the standardized parameter vector,  $\boldsymbol{\theta}^* = (EC_{50}, \delta)$ .

According to Branson, et al. [5], in this standardized model the effect size increases from half of  $E_{\max}$  to

$$(1 + \exp(-1))^{-1} E_{\max} = \left( 1 + (1 + (-1) + \frac{(-1)^2}{2!} + \frac{(-1)^3}{3!} + \dots) \right)^{-1} E_{\max} \approx 0.75 E_{\max}$$

as one increases dose from  $EC_{50}$  to  $EC_{50} + \delta$ . Another possible model choice is the log-linear model (Branson, et al. 2003),

$$f(d, \boldsymbol{\theta}) = E_0 + \delta \log(d + c) , \tag{2.3}$$

where  $\boldsymbol{\theta} = (E_0, \delta)$ , and  $c$  is a constant which satisfies the inequality,  $c > -d$ .

## 2.2 CLASSICAL DOSE-RESPONSE STUDY DESIGNS

Ruberg [30] discussed classical designs for dose-response studies and related issues. In drug development, dose-response clinical trials are generally performed to study four aspects of an investigational drug: 1) to discover evidence of any treatment effect, 2) to find differences between treatment effects and a placebo effect, 3) to characterize the relationship between dose and response, and 4) to choose the optimal dose. To achieve these objectives, two analytic approaches, ANOVA and regression modeling, are commonly used in classical clinical study designs.

ANOVA is more suitable for objectives 1) and 2), and contrast tests or multiple comparison tests are embedded when we have multiple doses and a placebo group. For example, when an alternative hypothesis to test the hypothesis associated with objective 1) is constructed specifically as the ordered mean responses with at least one inequality, the contrast test is the most powerful. To test differences between treatment effects and a placebo group in 2), the multiple comparison test is performed in several possible procedures. The most basic procedure is to compute a test statistic assuming that the test statistic follows  $k$ -variate analog to student's  $t$ -distribution for the  $k$  treatment groups under the null hypothesis. Both

step-down and step-up procedures are available to modify the procedure if needed. The step-down procedures construct multiple null hypotheses so that a higher dose or a higher response are tested first. The next higher dose or response are tested only when the former null hypothesis is rejected. This step is repeated in a descending order until a certain null hypothesis is accepted. The step-up procedures take reverse steps starting with a lower dose or response against the step-down procedures. The step-up procedures proceed the next hypothesis testing when a null hypothesis is accepted and stop when a null hypothesis is rejected. In both step-down and step-up procedures, the overall type I error for the multiple comparisons is controlled and hence, estimating an minimum efficacious dose (MED) to satisfy objective 4) is possible. The step-up procedures are more powerful than the step-down procedures when more doses are effective. However, in terms of increasing power, the single-step procedures appear to be better options as they utilize all dose information about possible response patterns in contrast to tests with  $k$ -multivariate t-statistics.

A regression modeling approach is useful for objective 3) characterizing the relationship between dose and response; and objective 4) choosing the optimal dose. The popular  $E_{max}$  models were previously described. Since we assume that the underlying dose is continuous in the model, it is possible to estimate the confidence intervals for the optimal dose. Parenthetically, it is not necessary to assume equal variances of the population response at each dose. Finally, by evaluating model fit, a best response curve function can be selected. However, it is not always easy to specify the functional form of a model. Furthermore, it may be difficult to differentiate a target dose from other doses if the confidence intervals for the optimal dose are relatively wide.

The International Conference on Harmonization Guideline E4 (ICH-E4) [21] reviewed four classical study designs for dose-response clinical trials. *Placebo controlled, randomized parallel dose-response clinical trials* are the most common classical designs where subjects are randomized into several treatment dose groups or a placebo group. Since these types of studies are designed to reduce confounding factors as much as possible, the analytic ap-

proaches are usually simple and treatment differences are easily evaluated. With a placebo group, a trend in response can be analyzed over a dose range. The ICH-E4 [21] commented that one disadvantage is that only a population-averaged dose-response curve can be studied because individuals do not have their own placebo control data. They also recommend the use of large sample sizes to increase the precision in the estimate of a treatment effect. When a combination therapy with more than one drug need to be analyzed, a special case of placebo-controlled designs called *Factorial designs* is useful.

*Forced titration studies and optional titration* (placebo-controlled titration to end-point) studies are designed so that each subject receives several different doses sequentially until they reach a certain fixed dose or a target response. Regardless of how the dose-response is established analytically, it is difficult to determine whether the desired dose-response occurred due to a delayed (carryover) effect or due to an immediate effect because of the optimal accumulation of the plasma concentration of the drug.

*Cross-over dose-response study designs* overcome the disadvantage of the placebo-controlled randomized parallel designs and usually use smaller sample sizes. In typical Latin squares designs, subjects are randomized to several doses. After a short period of time to wash-out carryover effects, they are assigned to different doses. This procedure is repeated until all subjects experience all different doses. Both population-averaged and individual dose-response curves can be evaluated but uncertainty may remain in terms of carryover effects, baseline comparability after the first period, and period-by-treatment interactions. Also the analyses may be not simple when some subjects withdraw in the middle of a study.

## 2.3 ADAPTIVE DOSE-RESPONSE STUDY DESIGNS

### 2.3.1 Adaptive Designs

Gallo, et al. [14] defined adaptive design as a clinical study design that uses accumulating data to decide how to modify aspects of the study as it continues without undermining the validity and integrity of the trial. Golub [16] commented that advantages of adaptive clinical trials are that 1) one can adopt modifications of a study design if the interim data is reliable, and 2) one can avoid possibly underpowered or uninformative conclusions. Hence, by design, clinical trials can be more effectively operated and correct estimates for arriving at the conclusions of a study can be obtained with minimal operational bias. Consequently, adaptive designs may be superior to classical designs in terms of achieving scientific conclusions rapidly while still maintaining study integrity [10]. on optimizing the operating characteristics associated with such designs. For example, Koyama, et al. [24] developed a technique to optimize two-stage adaptive designs and concluded that the three most influential design components in Stage 1 are the sample size, and the type I and II error rates.

There are many ways that an investigator can choose to adapt a study design. Chang, et al. [8] listed ten different adaptations as 1) adjusting sample size, 2) stopping early due to efficacy or futility, 3) changing the timing and the number of analyses, 4) dropping inferior treatment groups, 5) adding new treatment groups, 6) response-adaptive randomization, 7) modifying the target population, 8) changing study endpoint, 9) treatment switching, and 10) any combination of the other adaptations. Dragalin [11] and Bornkamp [4] classified adaptive designs into five major trial categories in terms of treatment arms and trial characteristics: single-arm trials, two-arm trials, multiple-arm trials, seamless design trials, and dose-finding trials. These categories are associated with drug development phases where particular classified designs are typically used.

In **single-arm trials**, only one treatment group is studied. These trials are often performed as Phase I clinical trials in drug development. Phase I is the first stage where a new

investigational drug is studied in humans for its safety, metabolism, and pharmacology evaluations. There are two types of adaptive single-arm designs: *two-stage designs* and *screening designs*. Two-stage adaptive designs are the studies where a single treatment group is studied over two stages to test if a drug shows a certain probability of response. Between the stages, an interim analysis is performed and the design is modified according to a pre-specified rule. An early stopping rule for futility and adaptive sampling are the common modifications, and the designs are also seen in Phase II trials. Screening designs are the designs which are modified over the entire drug screening process. The screening designs can be used to identify the most potentially efficacious drug(s) within a shortest time by design modifications.

In **two-arm trials**, one group is usually called the treatment (experimental) group and the other is a placebo (control) group. The two-arm trials are usually used as Phase II or Phase III clinical trials. In Phase II, preliminary data on drug efficacy are obtained in a target patient population. Study confounders are well-controlled with relatively small-scaled sample sizes. Phase III trials are called confirmatory studies, which are expanded controlled or uncontrolled, and additional information about drug efficacy and safety are evaluated with large-scaled sample sizes. Typical two-arm trials are *group sequential* designs and *adaptive group sequential* designs. Group sequential designs consist of two or more cohort stages and a stopping rule is applied for the preliminary result of each interim analysis between stages. With pre-specified stopping boundaries, there is an opportunity to make an early decision about going to the next stage or not at each step. Such boundaries are usually associated with hypothesis testings on clinically beneficial response and the overall type I error rate can be controlled. The group sequential designs are seen in Phase I stage as well. Adaptive group sequential designs are extensions of the group sequential designs which use a stopping rule for individual subjects instead of cohort group units.

In **multiple-arm trials**, more than two treatment groups are studied often in Phase II settings. Since the trial objectives are often finding a clinically efficacious trend over doses and selecting the optimal dose for a clinically beneficial effect, their roles can be a part of

dose-finding trials. Three types of adaptive designs, *Bayesian designs*, *pairwise comparisons with group sequential designs*, and *flexible designs*, are used in multiple-arm trials. Bayesian designs are used to make a decision about stopping or continuing the studies at the interim analyses based on either the posterior distribution of the parameters of interest or the estimated losses of discontinuation. Pairwise comparisons with group sequential designs are the group sequential designs where multiple treatment groups are compared with a placebo group by controlling the overall type I error. It is possible to adaptively drop some treatment arms due to inferior outcomes. Flexible designs are two-stage designs where multiple comparisons are performed in each stage and the results of the two stages are combined to test an overall hypothesis by controlling the familywise type I error rate. The advantage of the designs is to allow many types of adaptations such as sample size adjustment, early stopping, changing endpoint, and adding or dropping treatment arms.

**Seamless design trials** are used across two phases of drug development, that is, across Phase I and II, or, alternatively, across late Phase II and Phase III, to combine different trial aims within one protocol. Since it is possible to use more consistent operational systems with one protocol comparing with two separate protocols, operational biases are expected to be reduced. An example of adaptive designs within this category is the *learning/confirmatory design in Phase II/III*. Learning/confirmatory designs in Phase II/III combine a Phase II trial to explore a drug efficacy with multiple doses and possibly with secondary outcome, which can be observed quickly, and a Phase III trial to confirm the Phase II result with a primary outcome. When they are performed in one protocol, the overall type I error is to be controlled.

In **the dose-finding trials**, the main goal is to find a best dose to be used in the following confirmatory trials according to a pre-specified criterion. The selected dose is manufactured in a much larger scale and the clinical data are to be collected for the selected dose in the following studies. Thus, dose-finding is an important step in drug development. The next subsection details the adaptive designs classified into this category.

### 2.3.2 Adaptive Dose-Response Designs

Adaptive dose-response studies can employ broad modifications including adjusting sample size, stopping early due to efficacy or futility, changing the timing and the number of analyses, dropping inferior treatment groups, adding new treatment groups, and response-adaptive randomization. According to PISC of PhRMA [4], the general objectives of adaptive D-R studies are: (1) to show proof of concept; (2) to establish the D-R curve; (3) to select the optimal dose; and (4) to estimate the therapeutic window. For these objectives, various approaches for a D-R study have been developed. Typical examples include 1) a Bayesian adaptive dose allocation approach [3], 2) a D-optimal response-adaptive approach [12], 3) a multiple comparison procedures-modeling approach (MCP-Mod) [5, 6], 4) a Bayesian model-averaging approach [4], 5) a multiple trend test approach [4], and 6) a non-parameteric D-R modeling approach [4]. A brief description of each type of design is given below.

**Bayesian adaptive dose allocation approach.** In this approach, each new enrolled subject is assigned to a treatment group or a placebo group according to an allocation rule. When the next subject is enrolled, all available data at that time including ongoing subject data is used to select an optimal dose for the subject. The data is also used to update the estimated dose-response curve. Whether to stop the study or not due to futility or remarkable efficacy is determined based on the posterior probability of the response at the optimal dose. Timely data capturing is a challenging issue and the frequent access to the available data by sponsors or physicians can cause significant biases. Relatively large biases in target dose estimation have been shown to be present as demonstrated by the simulation study by PISC of PhRMA [4] when the number of treatment groups was large.

**D-optimal response-adaptive approach.** In a group sequential adaptive design, the D-optimal allocation rule is sometimes used. The D-optimal algorithm determines an allocation ratio by minimizing the variance of dose-response model parameters at the completion of each study cohort. At the end of the study, the final dose-response curve estimation and the optimal dose selection are performed. The study information is maximized by D-optimal

allocation rule. However, because of the characteristics of an allocation rule, the method may not be best suited for the case where the modification of adding and dropping treatment arms is forced in a study.

**Multiple Comparison Procedures-Modeling approach (MCP-Mod).** In this method, multiple comparison procedures (MCP) and modeling techniques (Mod) are combined within a single study. For the test on the existence of a dose-response curve, a multiple model candidate set is pre-specified. These models are standardized and used to compute test statistics assuming that the test statistics follow the multivariate  $t$ -distribution under a null hypothesis and a familywise error rate can be controlled. For the best model selection, a certain model selection criterion such as BIC is used. After the best dose-response curve is selected, a target dose is estimated using the selected best model.

**Bayesian model-averaging approach.** A set of simpler dose-response models are pre-specified and the individual model weight is calculated based on prior model probabilities. The posterior distributions on model parameters are updated from the prior distributions of the parameters using Markov Chain-Monte Carlo for each model. The models are averaged by the weights. This method is useful when the main objective of a dose-response study is estimating some quantities which are not dependent on the dose-response curve.

**Multiple trend test approach.** As the class of Emax models can cover various dose-response curves, the underlying dose-response curve is described by three different Emax models called lower, middle, and upper Emax models. The middle curve is chosen so that the differences in power of the three trend tests can be minimized. The maximized likelihood is used to estimate the dose-response curve, and the inverse regression method on the estimated curve is used to estimate the target dose for a pre-specified clinically significant effect. Since the target dose is not the attribute of the model, it may not be able to identify for the pre-specified and selected dose-response curve. The simulation result by PISC of PhRMA [4] implied that the design could misguide us to a further clinical trial when there is no significant evidence of a treatment response.

**Non-parameteric dose-response modeling approach.** This is a non-parametric regression modeling method using local polynomials (LOESS). As for the estimation of the dose-response effect, a multiple contrast test is used. A target dose is estimated using local quadratic regression or smoothing techniques. In the study by PISC of PhRMA [4], the estimation of target doses was not be so powerful as MCP-Mod. Also, the interpretation of the dose-response curve may not be straightforward.

Among these approaches, the first (Bayesian adaptive dose allocation approach) and second approaches (D-optimal response-adaptive approach) use adaptive designs where one is allowed to modify design aspects while a trial is performed in either a continuous or in a group-sequential manner. In contrast, the remaining four approaches have flexibility in the procedures of analysis but do not use an adaptive design.

## 2.4 MULTIPLE COMPARISON PROCEDURES-MODELING METHOD

The MCP-Mod approach by Branson, et al. [5] and Bretz, et al. [6] allows one to make three different decisions within a dose-finding study, (i) a proof of concept (PoC) that the treatment is effective, (ii) the best D-R relationship, and (iii) a choice of dose(s) to use in further development. In the establishment of PoC, a set of unnested candidate models which cover a suitable range of dose-response shapes is used instead of a pre-specified single dose-response model. Each candidate model is assessed by an optimally chosen contrast test and multiple comparison procedures (MCPs) are employed to combine model results in order to control the family-wise error rate (FWER). If PoC has been established, the best model is selected (Mod) from the statistically significant models in the candidate set. Furthermore, the selected D-R model is used to estimate a target dose for clinical use. We will now give a more detailed description of the individual steps.

The MCP step is necessary for the establishment of PoC. Branson, et al. and Bretz et al. assume that responses are observed for a given set of parallel groups of patients randomized to placebo,  $d_0$ , and  $k$  treatment doses,  $d_1, d_2, \dots, d_k$ . Suppose that one considers the following D-R model specification:

$$y_{ij} = f(d_i, \boldsymbol{\theta}) + \epsilon_{ij}, \quad i = 0, 1, \dots, k; \quad j = 1, \dots, n_i, \quad (2.4)$$

where for the  $j^{\text{th}}$  subject in the  $i^{\text{th}}$  group,  $y_{ij}$  is the response of the subject,  $d_i$  is the dose the subject receives,  $f(d_i, \boldsymbol{\theta})$  is the mean response of the group given a certain D-R model,  $f(\cdot)$ , with a parameter vector,  $\boldsymbol{\theta}$ , and  $\epsilon_{ij}$  is the independent normal error.  $f(\cdot)$  is standardized by reparameterizing the vector,  $\boldsymbol{\theta}$  into  $\boldsymbol{\theta}^*$ . Furthermore, a location parameter,  $\theta_0$ , and a scale parameter,  $\theta_1$  are incorporated as follows:

$$f(d_i, \boldsymbol{\theta}) = \theta_0 + \theta_1 f^0(d_i, \boldsymbol{\theta}^*), \quad (2.5)$$

where  $f^0(d, \boldsymbol{\theta}^*)$  denotes a suitably standardized model. The test statistics are defined by optimal contrasts. Using such statistics, one can decide which D-R models give us statistically significant fits. Specifically, suppose that one pre-specifies a set  $\mathcal{M}$  of  $M$  parameterized candidate models,  $f_m(d_i, \boldsymbol{\theta})$ ,  $m = 1, \dots, M$ , and that each model is standardized. Each D-R shape can be tested by a single contrast test associated with optimized coefficients obtained from each standardized model (see Branson, et al. and Bretz et al.). These coefficients are chosen in advance to maximize the power of the test when the true underlying mean response at each dose is the modeled value. The single contrast tests,  $m = 1, \dots, M$ , are by

$$T_m = \frac{\sum_{i=0}^k c_{mi} \bar{y}_i}{S \sqrt{\sum_{i=0}^k \frac{c_{mi}^2}{n_i}}}, \quad m = 1, \dots, M, \quad (2.6)$$

where  $S^2 = \frac{\sum_{i=0}^k \sum_{j=1}^{n_i} (y_{ij} - \bar{y}_i)^2}{N - k - 1}$ ,  $N = \sum_{i=0}^k n_i$ , and  $c_{m0}, \dots, c_{mk}$  are the optimal contrasts for model  $m$ . The contrasts are chosen to satisfy  $\sum_{i=0}^k c_{mi} = 0$ . The MCP to test whether or not

there is a dose-response is based on the maximum of  $T_1, \dots, T_M$  via these standardized models. To control the FWER, the multiplicity adjusted p-value and critical values can be numerically computed based on the assumption that the multiple test statistics,  $(T_1, \dots, T_M)'$ , are jointly multivariate  $t$ -distributed under the null hypothesis of no dose-response with correlation matrix determined by the model contrasts. If the adjusted p-value is smaller than a pre-chosen significant level  $\alpha$ , then one rejects the null hypothesis and establishes the PoC. If the null hypothesis cannot be rejected, then the procedure stops indicating that a D-R relationship cannot be established from the observed data. The MCP is useful because it accounts for the uncertainty of model specifications at the beginning of the trial.

Once the PoC is established, the model and dose selection is performed in the Mod step. After the controlled PoC tests, one takes statistically significant models from the candidate model set,  $\mathcal{M}$ . The best model is selected from these significant models based on its having the minimum p-value of the contrast test statistic. Other model selection criteria such as the Akaike Information Criterion (AIC) or the Bayesian Information Criterion (BIC) can also be used. As described in Bretz, et al. [6] the selected D-R model is used to estimate a target dose via inverse regression techniques, typically incorporating information about clinically relevant effects.

## **2.5 MODIFICATION OF ADDING AND/OR DROPPING TREATMENT ARMS IN ADAPTIVE DESIGNS**

Phillips [28] commented that the flexibility in adding treatment arms or dropping treatment arms based on efficacy or safety information is generally acceptable and considered to be advantageous. The biggest concerns are how to control the overall type I error and how to estimate confidence intervals. Additionally, in a dose-response study, how to integrate dose-response modeling in an adaptive design is another issue.

Both Hung, et al. [19] and Hommel [18] discussed the modification of dropping arm(s) and overall type I error in terms of testing an effective treatment dose. Hung et al. commented that it may be advisable to redistribute the unused sample size of the dropped arms to the rest of the retained arms. This may allow one to increase statistical power in detecting an effective dose. For establishing PoC, on the other hand, to our knowledge, there is not much discussion on dropping and/or adding arms. We employ the idea of sample redistribution for testing PoC in our adaptive design and evaluate if power is retained or improved from fixed designs of the same total sample sizes via simulations in Section 3.4.

## 2.6 CONDITIONAL ERROR FUNCTION

Employing the adaptive test procedure by Bauer and Köhne [2], Bretz, et al. [7] conceptualized the use of combination tests in the ‘seamless’ Phase II/III adaptive trials where a conditional error function (CEF) is applied. ‘Seamless’ is usually used to describe the adaptive designs that combine two trials of two phases into one protocol. CEF is defined as the probability of rejecting a global null hypothesis given the p-value of Stage 1 in a two-stage adaptive design.

A decision rule for rejecting the global null hypothesis of group  $i$ ,  $H_g$ , can be formulated using the CEF of the combination test as follows:

reject  $H_g$  if  $p_2 < A(p_1)$ ;

$$A(p_1) = P_H(\text{reject } H | p_1) = \begin{cases} 1 & \text{if } p_1 \leq \alpha_1 \\ 0 & \text{if } p_1 \geq \alpha_0 \\ \max\{p_2 | C(p_1, p_2) \leq \mathcal{C}_\alpha\} & \text{if } p_1 \in (\alpha_1; \alpha_0) \end{cases} \quad (2.7)$$

where  $p_1$  and  $p_2$  are the p-values from testing the null hypotheses in Stage 1 and Stage 2, respectively. Also,  $\alpha_0$  and  $\alpha_1$  are the stopping boundaries at the interim analysis for futility and efficacy, respectively, and  $\mathcal{C}_\alpha$  is a pre-chosen critical value at some  $\alpha$  level.

The above CEF,  $A(p_1)$ , is useful for combining any kind of statistics when one specifies the function form of its combination test,  $C(p_1, p_2)$ . A combination function which is commonly used in practice employs the weighted inverse normal method, that is,

$$C(p_1, p_2) = 1 - \Phi[\omega_1 \Phi^{-1}(1 - p_1) + \omega_2 \Phi^{-1}(1 - p_2)] \text{ where } \omega_1^2 + \omega_2^2 = 1 \text{ and } \omega_1, \omega_2 > 0$$

[15]. In order to use this method, one must pre-specify weights. An alternative choice is Fisher's combination test as discussed in Bauer and Kieser [1]. We follow their method and test the same set of pre-specified candidate models in both stages, meeting their assumptions. Fisher's combination test given by

$$C(p_1, p_2) = p_1 p_2 \leq \mathcal{C}_\alpha = \frac{\exp(-\chi_{4,1-\alpha}^2)}{2}$$

uses a  $\chi^2$ -statistic for the pooled two-stage data which is equal to  $\mathcal{X}^2 = \sum_{j=1}^2 \mathcal{X}_j^2 = -\sum_{j=1}^2 \ln p_j$  [15]. In this test, it is assumed that both  $p_1$  and  $p_2|p_1$  follow the uniform distribution  $(0, 1)$  under the null hypotheses in Stage 1 and Stage 2, respectively. The general concept of Bretz, et al. can be applied for a global establishment of PoC in a two-stage dose adaptive design with an adding and/or dropping treatment adaptation rule as well. In the establishment of the PoC, the hypothesis tests use optimal model contrasts.

The PISC of PhRMA [4] studied the power of each design with respect to the general objectives of dose-response studies. We hypothesized that, if design modifications are introduced to the analytically flexible approaches, we may improve statistical power. We introduce a design modification into one of the analytically flexible approaches and aim to gain better power for establishing PoC. Considering that Bayesian approaches are not currently preferred in drug trials [19], we consider modifying the design in one of the frequentist approaches. We will discuss the details of our approach in the next chapter.

### 3.0 EXTENDED MULTIPLE COMPARISON PROCEDURES AND MODELING APPROACH

Our methodology extends the MCP-Mod method from a classical one-stage design into a two-stage adaptive design under an adding and/or dropping treatment adaptation rule (ADTAR) in order to establish global PoC. We combine the results of the preliminary PoC test in Stage 1 and the adaptive PoC test in Stage 2. The contrast test statistics of Stage 2 are adjusted using weights which are computed according to the ADTAR in Stage 2. To combine the results of two stages, we apply the conditional error function (CEF) reviewed in Section 2.6.

#### 3.1 METHODOLOGY

##### 3.1.1 Overview

Our extended MCP-Mod method, Ex-MCP-Mod, has two major features. The first feature is that through ADTAR, candidate dose combination choices for Stage 2 are pre-specified and we adaptively select one dose combination based on the result of Stage 1. From a statistical point of view, ADTAR allows us to pre-formulate all candidate null hypotheses in the protocol for the PoC test of Stage 2, and we adaptively select one null hypothesis for Stage 2 after Stage 1 is performed. This pre-specification follows the recommendation of the FDA's draft guidance [13] as well.

The second feature is that ADTAR does not necessarily associate the same number of potential results of Stage 1 to each candidate dose combination in Stage 2. Therefore, even though one trial run assigns equal sample sizes to each adaptation choice, over many trial runs one can end up with different aggregate sample sizes in different adaptation choices and dose groups. We model this issue as missing data mechanism in Stage 2 and apply a weighting method for the Stage 2's contrast test statistics to compensate the effect of imbalanced sample allocation.

With these features, we aim to improve design power in the global PoC test over conventional and fixed two-stage designs. Statistical power of the Ex-MCP-Mod and the latter designs is evaluated for four different dose-response patterns through our simulation studies in Section 3.4. The Ex-MCP-Mod is shown to be a more robust method for establishing global PoC.

### 3.1.2 Adding and/or Dropping Treatment Adaptation Rule (ADTAR)

First, we mathematically describe the ADTAR. In the trial design, suppose that there are  $M$  candidate D-R models,  $f_1, \dots, f_M$ , and  $k$  potential doses. In Stage 1, we pre-specify treatment doses to use from  $k$  doses. In Stage 2, treatment doses will be adaptively selected from  $k$  doses based on the ADTAR. A placebo group,  $d_0$ , is always used throughout the trial. When we apply the PoC step of the MCP-Mod in Stage 1, each D-R model will be either significant or non-significant in the preliminary PoC test. Hence, a 'result' is associated with a  $M$ -vector consisting of 0's and 1's where 1 denotes a significant test result for a given model. Therefore,  $\mathcal{K}_1 = 2^M$  different results are possible for the test in Stage 1. In Stage 2, if we do not count the possible adaptation with no treatment doses, then there are  $\mathcal{K}_2 = 2^k - 1$  possible sets of dose that could be examined in this stage. Using an ADTAR, we may associate some of the possible PoC results in Stage 1 with one of the adaptive dose combinations in Stage 2. Suppose that, under an ADTAR, we specify a set of  $L$  adaptation choices,  $A = \{a_1, \dots, a_L\}$  where  $a_j$  denotes choice  $j$ . Each of  $L$  adaptations is pre-specified

from  $\mathcal{K}_2^* \leq \mathcal{K}_2$  possibilities. We denote  $b$  as one potential result in Stage 1 and  $B$  as the set of all possible results in Stage 1, i.e.,  $b \in B = \{b_1, \dots, b_{\mathcal{K}_1}\}$ .  $B$  is then partitioned into  $B_1 \cup B_2 \cup \dots \cup B_L$  according to the  $L$  adaptations, where  $B_i \cap B_j = \emptyset$  for  $i \neq j$ .  $\ell_j$  potential PoC results in Stage 1 are associated with the  $j^{\text{th}}$  adaptation choice,  $a_j$ , so that  $\sum_{j=1}^L \ell_j = \mathcal{K}_1$ . ADTAR is defined by an adaptation function,  $g : B \rightarrow A$ . We show that a pre-specified ADTAR in a two-stage adaptive design helps to control the FWER in a global PoC test in Section 3.3.

Test statistics for Stage 1 and Stage 2 can be obtained using the same procedure as in the MCP-Mod method. Recall from Section 2.4 that the original MCP-Mod method assumed that the multiple test statistics,  $(T_1, \dots, T_M)'$  as defined in 2.4, are jointly multivariate  $t$ -distributed under the null hypothesis of no dose-response. Next we discuss the same approach in two-stage designs. Let  $(T_{s,1}, \dots, T_{s,M})'$  denote a vector of the contrast test statistics in Stage  $s$  ( $s = 1, 2$ ). Further, let  $\text{MVT}_{M, \nu_s}$  denote a  $M$ -dimensional multivariate  $t$ -distribution with  $\nu_s$  degrees of freedom and a numerically computed correlation matrix,  $\mathbf{R}_s = (\rho_{uv})_s$ , where the matrix elements for Stage 1 and Stage 2 are given by

$$\rho_{uv,1} = \frac{\sum_{i=0}^{k_1} \frac{c_{ui}c_{vi}}{n_{1i}}}{\sqrt{\sum_{i=0}^{k_1} \frac{c_{ui}^2}{n_{1i}}} \sqrt{\sum_{i=0}^{k_1} \frac{c_{vi}^2}{n_{1i}}}} \quad \text{and}$$

$$\rho_{uv,2} = \frac{\sum_{i=0}^{k_2} \frac{c_{ui}^*c_{vi}^*}{n_{2i}}}{\sqrt{\sum_{i=0}^{k_2} \frac{c_{ui}^{*2}}{n_{2i}}} \sqrt{\sum_{i=0}^{k_2} \frac{c_{vi}^{*2}}{n_{2i}}}}, \quad 1 \leq u, v \leq M.$$

The contrast vectors,  $\mathbf{c}_m = (c_{m0}, \dots, c_{mk_2})'$  and  $\mathbf{c}_m^* = (c_{m0}^*, \dots, c_{mk_2}^*)'$ , are optimal in Stage 1 and Stage 2, respectively. Recalling that  $\nu_s = N_s - k_s - 1$  where  $N_s$  is the total sample size in Stage  $s$  and  $k_s$  is the number of treatment doses in Stage  $s$ , then, under the null hypothesis, for Stage 1,  $(T_{1,1}, \dots, T_{1,M})' \sim \text{MVT}_{M, \nu_1}(\mathbf{0}, \mathbf{R}_1)$ , and for Stage 2,  $(T_{2,1}, \dots, T_{2,M}|a_j)' \sim \text{MVT}_{M, \nu_2}(\mathbf{0}, \mathbf{R}_2)$ . In summary, in both Stage 1 and Stage 2 given a selected adaptation choice, we can apply the MCP-Mod method based on the same assumption. In the following section, we show how to treat the dependency of Stage 2 on Stage 1 by modeling imbalanced aggregate sample sizes in Stage 2 as missing data mechanism.

### 3.1.3 Modeling Stage 2 through Missing Data Analysis

If an ADTAR is used in a two-stage design, then Stage 2 depends on the function  $g$  which uses a result  $b \in B_j$  to construct the choice of adaptation  $a_j$ . Across multiple trials, some doses may not be sampled in each trial’s Stage 2. Hence, over multiple trials, the aggregate samples for different doses in Stage 2 could be imbalanced depending on both the result in Stage 1 and the subsequent adaptation employed. In our setting, we can view the adaptation for Stage 2 as creating a missing data mechanism [25]. Accordingly, the mean estimate for dose  $i$  can be adjusted by applying a missing data analysis method. The missing data mechanism within Stage 2 is Missing At Random (MAR) because the missingness with respect to the doses depends only on the observed part of the  $\mathcal{K}_1$  potential PoC results in Stage 1. An example illustrating the missing data mechanism is shown in Section 3.4.3. Since the adaptation choice determines which doses are ‘missing’, we can then condition on that choice rendering the missing data mechanism as Missing Completely At Random (MCAR), i.e., quasi-randomized [25]. Under the assumption of quasi-randomization, one can use the weighting class estimator of missing data analysis. This unbiased estimator uses the idea that the observations associated with adaptation choice,  $a_j$  (‘stratum  $j$ ’ by Little and Rubin), receive a weight for each dose group,  $i$ . We let  $w_{ij}$  denote the weight which corresponds to the mean estimate of dose  $i$  for adaptation  $a_j$ . In Section 3.2, we derive the weights,  $w_{ij}$ , for dose  $i$  and adaptation  $a_j$ .

Next, we demonstrate how the weights are used in the PoC test statistics in Stage 2. In our method, at the end of each stage, the PoC step of the MCP-Mod is applied as a preliminary test (Section 3.1.2). Under the null hypothesis of no dose-response relationship, the  $M$  contrast test statistics jointly follow a multivariate  $t$ -distribution in each stage. Since Stage 2 depends on the ADTAR function, we adjust the contrast test statistics in Stage 2 using weight  $\omega_{ij}$ . The adjusted statistic on model  $m$  for adaptation  $a_j$  in Stage 2 is

$$T_{2j,m} = \frac{\sum_{i=0}^{k_2} \omega_{ij} c_{mi}^* \bar{y}_{ij}}{S_2 \sqrt{\sum_{i=0}^{k_2} \frac{(\omega_{ij} c_{mi}^*)^2}{n_{2i}}}}, \quad m = 1, \dots, M,$$

where  $S_2^2 = \frac{\sum_{i=0}^{k_2} \sum_{h=1}^{n_{2i}} (y_{hij} - \bar{y}_{ij})^2}{N_2 - k_2 - 1}$ . Consequently, in Stage 2, the relationship  $(T_{2,1}, \dots, T_{2,M} | a_j)' \sim \text{MVT}_{M, \nu_2}(\mathbf{0}, \mathbf{R}_2)$  can be equivalently written as  $(T_{2j,1}, \dots, T_{2j,M})' \sim \text{MVT}_{M, \nu_2}(\mathbf{0}, \mathbf{R}_2^*)$  where  $\mathbf{R}_2^* = (\rho_{uv}^*)_2$ ,

$$\rho_{uv,2}^* = \frac{\sum_{i=0}^{k_2} \frac{\omega_{ij}^2 c_{ui}^* c_{vi}^*}{n_{2i}}}{\sqrt{\sum_{i=0}^{k_2} \frac{\omega_{ij}^2 c_{ui}^{*2}}{n_{2i}}} \sqrt{\sum_{i=0}^{k_2} \frac{\omega_{ij}^2 c_{vi}^{*2}}{n_{2i}}}},$$

and  $1 \leq u, v \leq M$ . By using the above weighted statistics and correlations, we compensate the effect of imbalanced sample allocation due to ADTAR.

### 3.1.4 Combining Stage 1 and Stage 2

In the last step, the same way as in the original MCP-Mod method (Section 2.4), we use the multiple contrast test statistics and obtain a p-value in each stage by use of multivariate t-distributions, i.e.,  $(T_{1,1}, \dots, T_{1,M})' \sim \text{MVT}_{M, \nu_1}(\mathbf{0}, \mathbf{R}_1)$  in Stage 1 and  $(T_{2j,1}, \dots, T_{2j,M})' \sim \text{MVT}_{M, \nu_2}(\mathbf{0}, \mathbf{R}_2^*)$  in Stage 2. Let  $p_1$  and  $p_2$  denote the obtained p-values in Stage 1 and Stage 2, respectively. In the global PoC test, we plug these p-values into the CEF formula from the combination technique (Section 2.6) to test the global null hypothesis of no dose-response relationship. The global PoC is established if  $p_2 < A(p_1) = \frac{C_\alpha}{p_1}$  where  $C_\alpha$  is a pre-chosen critical value.

## 3.2 DERIVATION OF WEIGHTS

In this section, we show the derivation of the weight for adjusting the mean response of dose group  $i$  and adaptation choice  $j$  in Stage 2 given an adaptive two-stage design with an ADTAR function. Suppose that we perform a preliminary PoC test using a set of  $M$  candidate dose-response models in Stage 1 and Stage 2, respectively. Let  $B$  denote the set containing all possible PoC results in Stage 1, i.e.,  $B = \{b_1, \dots, b_{\mathcal{K}_1}\}$  where  $\mathcal{K}_1 = 2^M$ . Remember that, under the ADTAR, we pre-specify a set of  $L$  adaptation choices,  $A =$

$\{a_1, \dots, a_L\}$  where  $a_j$  denotes choice  $j$ .  $B$  is partitioned into  $B_1 \cup B_2 \cup \dots \cup B_L$  where  $B_i \cap B_j = \emptyset$  for  $i \neq j$ . If we let  $|B_j| = \ell_j$ , then  $\sum_{j=1}^L |B_j| = \sum_{j=1}^L \ell_j = \mathcal{K}_1$ . ADTAR is defined by an adaptation function,  $g : B \rightarrow A$ .

Let  $\mathbf{y}_b = (y_{b0}, \dots, y_{bk})$  denote a vector of  $k + 1$  dose responses in Stage 2 associated with one potential result  $b$ ,  $b \in B$ , in Stage 1. Let  $\mathbf{Y} = (y_{bi})$ . For result  $b$ , we define  $Q_{bj}$  as the sample indicator of Stage 2.

$$Q_{bj} = \begin{cases} 1 & \text{if } j = g(b), \\ 0 & \text{otherwise.} \end{cases}$$

$Q_{bj}$  is used to determine the sample set of Stage 2. Let  $\mathbf{Q}_j = (Q_{1j}, \dots, Q_{\mathcal{K}_1j})$ . The dose adaptation process is characterized by a distribution for  $\mathbf{Q}_j$  given both  $\mathbf{Y}$  and the adaptive design information,  $\mathbf{Z}$ . In order to infer quasi-randomization in Stage 2, the following two assumptions are required [25]:

1. The distribution of the random indicator variables,  $\mathbf{Q}_j$ , is determined before any values of  $\mathbf{Y}$  in Stage 2 are known. Thus,  $f(\mathbf{Q}_j|\mathbf{Y}, \mathbf{Z}) = f(\mathbf{Q}_j|\mathbf{Z})$ .
2. Every result  $b$  has a positive (known) probability of being selected. Hence, if  $\tau_{bj} = E(Q_{bj}|\mathbf{Y}, \mathbf{Z}) = Pr(Q_{bj} = 1|\mathbf{Y}, \mathbf{Z})$ , then  $\tau_{bj} > 0$  is required for all  $b$ .

In this context, our adaptation design matrix  $\mathbf{Z}$  is a  $L \times (k + 1)$  matrix indicating the  $L$  adaptation ‘strata’ for the  $k + 1$  doses. Then, both of the assumptions are held and  $\tau_{bj} = E(Q_{bj}|\mathbf{Y}, \mathbf{Z}) = E(Q_{bj}|\mathbf{Z}) = E(Q_{bj}|j = g(b)) = 1$ . The idea expressed in this equation is as follows: the probability of  $j = g(b)$  under the condition that  $j = g(b)$  is 1. This statement is the second condition for using the missing data mechanism. Following Little and Rubin, the trial is encoded as  $\mathbf{Z}$  and the ADTAR function  $g$  is encoded as  $Q_{bj}$ . Using  $\mathbf{Z}$  and  $Q_{bj}$ , our condition can be expressed as  $Pr(Q_{bj} = 1|\mathbf{Y}, \mathbf{Z})$ . Thus,  $\tau_{bj} = 1$ .

Let  $I_{ij}$  be an indicator variable and the  $ji^{th}$  element of the design matrix,  $\mathbf{Z}$ , such that  $\mathbf{Z}_{L \times (k+1)} = (I_{ij})'$ . According to the ADTAR function,  $I_{ij}$  is defined as follows.

$$I_{ij} = \begin{cases} 1 & \text{dose group } i \text{ is included in the adaptation choice } j, \\ 0 & \text{otherwise,} \end{cases}$$

where  $i = 0, \dots, k$  and  $j = 1, \dots, L$ . Now, let  $n_b$  denote the sample size per dose group in Stage 2 given result  $b$  in Stage 1. Furthermore, let  $n'_j$  denote the population size per dose group in Stage 2 in stratum  $j$ . If equal sample sizes are assumed in Stage 2 and if the total sample size is  $N$  in Stage 2, then  $n'_j = \sum_{g^{-1}(j)} n_b = \ell_j n_b$  and  $n_b = \frac{N}{\sum_{i=0}^k I_{ig(b)}}$ . If result  $b$  in Stage 1 is associated with adaptation  $a_j$ , for each dose group  $i$  a stratified random sampling takes a simple random sample scheme of  $n_b$  units from  $n'_j$  population units within stratum  $j$ . In a typical weighting approach, if a unit selected from a target population with probability  $\pi$  represents  $\pi^{-1}$  units in the population, then the unit should be given the weight  $\pi^{-1}$  in estimates of population quantities. In our particular case,  $\pi_{ij}^{-1}$  is used as the weight for the population of dose  $i$  in stratum  $j$ . In Stage 2, let  $\varphi_{ib}^{-1}$  be the weight for a selected unit of dose  $i$  in the  $j^{\text{th}}$  adaptation based on result  $b$ . Since function  $g$  maps result  $b$  to stratum  $j$ ,

$$\varphi_{ib}^{-1} = \begin{cases} \left(\frac{n_b}{n'_j}\right)^{-1} = \ell_j & \text{if } I_{ij} = 1, \\ \text{does not exist} & \text{if } I_{ij} = 0, \end{cases} \quad \text{with } j = g(b).$$

Using  $\varphi_{ib}$ ,  $\pi_{ij} = \varphi_{ib} \tau_{bj}$ . The Horvitz-Thompson weighted estimator of the population total of dose  $i$ , say  $T_i$ , over the adaptation choices uses stratum weight,  $\pi_{ij}$ , as follows.

$$t_{i \text{ HT}} = \sum_{j=1}^L \pi_{ij}^{-1} y_{ij}. \quad (3.1)$$

As a result, the stratified mean of dose  $i$  is written as:

$$\bar{y}_{i \text{ st}} = \frac{1}{L} \sum_{j=1}^L \phi_{ij} y_{ij}, \quad (3.2)$$

where  $\phi_{ij} = \frac{L\pi_{ij}^{-1}}{\sum_{v=1}^L \pi_{iv}^{-1}}$ . Let  $\bar{y}_i^{(w)}$  denote our weighting class estimator of mean response for dose group  $i$ , and let  $\bar{y}_{ij}$  be the mean response of dose  $i$  in adaptation  $a_j$  in Stage 2.

Furthermore, let  $\phi_{ijb}$  denote the weight for dose  $i$  based on result  $b \in B_j$ , then from (3.2)

$\bar{y}_i^{(w)} = \frac{1}{L} \sum_{j=1}^L \phi_{ijb} \bar{y}_{ij}$ . This estimator is further simplified as follows.

$$\begin{aligned} \bar{y}_i^{(w)} &= \frac{1}{L} \sum_{j=1}^L \phi_{ijb} \bar{y}_{ij} \\ &= \frac{1}{L} \sum_{j=1}^L \frac{L(\varphi_{ib} \tau_{bj})^{-1}}{\sum_{v=1}^L (\varphi_{ib} \tau_{bv})^{-1}} \bar{y}_{ij} \\ &= \sum_{j=1}^L \frac{\ell_j I_{ij} \tau_{bj}^{-1}}{\sum_{v=1}^L \ell_v I_{iv} \tau_{bv}^{-1}} \bar{y}_{ij}. \end{aligned}$$

In light of the form of weighted estimator (3.1),  $w_{ij}$  is derived as

$$\begin{aligned} w_{ij} &= \frac{\ell_j I_{ij} \tau_{bj}^{-1}}{\sum_{v=1}^L (\ell_v I_{iv} \tau_{bv}^{-1})} \\ &= \frac{\ell_j I_{ij}}{\sum_{v=1}^L (\ell_v I_{iv})}. \end{aligned} \tag{3.3}$$

### 3.3 CONTROLLING THE FWER CLOSE TO A PRE-CHOSEN $\alpha$ -LEVEL

‘Strong control of the FWER’ is the concept that the probability of erroneously rejecting a true null hypothesis does not exceed a pre-chosen confidence level,  $\alpha$ , in a multiple level- $\alpha$  test, irrespective of how many and which are in fact true [1]. Strong control of the FWER is a useful property for an experimental system using an adaptive two-stage design to be valid. However, with its performance, there would be a concern that the trial design becomes so conservative that the statistical power in establishing global PoC gets much lower than we prefer. In this section we show that a pre-specified ADTAR allows us to control the FWER close to a pre-chosen  $\alpha$ -level in the global PoC test in an adaptive two-stage design.

To construct our proof, we will use a modification of the argument outlined in Bauer and Kieser [1]. Let  $p_1$  be the  $p$ -value of an overall PoC test for  $H_{0,\mathcal{M}}$  in Stage 1, i.e., there is no dose-response relationship within the model set,  $\mathcal{M} = \{f_1, \dots, f_M\}$ , in Stage 1.

Suppose that, to make an adaptation decision, we look at a preliminary PoC result,  $b \in B = \{b_1, \dots, b_{2M}\}$  in Stage 1. Let  $G_{2j} \in G_2$  denote a test scenario which employs a specific distributional assumption. Let  $G_{2j}$  be chosen from a finite set of test scenarios,  $G_2$ , for testing  $H_{0,\mathcal{M}}$  in Stage 2. In our two-stage design with ADTAR, the choice of  $G_{2j}$  are determined by both the number of adaptation choices and weights  $\omega_{ij}$  (determined by missing data mechanism due to ADTAR); the weights are specified based on the selected dose adaptation,  $a_j \in A = \{a_1, \dots, a_L\}$ .

In our setting, we define the adaptation function,  $g : B \rightarrow A$  for all candidate models, in the adaptation protocol. Under  $H_{0,\mathcal{M}}$ , let  $p_2^M$  be the  $p$ -value of an overall PoC test in Stage 2 and let  $h_0(p_2^M | G_{2j}, p_1)$  be the conditional probability density of  $p_2^M$  in the adaptive procedure given  $p_1 \in (\alpha_1, \alpha_0)$  and the choice  $G_{2j}$ . If the PoC test under  $G_{2j}$  is implemented in a stochastically independent sample, then  $p_2^M$  is uniformly distributed on  $[0,1]$  for any given  $G_{2j}$  and  $p_1 \in (\alpha_1, \alpha_0)$ . Accordingly,  $h_0(p_2^M | G_{2j}, p_1) = 1$ . The detailed proof of this property is located in the appendix of Bauer and Kieser [1]. The consequence of this property is that, conditioning on  $a_j$ , the specification of the overall PoC test in Stage 2 for  $H_{0,\mathcal{M}}$  is determined prior to Stage 2, and then, the vector of test statistics,  $(T_{2j,1}, \dots, T_{2j,M})'$ , based on the corresponding weighted multivariate  $t$ -distribution,  $\text{MVT}_{M,\nu_2}(\mathbf{0}, \mathbf{R}_2^*)$  where

$$\mathbf{R}_2^* = (\rho_{uv}^*)_2, \quad \rho_{uv,2}^* = \frac{\sum_{i=0}^{k_2} \frac{\omega_{ij}^{(u)} \omega_{ij}^{(v)} c_{ui}^* c_{vi}^*}{n_{2i}}}{\sqrt{\sum_{i=0}^{k_2} \frac{\omega_{ij}^{(u)2} c_{ui}^{*2}}{n_{2i}}} \sqrt{\sum_{i=0}^{k_2} \frac{\omega_{ij}^{(v)2} c_{vi}^{*2}}{n_{2i}}}}, \quad 1 \leq u, v \leq M,$$

and  $\nu_2 = N_2 - k_2 - 1$ , is computed at the end of Stage 2.

Let  $q_0(G_{2j} | p_1)$  denote the conditional probability of choosing  $G_{2j}$  given  $p_1$ . This conditional probability is defined for  $p_1 \in (\alpha_1, \alpha_0)$ . Because  $p_1$  is uniformly distributed and  $h_0(p_2^M | G_{2j}, p_1) = 1$ , the type I error rate of the overall PoC test for  $H_{0,\mathcal{M}}$  over the two

stages, namely, the global PoC test, is controlled at a prechosen  $\alpha$ -level as follows:

$$\begin{aligned}
P_{H_{0,\mathcal{M}}}(\text{reject } H_{0,\mathcal{M}}) &= \int_0^{\alpha_1} dp_1 + \int_{\alpha_1}^{\alpha_0} \left[ \sum_{j \in G_2} \int_0^{\mathcal{C}_{\alpha_2/p_1}} h_0(p_2^M | G_{2j}, p_1) q_0(G_{2j} | p_1) dp_1 \right] dp_2^M \\
&= \alpha_1 + \int_{\alpha_1}^{\alpha_0} \left[ \sum_{j \in G_2} q_0(G_{2j} | p_1) \right] (\mathcal{C}_{\alpha_2/p_1}) dp_1 \\
&\leq \alpha_1 + \int_{\alpha_1}^{\alpha_0} (\mathcal{C}_{\alpha_2/p_1}) dp_1 = \alpha.
\end{aligned}$$

Note that the last inequality holds because the set of testing methods in Stage 2,  $G_2$ , is determined by the range of  $j$ .  $a_j$  does not need to cover the full space of all possible PoC test scenarios and thus is associated with a probability  $\leq 1$ . Therefore, pre-specifying  $g$ , adding and/or dropping of doses, controls the FWER closed to the pre-chosen  $\alpha$ -level in an adaptive two-stage design.

### 3.4 SIMULATION STUDY: EVALUATING EX-MCP-MOD

To evaluate our approach's performance for detecting a D-R curve in the proposed design, we compared it to the one-stage design of the original MCP-Mod method and four fixed two-stage designs via simulations of 10,000 trials. To evaluate our missing data weighting procedure, we also compared the proposed design to the same ADTAR design without weighting via simulations. Figure 1 illustrates the proposed design and compared study designs. Note that the fixed two-stage designs have two components, Study 1 and Study 2. Study 1 of each fixed two-stage design is identical to Stage 1 in the adaptive design. Study 2 of each fixed two-stage design is identical to one of the adaptive choices of Stage 2 in the adaptive design. Then, Study 1 and Study 2 are chosen independently of each other. Overall performance was measured by quantifying the probability of detecting a D-R curve in 16 different scenarios, which were the combinations of four different models (E<sub>max</sub>, Logistic, Quadratic, and Constant) and four different total sample sizes ( $N = 80, 200, 400, \text{ and } 600$ )

for each study design. In the proposed design using an example of ADTAR, ADTAR I, the equal sample sizes for group  $i$  in Stage 1 were  $n_{1i} = 10, 25, 50,$  and  $75$ . In all the adaptations, balanced sample sizes for each group in Stage 2 ( $n_{2i}$ ) were chosen so that the total number of observations between Stage 1 and Stage 2 would be the same. Note that type I error rates are merely probabilities of detecting a significant D-R curve under the null hypothesis of a flat curve. Thus, similar to Branson, et al. [5] and Bretz, et al. [6], we used this assessment method to evaluate our performance with respect to ‘proof of activity (PoA)’. The common components of the simulation design and ADTAR I are shown below. We used the same notation described in Section 3.1.

### 3.4.1 Simulation Setup

- Data generation: Four datasets were generated based on a population mean vector of Emax, Logistic, and Quadratic models, respectively. Each model specification was the same as one of the candidate D-R models. All the models had the property that a maximum effect size within the interval  $[0, 1]$  was 0.8 and that the response at the placebo was 0.2. The constant data-generating model was  $f(d_i) = 0.2$ . We assumed independent outcomes from  $N(f(d_i), \sigma^2)$  at  $d_i$  where  $\sigma = 1.478$ .
- Candidate D-R models: The number of candidate D-R models was three ( $M = 3$ ). The model specifications are summarized on Table 1 and Figure 4 where  $ED_{50} = 0.2$  for  $f_1^0$ ,  $\delta_{\max} = 0.6$  for  $f_2^0$ , and  $ED_{50} = 0.2$  and  $\delta = 0.6$  for  $f_3^0$ .
- Doses: The number of potential doses was four ( $k = 4$ ) as in  $(d_0, d_1, d_2, d_3, d_4) = (0, 0.05, 0.2, 0.6, 1.0)$ .

In Stage 1,  $(d_0, d_1, d_2, d_3) = (0, 0.05, 0.2, 0.6)$  was used under ADTAR I.

In Stage 2, doses were selected based on ADTAR I described below.

- Significance level: We employed a 2.5 % one-sided significance level to achieve a 95 % confidence interval in all PoC tests. This is the same confidence interval used in Bretz, et al. [6]. The mismatch between our significance level and confidence interval is due

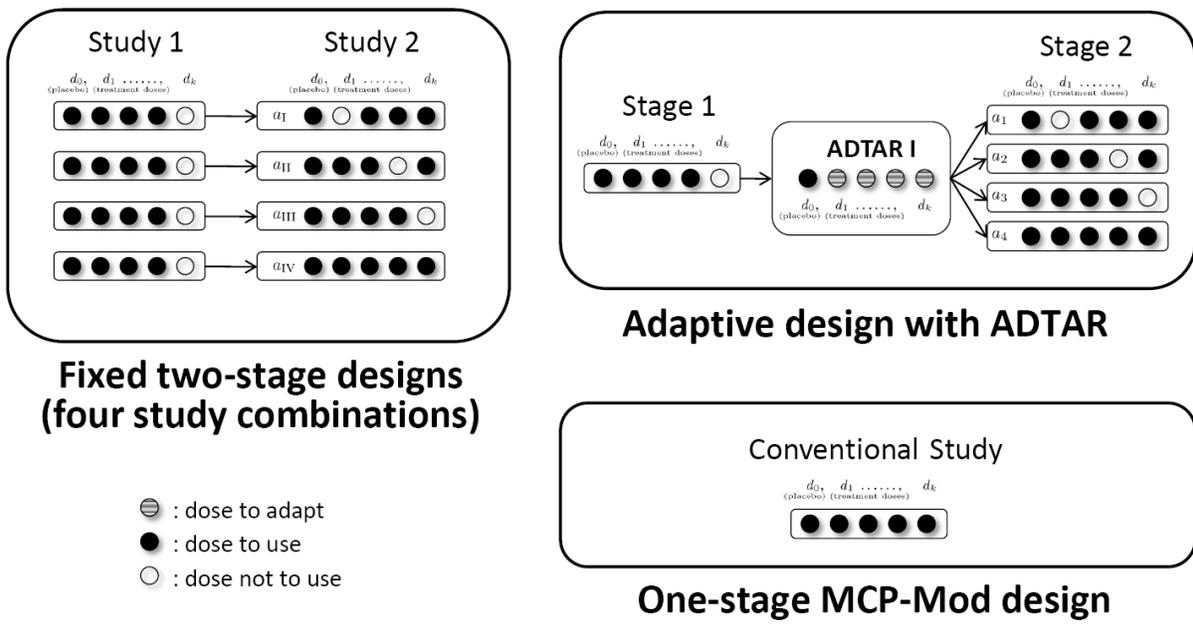


Figure 1: Evaluation: Adaptive Design with ADTAR, Fixed Two-stage Designs, and One-stage MCP-Mod Design

to the fact that we used a one-sided test in order to perform the necessary symmetric two-sided test [5].

- ADTAR I: When ADTAR is required, we used the rule as follows. There were  $\mathcal{K}_1 = 2^M = 8$  possible results in Stage 1 so that  $B = \{b_1, \dots, b_8\}$ . We let the number of adaptation choices be four ( $L = 4$ ) so that the set of adaptations would be  $A = \{a_1, \dots, a_4\}$  (see Table 2). We defined the function of ADTAR I,  $g : B \rightarrow A$ , as follows:

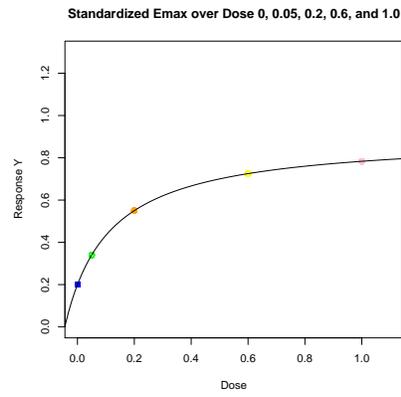
$$g : \begin{cases} b_1, b_2 & \rightarrow a_1 \\ b_4, b_6 & \rightarrow a_2 \\ b_7 & \rightarrow a_3 \\ b_3, b_5, b_8 & \rightarrow a_4 \end{cases}$$

This ADTAR I does not allow stopping after Stage 1. This implies that  $\alpha_1 = 0$  and  $\alpha_0 = 1$  in 2.7. For ADTAR I, the contrast weights are computed using 3.3 and summarized on Table 3.

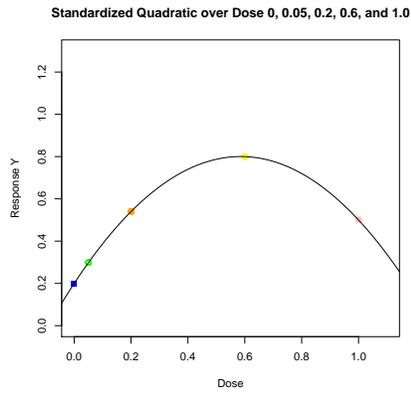
Table 1: Original and Standardized Model Formula

Model	Original formula	Standardized formula
$f_1$ : Emax	$f_1(d) = E_0 + \frac{E_{\max}d}{ED_{50}+d}$	$f_1^0(d) = \frac{d}{ED_{50}+d}$
$f_2$ : Quadratic	$f_2(d) = E_0 + \beta_1d + \beta_2d^2$	$f_2^0(d) = d + \frac{\beta_2}{ \beta_1 }d^2$ $= d - \delta_{\max}d^2$
$f_3$ : Logistic	$f_3(d) = E_0 + \frac{E_{\max}}{1+\exp(\frac{ED_{50}-d}{\delta})}$	$f_3^0(d) = \frac{1}{1+\exp(\frac{ED_{50}-d}{\delta})}$

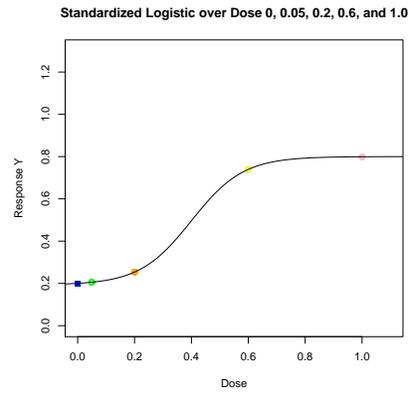
$$d \in \{d_0, d_1, \dots, d_4\}$$



(a)  $f_1$



(b)  $f_2$



(c)  $f_3$

Figure 2: Standardized Emax ( $f_1$ ), quadratic ( $f_2$ ), and logistic ( $f_3$ ) models.

Table 2: ADTAR I

$B_j$	$b$	Significant Models in Stage 1	$l_j$	$a_j$	doses to add	doses to drop	doses used in Stage 2	balanced sample, $n_{2i}$ for $n_{1i} = 25$ ( $n_{2i}$ for $n_{1i} = 75$ )
$B_1$	$b_1$	$f_1, f_2, f_3$	2	$a_1$	$d_4$	$d_1$	$d_0, d_2, d_3, d_4$	25 (75)
	$b_2$	$f_1, f_2$						
$B_2$	$b_4$	$f_2, f_3$	2	$a_2$	$d_4$	$d_3$	$d_0, d_1, d_2, d_4$	25 (75)
	$b_6$	$f_2$						
$B_3$	$b_7$	$f_3$	1	$a_3$	None	None	$d_0, d_1, d_2, d_3$	25 (75)
$B_4$	$b_3$	$f_1, f_3$	3	$a_4$	$d_4$	None	$d_0, d_1, d_2, d_3, d_4$	20 (60)
	$b_5$	$f_1$						
	$b_8$	None						

Table 3: Weights ( $\omega_{ij}$ ) Under The Adaptation Choices in ADTAR I

	$a_1$	$a_2$	$a_3$	$a_4$
$d_0$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{3}{8}$
$d_1$	0	$\frac{1}{3}$	$\frac{1}{6}$	$\frac{1}{2}$
$d_2$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{3}{8}$
$d_3$	$\frac{1}{3}$	0	$\frac{1}{6}$	$\frac{1}{2}$
$d_4$	$\frac{2}{7}$	$\frac{2}{7}$	0	$\frac{3}{7}$

### 3.4.2 Simulation Results

Simulation Results of all 16 scenarios (four different datasets by four total sample sizes) are summarized in Table 4 and Table 5 (The R programs for implementing the simulation studies are available in Appendices A.1, A.2, and A.3). When we italicize, *ADTAR I* denotes the proposed adaptive two-stage design. Using the same data, we also evaluated PoA of *ADTAR I* without weighting Stage 2’s test statistics. This non-weighting ADTAR design method is denoted as *Non-weighting ADTAR I*. Among the four fixed two-stage designs, *Maximum* denotes the design with the maximum PoA and *Minimum* denotes the design with the minimum PoA, respectively. The one-stage design of the original MCP-Mod method is denoted as *One-stage MCP-Mod*. From both Table 4 and Table 5, the FWER seems to be controlled roughly at a 5% level in all study designs for all total sample sizes. *ADTAR I* showed relatively consistent PoA activities across the three different dose-response curves. Figure 3 shows the global PoA plots of total sample size 200 to compare *ADTAR I*, *Non-weighting ADTAR I*, *Maximum*, *Minimum*, and *One-stage MCP-Mod* for each D-R curve. The results of these comparisons were similar for all sample sizes considered. That is, *ADTAR I* was always more powerful than *Non-weighting ADTAR I* across the three dose-response curves. *ADTAR I* was slightly more powerful than *Maximum* for the Emax and logistic dose-response curves. *One-stage MCP-Mod* also showed higher PoA than *Maximum* for the both dose-response curves. However, its PoA for the quadratic dose-response was lower than *Minimum* while *ADTAR I* showed higher PoA than *Minimum*.

### 3.4.3 Effect of Sample Size Imbalance across Groups

This section evaluates the sample-compensating effect of weighting based on the missing data analysis described in Section 3.1.3. As shown in Figure 4a, we consider a hypothetical case where the eight possible PoC results in Stage 1 ( $b_1, \dots, b_8$ ) are equally likely to occur in a trial using our ADTAR two-stage design. In this example, if the total sample sizes in

Table 4: PoAs of *ADTARI*, *Non-weighting ADTARI*, *One-Stage MCP-Mod*, and four two-stage fixed designs under varying D-R shape  $\times$  sample size (1)

Design Choice	Test Stage	$n_{1i}/\text{gp}$ $n_{2i}/\text{gp}$ ( $N$ )	Data-Generating Function				$n_{1i}/\text{gp}$ $n_{2i}/\text{gp}$ ( $N$ )	Data-Generating Function			
			Const.	E <sub>max</sub>	Quad.	Logist.		Const.	E <sub>max</sub>	Quad.	Logist.
<i>ADTARI</i>	Stage 1	10	0.0369	0.1679	0.2070	0.1811	25	0.0382	0.3207	0.4100	0.3559
	Stage 2	8-10	0.0467	0.2243	0.1648	0.2806	20-25	0.0467	0.4408	0.3266	0.5544
	Global	(80)	0.0499	0.3206	0.3039	0.3791	(200)	0.0483	0.6102	0.5896	0.7134
<i>Non-weighting ADTARI</i>	Stage 2	8-10	0.0450	0.2038	0.1428	0.2550	20-25	0.0422	0.3908	0.2679	0.4928
	Global	(80)	0.0490	0.3089	0.2884	0.3643	(200)	0.0461	0.5888	0.5570	0.6844
<i>One-Stage MCP-Mod</i>	Global	(80)	0.0424	0.3294	0.2262	0.4286	(200)	0.0418	0.6428	0.4484	0.7739
$a_I$	Stage 1	10	0.0397	0.1759	0.2130	0.1888	25	0.0369	0.3299	0.4120	0.3620
	Stage 2	10	0.0455	0.2210	0.1520	0.2650	25	0.0420	0.4171	0.2668	0.5111
	Global	(80)	0.0503	0.3329	0.2999	0.3804	(200)	0.0484	0.6017	0.5614	0.6952
$a_{II}$	Stage 1	10	0.0381	0.1679	0.2103	0.1792	25	0.0368	0.3277	0.4093	0.3528
	Stage 2	10	0.0390	0.2038	0.1012	0.2264	25	0.0366	0.3816	0.1547	0.4462
	Global	(80)	0.0440	0.3044	0.2470	0.3439	(200)	0.0426	0.5907	0.4626	0.6556
$a_{III}$	Stage 1	10	0.0361	0.1704	0.2032	0.1842	25	0.0372	0.3241	0.4067	0.3551
	Stage 2	10	0.0394	0.1716	0.2068	0.1781	25	0.0410	0.3289	0.4118	0.4075
	Global	(80)	0.0447	0.2747	0.3515	0.3043	(200)	0.0444	0.5472	0.6625	0.6238
$a_{IV}$	Stage 1	10	0.0345	0.1688	0.2145	0.1837	25	0.0390	0.3237	0.4213	0.3542
	Stage 2	8	0.0439	0.2078	0.1503	0.2590	20	0.0437	0.4028	0.2702	0.4982
	Global	(80)	0.0501	0.3146	0.2992	0.3723	(200)	0.0466	0.5951	0.5675	0.6888

Table 5: PoAs of *ADTARI*, *Non-weighting ADTARI*, *One-Stage MCP-Mod*, and four two-stage fixed designs under varying D-R shape  $\times$  sample size (2)

Design Choice	Test Stage	$n_{1i}/\text{gp}$ $n_{2i}/\text{gp}$ ( $N$ )	Data-Generating Function				$n_{1i}/\text{gp}$ $n_{2i}/\text{gp}$ ( $N$ )	Data-Generating Function			
			Const.	E <sub>max</sub>	Quad.	Logist.		Const.	E <sub>max</sub>	Quad.	Logist.
<i>ADTARI</i>	Stage 1	50	0.0403	0.5411	0.6667	0.5979	75	0.0393	0.7054	0.8205	0.7601
	Stage 2	40-50	0.0464	0.7327	0.5658	0.8261	60-75	0.0468	0.8865	0.7340	0.9434
	Global	(400)	0.0504	0.8747	0.8555	0.9369	(600)	0.0521	0.9674	0.9527	0.9893
<i>Non-weighting ADTARI</i>	Stage 2	40-50	0.0417	0.6487	0.4409	0.7587	60-75	0.0397	0.8008	0.5722	0.9005
	Global	(400)	0.0478	0.8551	0.8261	0.9205	(600)	0.0473	0.9570	0.9382	0.9845
<i>One-Stage MCP-Mod</i>	Global	(400)	0.0411	0.8879	0.7032	0.9618	(600)	0.0446	0.9716	0.8663	0.9949
$a_I$	Stage 1	50	0.0386	0.5378	0.6685	0.5878	75	0.0422	0.7053	0.8287	0.7508
	Stage 2	50	0.0454	0.6569	0.4245	0.7817	75	0.0447	0.8177	0.5752	0.9117
	Global	(400)	0.0481	0.8645	0.8246	0.9259	(600)	0.0514	0.9601	0.9409	0.9844
$a_{II}$	Stage 1	50	0.0368	0.5422	0.6636	0.5837	75	0.0378	0.7117	0.8231	0.7670
	Stage 2	50	0.0405	0.6306	0.2462	0.7159	75	0.0401	0.7863	0.3355	0.8582
	Global	(400)	0.0477	0.8493	0.7368	0.8999	(600)	0.0450	0.9498	0.8799	0.9772
$a_{III}$	Stage 1	50	0.0396	0.5514	0.6693	0.5883	75	0.0370	0.7098	0.8297	0.7594
	Stage 2	50	0.0379	0.5467	0.6664	0.5888	75	0.0351	0.7074	0.8322	0.7598
	Global	(400)	0.0446	0.8170	0.9070	0.8521	(600)	0.0405	0.9318	0.9793	0.9541
$a_{IV}$	Stage 1	50	0.0421	0.5469	0.6673	0.5853	75	0.0364	0.6961	0.8273	0.7561
	Stage 2	40	0.0448	0.6448	0.4411	0.7750	60	0.0446	0.7991	0.5857	0.8977
	Global	(400)	0.0505	0.8576	0.8268	0.9247	(600)	0.0486	0.9527	0.9397	0.9831

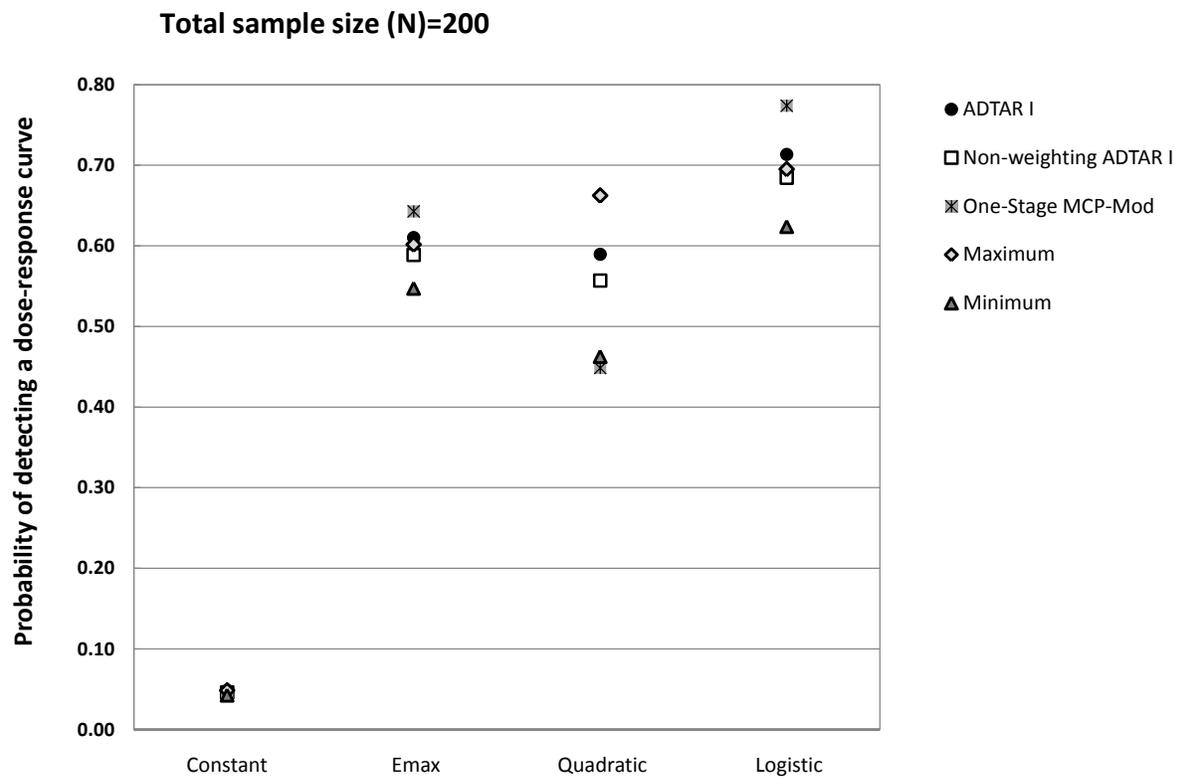
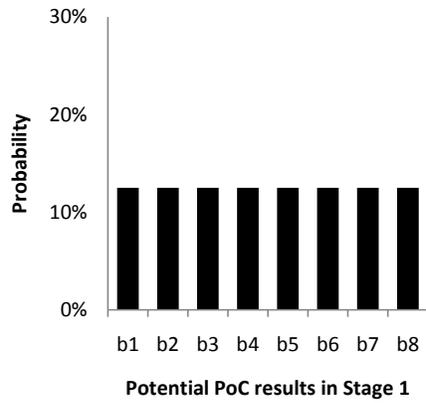


Figure 3: Proof of Activity in Global Stage for Different Dose-Response Patterns in *ADTAR I*, *Non-weighting ADTAR I*, *One-Stage MCP-Mod*, *Maximum*, and *Minimum* designs

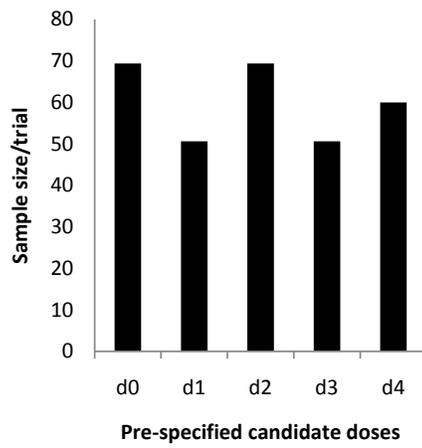
both stages are 300, then the aggregate sample sizes for the pre-specified candidate doses,  $d_0, \dots, d_4$ , are computed as shown in Figure 4b. The resulting sample sizes are not balanced across doses in 10,000 trials (35% variation). This is because 1) ADTAR I (Section 2) does not map the potential results,  $b_1, \dots, b_8$ , to the pre-specified adaptation choices,  $a_1, \dots, a_4$ , in a one-to-one fashion and 2) the number of potential results in Stage 1 that map to the adaptation choices is not the same across the adaptation choices. Furthermore, doses to be used in Stage 2 are not the same across the four adaptation choices. Hence, adjusting the aggregate sample sizes with our proposed weights (Table 3), the sample size imbalance is much improved ( $\leq 5\%$  variation) as shown in Figure 4c. This example illustrates how a missing data mechanism in Stage 2 can result using an ADTAR approach and how our weighting based on a missing data analysis can compensate the resulting imbalance.

### 3.4.4 Controlling the FWER

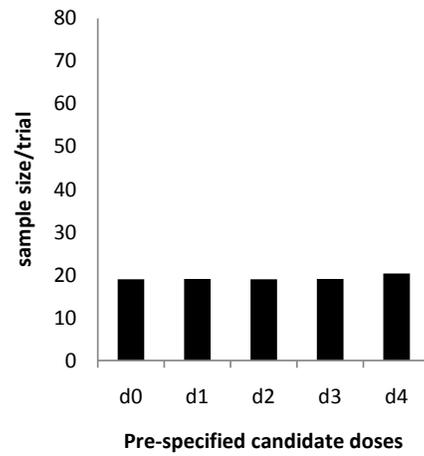
In this section, we evaluate how much a pre-specified ADTAR can control the FWER close to a pre-chosen 0.05 level in a two-stage design (as discussed in Section 3.3). To do so, we compare our ADTAR design with a random two-stage design without ADTAR. We simulated 10,000 trials using a random two-stage design, denoted as *Random Two-Stage*, where all 15 dose combinations are equally possible choices in Stage 2 (Figure 5). Table 6 shows both Stage 2's and global PoA results for four different models (Emax, Logistic, Quadratic, and Constant) and four different total sample sizes ( $N = 80, 200, 400$ , and 600). As shown for the data generated by the constant model, Stage 2's PoA results are as low as 0.031 across different total sample sizes. As a result, the global PoA under the constant model also seems to be lower than the ideal value of 0.05 for all sample sizes. As we can see in this example, without specifying ADTAR in a two-stage design, the FWER can be over-controlled especially with many dose-combination choices in Stage 2.



(a) Stage 1's potential results



(b) Sample sizes without weighting



(c) Sample sizes with weighting

Figure 4: Equal probability of Stage 1's results and sample sizes with and without weighting.

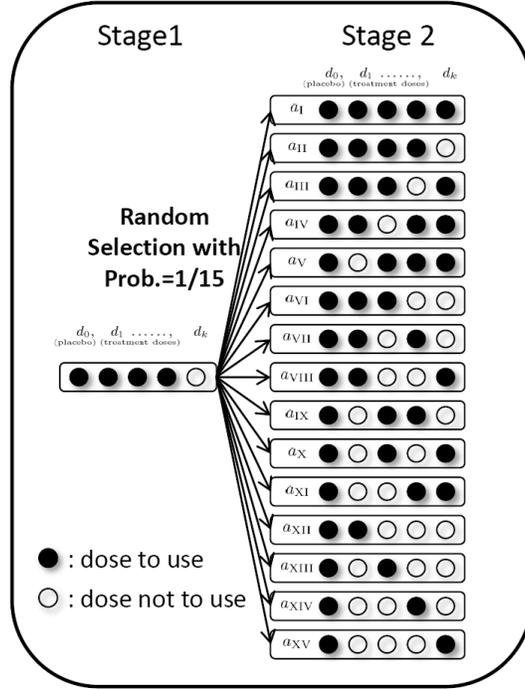


Figure 5: *Random Two-Stage* design

Table 6: PoAs of *Random Two-Stage* design under varying D-R shape  $\times$  sample size

Design Choice	Test Stage	$n_{2i}/gp$ ( $N$ )	Data-Generating Function				$n_{2i}/gp$ ( $N$ )	Data-Generating Function			
			Const.	Emax	Quad.	Logist.		Const.	Emax	Quad.	Logist.
<i>Random Two-Stage</i>	Stage 2	8-20	0.0354	0.1874	0.1337	0.2012	20-50	0.0319	0.3688	0.2673	0.3970
	Global	(80)	0.0438	0.2975	0.2845	0.3098	(200)	0.0391	0.5705	0.5544	0.5937
<i>Random Two-Stage</i>	Stage 2	40-100	0.0372	0.5896	0.4242	0.6060	60-150	0.0359	0.7377	0.5533	0.7090
	Global	(400)	0.0437	0.8347	0.8113	0.8318	(600)	0.0435	0.9295	0.9261	0.9145

### 3.5 DISCUSSION

In this chapter, we presented an extension of the MCP-Mod method by Branson, et al. and Bretz, et al. from a classical clinical trial design to an adaptive two-stage trial design. Among a set of potential doses, several doses and a placebo were used in the first stage. In the second stage, doses were selected (including the placebo) according to our pre-specified dose adaptation rule called ADTAR. In both stages, we used the same set of the dose-response candidate models. Each stage used the model-associated multiple contrast test statistics for the preliminary hypothesis testing on dose-response relationship. We then combined the test results of both stages in order to establish global PoC via a conditional error function.

We focused on one of the three objectives in the original MCP-Mod method; our research goal was to establish global PoC after adaptively adding and/or dropping treatment doses in the second stage based on the result of the first stage. For the second stage, we showed how to model potentially unequal sample sizes and dropped doses across adaptation choices due to ADTAR function using the missing data weighting method.

In the simulation studies, we evaluated the performance of our proposed ADTAR approach for detecting a dose-response curve called Proof of Activity (PoA). The ADTAR approach roughly preserved the FWER at a 5% level. The missing data weighting method in Stage 2 improved PoA within the same ADTAR design. Furthermore, the ADTAR approach showed more a robust performance across the three forms of the true dose-response curves in establishing global PoC compared to the conventional study designs. In particular, for the quadratic dose-response, the ADTAR approach was superior in PoA to the one-stage design of the original MCP-Mod method. Additionally, the ADTAR approach showed adequate power compared to the maximum and minimum power from the four fixed two-stage designs.

## 4.0 CONSTRAINTS OF DOSE CHOICES IN THE MULTIPLE COMPARISON PROCEDURES AND MODELING APPROACH

In this chapter, we describe constraints for choosing doses in using the MCP-Mod and our extended MCP-Mod (Ex-MCP-Mod) methods. We identify the constraints which are imposed by both the estimation form of correlation matrix in an adaptive two-stage design and floating-point arithmetic. We discuss the relationship of these constraints and ADTAR in the Ex-MCP-Mod method as well.

### 4.1 CONSTRAINTS TO DOSE CHOICES IN THE MCP-MOD

To implement the MCP-Mod method or Ex-MCP-Mod method in Sections 2.4 and 3.1, respectively, several pre-specifications are required in the design protocol. A set of dose choices is one such pre-specification. In our previous research, we noted that a set of potential levels of doses in the MCP-Mod method and Ex-MCP-Mod method were constrained. In this section, we further discuss the constraint in choosing a set of doses for the MCP-Mod method.

As seen in Section 2.4, it is assumed that the vector of  $M$  model-associated contrast statistics as in (2.4) of the MCP-Mod method follows a  $M$ -variate  $t$ -distribution under the null hypothesis. To hold this assumption, its numerically computed correlation matrix,  $\mathbf{R} = (\rho_{uv})$ ,  $1 \leq u, v \leq M$ , must be positive definite. Since the matrix elements,  $\rho_{uv}$ , are functions of doses, a set of doses must be chosen to satisfy this requirement.

For simplicity, we consider equal sample sizes for the placebo group and  $k$  dose groups in Section 2.4. Then, correlations  $\rho_{uv}$  are simplified as follows.

$$\begin{aligned}\rho_{uv} &= \frac{\sum_{i=0}^k c_{ui}c_{vi}}{\sqrt{\sum_{i=0}^k c_{ui}^2}\sqrt{\sum_{i=0}^k c_{vi}^2}} \\ &= \frac{\mathbf{c}'_u \mathbf{c}_v}{\|\mathbf{c}_u\| \|\mathbf{c}_v\|}, \quad 1 \leq u, v \leq M.\end{aligned}\tag{4.1}$$

Remember that the optimal contrast vectors,  $\mathbf{c}_m$ ,  $m = 1, \dots, M$ , satisfy the regulatory conditions and maximize the non-centrality parameter of the multivariate  $t$ -distribution. Let  $\boldsymbol{\mu}_m^0 = (\mu_{m0}^0, \dots, \mu_{mk}^0)'$  be the standardized mean vector obtained from the standardized model,  $f_m^0$ , with doses  $d_0, \dots, d_k$ , and let  $\mu_m^0$  be the average of the  $k + 1$  mean values. Then, each optimal contrast vector is expressed with  $\boldsymbol{\mu}_m^0$  and  $\mu_m^0$  as follows [5].

$$\mathbf{c}_m = \frac{\boldsymbol{\mu}_m^0 - \mu_m^0 \mathbf{1}}{\|\boldsymbol{\mu}_m^0 - \mu_m^0 \mathbf{1}\|}, \quad m = 1, \dots, M.\tag{4.2}$$

To illustrate, let us focus on the three standardized models: Emax ( $f_1^0$ ), quadratic ( $f_2^0$ ), and logistic ( $f_3^0$ ) models as shown in Table 1 and Figure 4 in Section 3.4. These models were used in both the MCP-Mod (Bretz, et al., 2005) and Ex-MCP-Mod methods. Again, given a set of candidate dose-response models, dose specification determines the correlations and the model contrast vectors as in (4.1) and (4.2), respectively. Therefore, one can expand the following discussion to the other models of Bretz, et al. [6].

Based on the three standardized models, we compute a  $3 \times 3$  correlation matrix,  $\mathbf{R}_3 \times 3$ .

$$\mathbf{R}_3 \times 3 = \begin{pmatrix} 1 & \rho_{12} & \rho_{13} \\ \rho_{21} & 1 & \rho_{23} \\ \rho_{31} & \rho_{32} & 1 \end{pmatrix} = \begin{pmatrix} 1 & \rho_{12} & \rho_{13} \\ \rho_{12} & 1 & \rho_{23} \\ \rho_{13} & \rho_{23} & 1 \end{pmatrix}\tag{4.3}$$

Since symmetric  $\mathbf{R}_3 \times 3$  must be positive definite, we obtain the following necessary and sufficient conditions from Sylvester's criterion [17].

1. The upper  $1 \times 1$  matrix has a positive determinant.
2. The upper  $2 \times 2$  matrix has a positive determinant, i.e.,  $1 - \rho_{12}^2 > 0$ .

3. The  $3 \times 3$  matrix has a positive determinant, i.e.,  $1 + 2\rho_{12}\rho_{23}\rho_{13} - (\rho_{12}^2 + \rho_{23}^2 + \rho_{13}^2) > 0$ .

Note that the first condition is always met in the computed matrix. The third condition is the main component of the constraint for dose specification in the MCP-Mod method. These three conditions together signify that the correlation elements must be within the three-dimensional shape given by the following equation.

$$\begin{pmatrix} \rho_{23} \\ \rho_{13} \end{pmatrix} \begin{pmatrix} \frac{1}{1-\rho_{12}^2} & \frac{-\rho_{12}}{1-\rho_{12}^2} \\ \frac{-\rho_{12}}{1-\rho_{12}^2} & \frac{1}{1-\rho_{12}^2} \end{pmatrix} \begin{pmatrix} \rho_{23} & \rho_{13} \end{pmatrix} = 1, \quad (4.4)$$

where  $-1 < \rho_{12}, \rho_{23}, \rho_{13} < 1$ . Let  $(\rho_{23}, \rho_{13}, \rho_{12})$  be a point in the  $XYZ$  coordinate system. Performing a coordinate transformation using  $X = \frac{U+V}{\sqrt{2}}$  and  $Y = \frac{-U+V}{\sqrt{2}}$ , we illustrate the three-dimensional shape as Figure 6a. The shape seems similar to a tetrahedron having vertices at  $(1, -1, -1)$ ,  $(-1, 1, -1)$ ,  $(-1, -1, 1)$ , and  $(1, 1, 1)$  in the  $XYZ$ -coordinate. However, the surface cut at a constant plane,  $Z = c$ , has an elliptical shape defined as  $\frac{U^2}{1+c} + \frac{V^2}{1-c} = 1$  (see Figure 6c and 6d). Especially at  $Z = 0$ , the surface is circular defined as  $X^2 + Y^2 = 1$  (Figure 6b). The three-dimensional object within the boundary described by (4.4) is a convex and open set. Its volume is

$$\int_{x^2+y^2+z^2-2xyz < 1} dx dy dz = \frac{\pi^2}{2},$$

i.e., it is greater than half of the circumscribing cube. Accordingly, we expect that many dose combinations will allow the computed correlation matrix,  $\mathbf{R}_3 \times 3$ , to be numerically positive definite.

In the Ex-MCP-Mod method, recall that we use weighted contrast test statistics in the second stage of a two-stage adaptive design. Using such weights, the correlation matrix is also weighted. Suppose that an ADTAR function defines  $L$  adaptation choices in the second stage. Let  $\omega_{ij}$  denote the weight which is used to adjust the mean estimate of dose group  $i$  for a contrast test statistic given adaptation choice  $a_j \in \{a_1, \dots, a_L\}$ . The weight formula is derived based on the ADTAR function and  $\sum_{j=1}^L \omega_{ij} = 1$  (see Section 3.2). Assuming equal

sample sizes within the selected adaptation which uses the placebo and  $k_2$  dose groups, the weighted correlation formula in the second stage is shown as follows.

$$\rho_{uv,2} = \frac{\sum_{i=0}^{k_2} \omega_{ij}^2 c_{ui} c_{vi}}{\sqrt{\sum_{i=0}^{k_2} \omega_{ij}^2 c_{ui}^2} \sqrt{\sum_{i=0}^{k_2} \omega_{ij}^2 c_{vi}^2}}, \quad 1 \leq u, v \leq M. \quad (4.5)$$

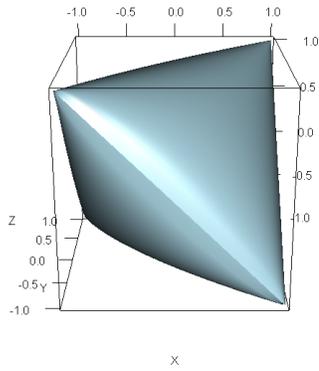
If we use the same set of the candidate dose-response models in Table 1, then the weighted correlation matrix of the second stage is  $\mathbf{R}_2 \ 3 \times 3 = (\rho_{uv})_2$ ,  $1 \leq u, v \leq 3$ . To construct the valid Ex-MCP-Mod method,  $\mathbf{R}_2 \ 3 \times 3$  must also be numerically positive definite. For this reason, in the next section, our simulation study investigates positive definiteness of both un-weighted and weighted matrices for the MCP-Mod method and the Ex-MCP-Mod method, respectively.

## 4.2 SIMULATION STUDY

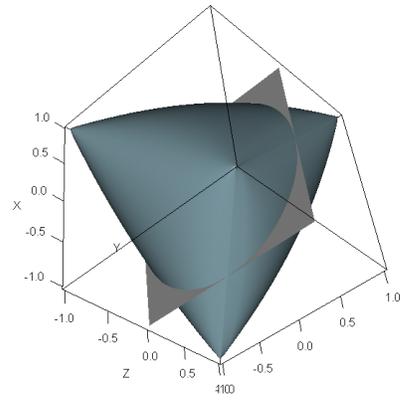
In this section, we show how numerical positive definiteness of a computed correlation matrix depends on dose choices in the MCP-Mod method via a simulation study. We investigate appropriate dose combinations by level plots on the smallest eigenvalues and which dose combinations allow for a computed correlation matrix to be numerically positive definite. For the Ex-MCP-Mod method, we investigate the effect of adaptive weighting of ADTARs on the positive definiteness of the un-weighted matrix.

### 4.2.1 Simulation Method

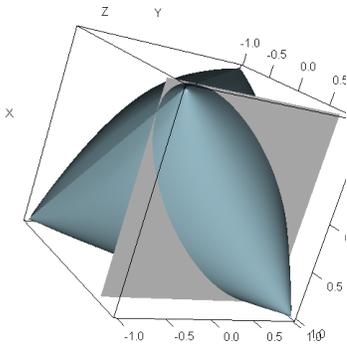
The MCP-Mod method and Ex-MCP-Mod method require pre-specifications on both dose levels and candidate dose-response models in a design protocol. In our simulation study, we assume the same specification of Section 4.1 with respect to candidate dose-response models as  $f_1$ ,  $f_2$ , and  $f_3$  (Table 1 and Figure 4) in the MCP-Mod/Ex-MCP-Mod methods. Given such model specifications, we search for the appropriate number of doses and the appropriate



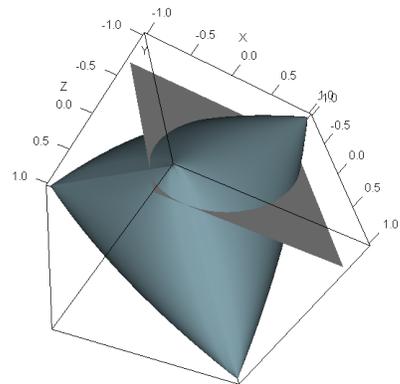
(a) 3D-Shape



(b) 3D-Shape cut at  $Z = 0$



(c) 3D-Shape cut at  $Z = 0.7$



(d) 3D-Shape cut at  $Z = -0.7$

Figure 6: Three-Dimensional Shape of Correlation Elements.

levels of doses from the dose range,  $[0, 1.0]$ . The dose range was chosen to be able to cover both the dose of the minimum response effect and the dose of the maximum response effect on each standardized dose-response model. The simulation study was implemented using the R package (Version 2.10.1). The R programs for implementing the simulation study are available in Appendix A.4. Since the R system is implemented using IEEE double standard precision numbers [23], results of computations between  $-10^{-16}$  and  $10^{-16}$  cannot be distinguished from 0. Therefore, we consider any numbers in this interval to be 0 when establishing positive definiteness of evaluate values.

#### 4.2.2 Optimal Number of Doses

First, to find the appropriate number of doses, we simulate all possible dose combinations from dose range  $[0, 1.0]$  using a placebo group ( $d_0 = 0$ ) and the respective two dose groups ( $k = 2$ :  $d_1$  and  $d_2$ ), three dose groups ( $k = 3$ :  $d_1$ ,  $d_2$  and  $d_3$ ), and four dose groups ( $k = 4$ :  $d_1$ ,  $d_2$ ,  $d_3$  and  $d_4$ ). An equal interval size between doses is fixed as 0.20, 0.10, 0.07, or 0.05. The associated numbers of potential dose choices within the dose range are 5, 10, 15, and 20, respectively. The smallest eigenvalues of the correlation matrices are computed for all dose combinations based on (4.1) and (4.2).

Employing equivalent conditions for positive definiteness of a matrix, a correlation matrix is positive definite if all eigenvalues are positive real numbers [26]. The correlation matrix in (4.3) is symmetric, and we know that all eigenvalues of a symmetric matrix are real numbers. Therefore, to establish positive definiteness of a correlation matrix, it is sufficient to prove that the smallest eigenvalue is positive. For each set of doses, we evaluate numerical positive definiteness of computed matrices by computing the minimum and maximum of the smallest eigenvalues of the correlation matrices. If a range of the smallest eigenvalues covers positive real values, the corresponding number of doses is an appropriate candidate as the number of doses in the MCP-Mod method.

Table 7: The Minimum and Maximum of The Smallest Eivgenvalues of The Correlation Matrices For Different Number of Doses Simulated From Equal Dose Intervals in  $[0,1]$

a set of doses	number of potential doses in $[0,1]$	number of potential doses in $[0,1]$			
		5	10	15	20
$d_0, d_1$ ( $k = 1$ )	minimum	$-2.44 \times 10^{-17}$	$-9.74 \times 10^{-17}$	$-9.74 \times 10^{-17}$	$-9.74 \times 10^{-17}$
	maximum	$1.37 \times 10^{-16}$	$1.59 \times 10^{-16}$	$3.59 \times 10^{-16}$	$1.59 \times 10^{-16}$
$d_0, d_1, d_2$ ( $k = 2$ )	minimum	$-1.12 \times 10^{-16}$	$-4.88 \times 10^{-16}$	$-5.35 \times 10^{-16}$	$-5.86 \times 10^{-16}$
	maximum	$1.59 \times 10^{-16}$	$5.01 \times 10^{-16}$	$5.19 \times 10^{-16}$	$5.57 \times 10^{-16}$
$d_0, d_1, d_2, d_3$ ( $k = 3$ )	minimum	$2.30 \times 10^{-3}$	$6.94 \times 10^{-5}$	$1.06 \times 10^{-5}$	$2.97 \times 10^{-6}$
	maximum	0.0104	0.0150	0.0159	0.0159
$d_0, d_1, d_2, d_3, d_4$ ( $k = 4$ )	minimum	0.0125	$4.42 \times 10^{-4}$	$6.57 \times 10^{-5}$	$1.80 \times 10^{-5}$
	maximum	0.0125	0.0163	0.0176	0.0179

Note: The sample size,  $n_i = 20$ , was used for each group. Minimum and maximum refer to the smallest eigenvalues.

Table 7 shows the computed ranges of the smallest eigenvalues for different numbers of treatment doses,  $k = 1, 2, 3$ , and 4, and different numbers of potential dose choices in  $[0, 1.0]$ . In this table, when the number of treatment doses is  $k = 1$  or  $k = 2$ , the computed correlation matrices are not numerically positive definite. When the number of treatment doses is  $k = 3$  or  $k = 4$ , the computed correlation matrices are numerically positive definite. Figures 7 and 8 show the range determined by the minimum and maximum log 10 of the smallest eigenvalues of the computed correlation matrix for  $k = 3$  and  $k = 4$ , respectively. From the result, we see that at least more than two treatment groups need to be specified in the MCP-Mod method. As a necessary condition, the same condition must hold if one plans to use the Ex-MCP-Mod employing an ADTAR in an adaptive two-stage design.

### 4.2.3 Optimal Levels of Doses

Next, to find appropriate levels of doses, we simulate sets of the placebo group ( $d_0 = 0$ ) and the respective number of dose groups ( $k = 1, 2, 3$  and 4) in increasing order of doses, from  $d_1$  to  $d_4$ , using various equal intervals:  $10^{-1}, 10^{-2}, \dots, 10^{-10}$ . For example, we evaluate doses  $(0, 0.01, 0.02, 0.03)$  for  $10^{-2}$  intervals with  $k = 3$ . For each set of doses, we evaluate numerical positive definiteness of the computed correlation matrices by computing the minimum of the smallest eigenvalues against each equal dose interval.

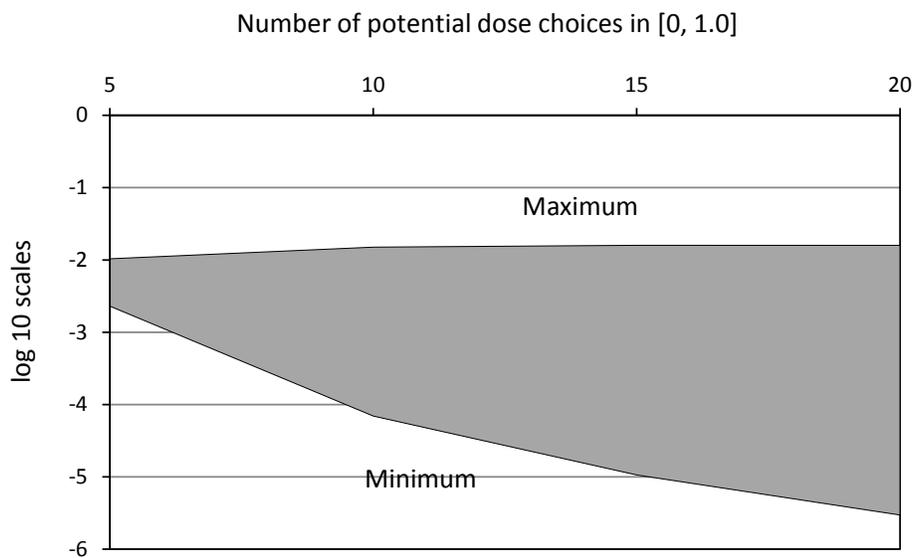


Figure 7: The Minimum and Maximum Log 10 of the Smallest Eigenvalues of  $\mathbf{R}_3 \times 3$  for a Placebo and Three Treatment Doses ( $d_1$ ,  $d_2$ , and  $d_3$ ).

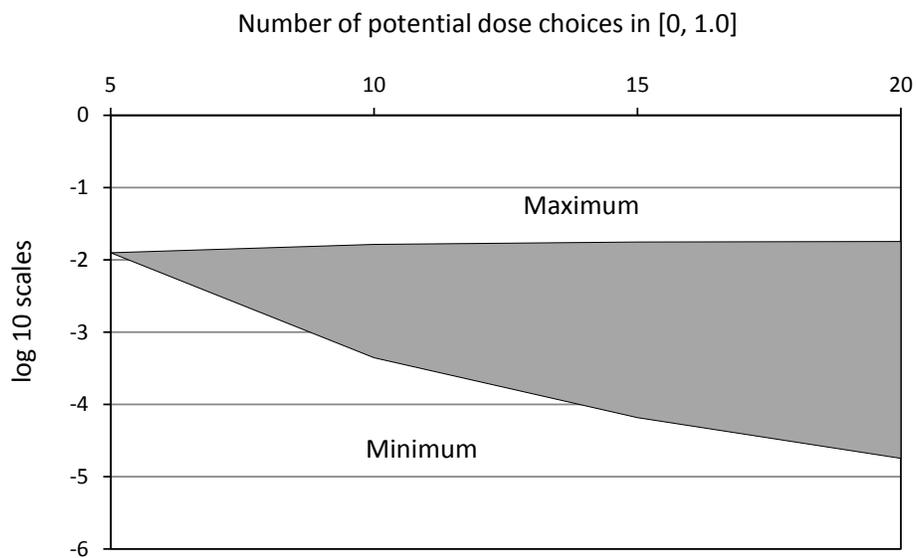


Figure 8: The Minimum and Maximum Log 10 of the Smallest Eigenvalues of  $\mathbf{R}_3 \times 3$  for a Placebo and Four Treatment Doses ( $d_1$ ,  $d_2$ ,  $d_3$ , and  $d_4$ ).

Table 8: The Minimum of The Smallest Eivgenvalues of The Correlation Matrices Against Various Dose Intervals For Different Number of Doses

equal dose intervals	$10^{-1}$	$10^{-2}$	$10^{-3}$	$10^{-4}$	$10^{-5}$
(a set of doses)					
$d_0, d_1$	$8.65 \times 10^{-17}$	$-2.44 \times 10^{-17}$	$-9.32 \times 10^{-18}$	$1.59 \times 10^{-16}$	$1.37 \times 10^{-16}$
$d_0, d_1, d_2$	$2.32 \times 10^{-16}$	$1.14 \times 10^{-16}$	$-1.70 \times 10^{-16}$	$1.45 \times 10^{-16}$	$-1.44 \times 10^{-16}$
$d_0, d_1, d_2, d_3$	$8.41 \times 10^{-5}$	$2.07 \times 10^{-8}$	$2.35 \times 10^{-12}$	$4.72 \times 10^{-16}$	$3.76 \times 10^{-16}$
$d_0, d_1, d_2, d_3, d_4$	$3.06 \times 10^{-4}$	$7.93 \times 10^{-8}$	$9.35 \times 10^{-12}$	$1.11 \times 10^{-15}$	$5.24 \times 10^{-17}$
equal dose intervals	$10^{-6}$	$10^{-7}$	$10^{-8}$	$10^{-9}$	$10^{-10}$
(a set of doses)					
$d_0, d_1$	$1.59 \times 10^{-16}$	$8.65 \times 10^{-17}$	$1.25 \times 10^{-16}$	$-2.44 \times 10^{-17}$	$4.57 \times 10^{-17}$
$d_0, d_1, d_2$	$-9.15 \times 10^{-17}$	$3.17 \times 10^{-16}$	$8.65 \times 10^{-17}$	$-3.48 \times 10^{-16}$	$3.95 \times 10^{-16}$
$d_0, d_1, d_2, d_3$	$-1.41 \times 10^{-16}$	$5.43 \times 10^{-18}$	$8.65 \times 10^{-17}$	$-1.12 \times 10^{-16}$	$-2.68 \times 10^{-16}$
$d_0, d_1, d_2, d_3, d_4$	$1.08 \times 10^{-16}$	$3.34 \times 10^{-16}$	$-2.69 \times 10^{-16}$	$5.19 \times 10^{-16}$	$2.12 \times 10^{-16}$

Table 8 shows the computed minimum values of the smallest eigenvalues. Figure 9 shows the log 10 plots of the computed minimum values of the smallest eigenvalues. In this figure, when the number of dose groups is  $k = 1$  or  $k = 2$ , the computed correlation matrices are not numerically positive definite for any of the evaluated dose intervals. When the number of dose groups is  $k = 3$  or  $k = 4$ , the computed correlation matrices are numerically positive definite. However, they numerically approach the indefinite as the dose interval gets smaller. From this result, the equal interval between doses needs to be greater than or equal to  $10^{-3}$  for  $k \geq 3$  in the MCP-Mod method as it requires a numerically positive definite correlation matrix. In our situation, for two doses and placebo, the MCP-Mod can never be used.

#### 4.2.4 The Best Configurations of Doses

To show the best configurations of levels of doses in the MCP-Mod method, we simulate all possible sets of the placebo and three dose groups (as ordered from  $d_1$  to  $d_3$ ) with an equal dose interval, 0.05, in the range of  $[0, 1.0]$ . The relevant number of potential doses in  $[0, 1.0]$  is 20. For each fixed value of the largest dose group,  $d_3$ , we create a level plot of the smallest eigenvalues associated with all possible combinations of the first and second dose groups,  $d_1$  and  $d_2$ . In such a level plot, the best configurations of dose levels can be presented by relatively large positive real values.

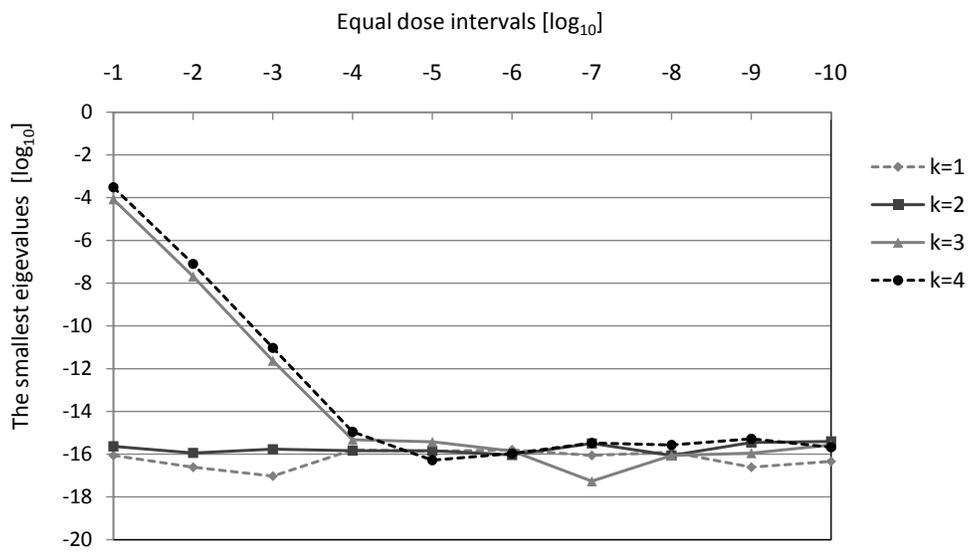


Figure 9: Log 10 of The Smallest Eigenvalues of  $\mathbf{R}_3 \times 3$  for Varying Dose Intervals in  $[0, 1.0]$ .

Figures 10 and 11 show the log 10 level plots of the smallest eigenvalues for all possible dose combinations between  $d_1$  and  $d_2$  at the descending fixed values of  $d_3$ : 1.00, 0.95, 0.90, ..., 0.15. For example, in Figure 10-(a)  $(d_1, d_2)=(0.20, 0.65)$  is one of the best configurations between the rest of the two dose groups with  $d_3 = 1.00$ . Overall, the best configurations seem to be established when dose intervals are relatively large and when levels of doses are spread out enough over the range of  $[0, 1.0]$ .

#### 4.2.5 The Weighting Effect on Positive Definiteness

As a result of Sections 4.2.2 and 4.2.3, if the number of treatment groups is greater than or equal to three and if equal dose intervals are greater than or equal to  $10^{-3}$ , the computed correlation matrix in the MCP-Mod is numerically positive definite. Accordingly, these are the necessary conditions for holding the Ex-MCP-Mod method. In order to numerically investigate if the two conditions are also sufficient for the Ex-MCP-Mod method, i.e., if weighting in the second stage of the Ex-MCP-Mod does not change the numerical positive definiteness of the computed un-weighted matrix, we simulate all possible ADTAR functions where the number of treatment groups in the second stage ( $k_2$ ) is either three or four ( $k_2 \geq 3$ ) out of four potential doses. Equal dose intervals are set to be 0.20, 0.10, or 0.05. The associated numbers of potential dose choices within the dose range are 5, 10, and 20, respectively. Then, we compute the weighted matrices in (4.5) and the minimum of the smallest eigenvalues from all the ADTARs for each set of doses. In the respect cases of  $k_2 \in \{3, 4\}$  and  $k_2 \in \{2, 3, 4\}$ , if the lower bound of the minimum eigenvalues is positive for all the simulated sets of doses, then the corresponding  $k_2$  is considered to be one sufficient condition to construct a valid ADTAR in the Ex-MCP-Mod method. Our extensive simulation showed that the lower bound of the minimum eigenvalues was greater than 0 ( $> 10^{-6}$ ) for simulating  $k_2 \in \{3, 4\}$  but not numerically positive ( $< 10^{-15}$ ) for simulating  $k_2 \in \{2, 3, 4\}$ . Consequently, in an ADTAR function at least three treatment groups must be used in each adaptation choice. This simulation result does not provide any additional condition to the necessary conditions

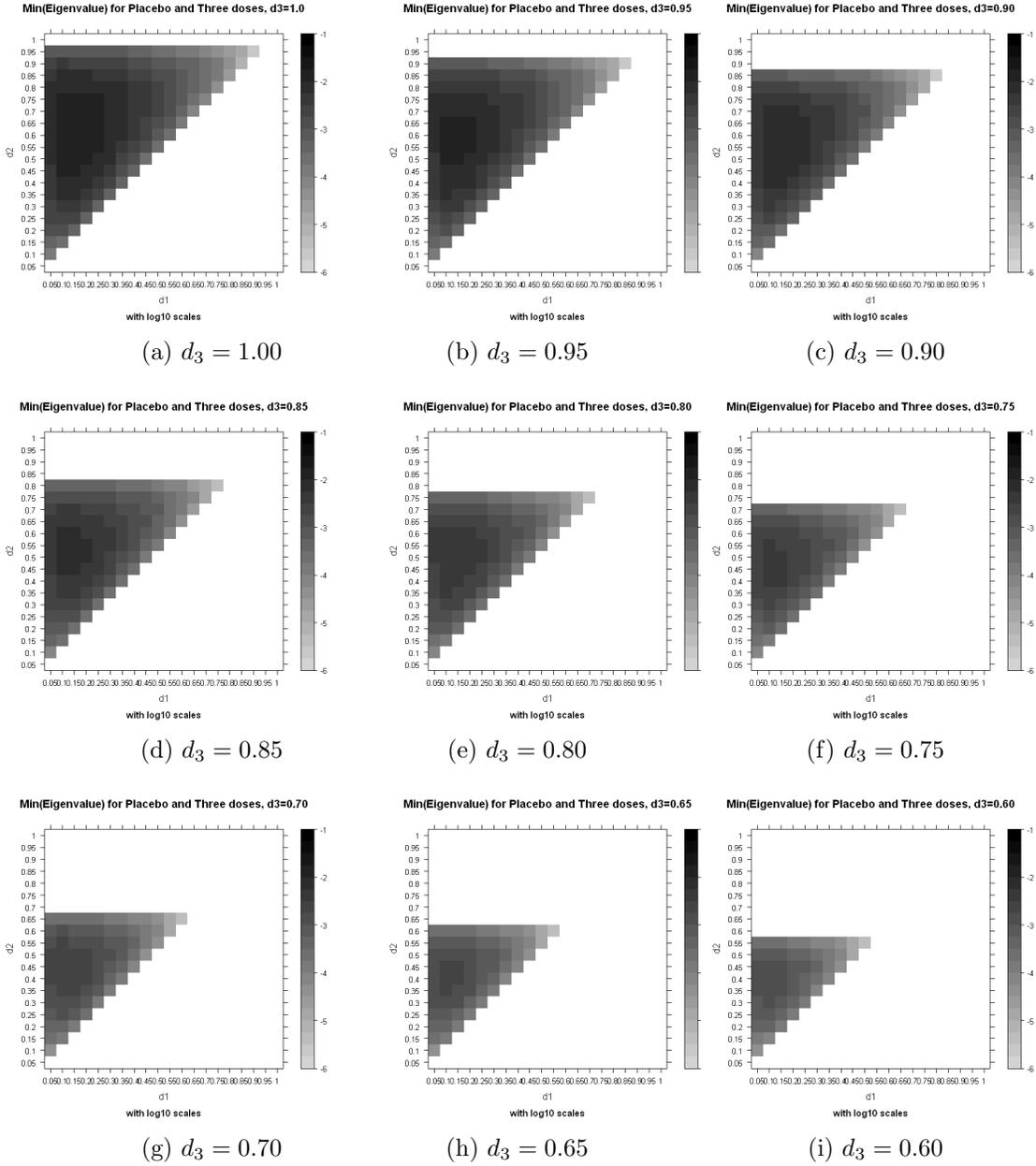


Figure 10: Log 10 of the Smallest Eigenvalues of  $\mathbf{R}_3 \times 3$  for a Placebo and Three Treatment Doses ( $d_1, d_2$ , and  $d_3$ ).

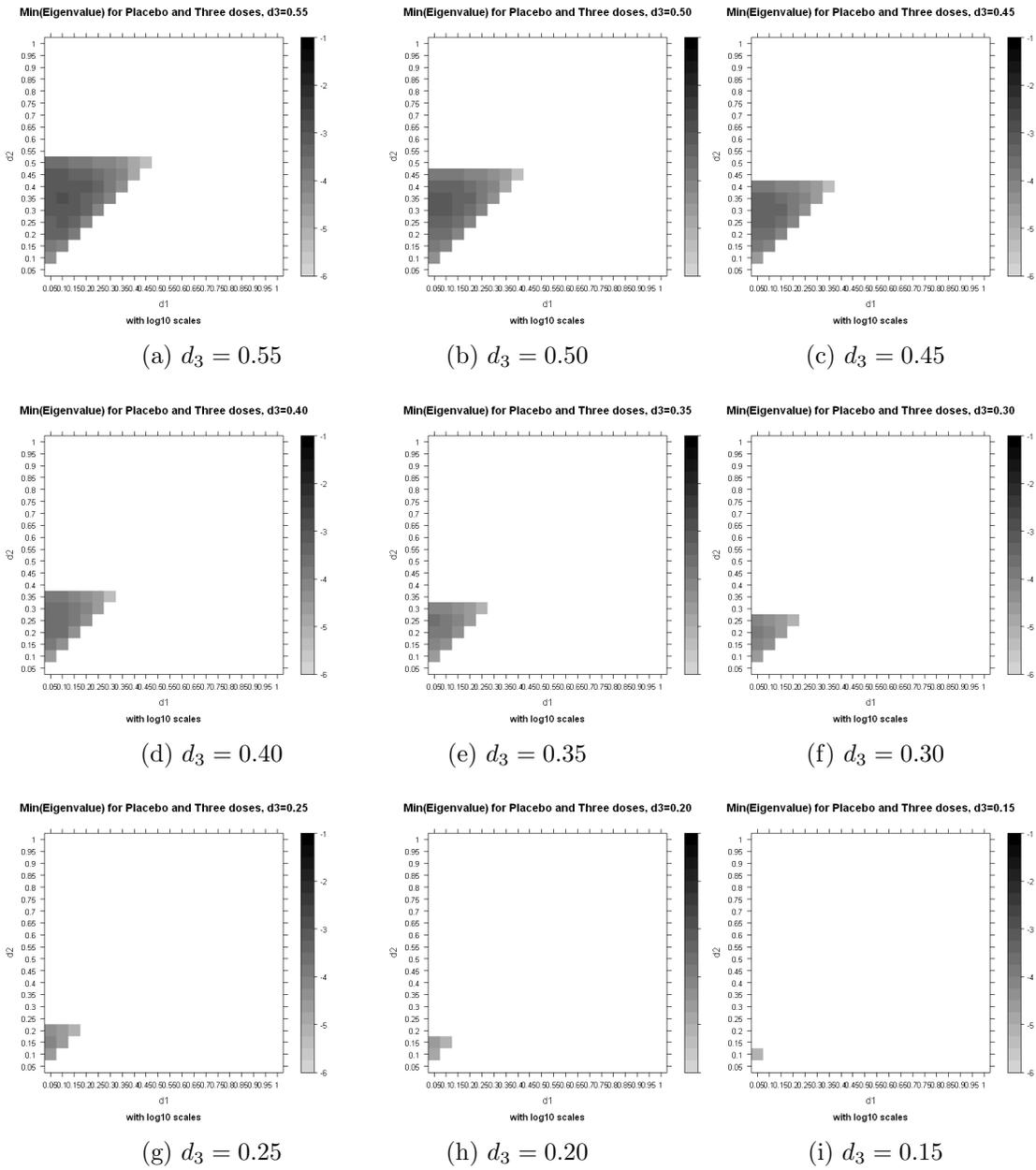


Figure 11: Log 10 of the Smallest Eigenvalues of  $\mathbf{R}_3 \times 3$  for a Placebo and Three Treatment Doses ( $d_1, d_2$ , and  $d_3$ ).

in the Ex-MCP-Mod method. Therefore, we conclude that weighting based on an ADTAR function does not change the numerical positive definiteness of the computed un-weighted matrix. The necessary and sufficient conditions for the MCP-Mod method remain the same for the Ex-MCP-Mod method.

### 4.3 DISCUSSION

Our analysis and experiments showed that given the three candidate dose-response models, the Emax, quadratic, and logistic models, the MCP-Mod method is stable if the number of treatment groups does not fall below three and if equal dose intervals are greater than or equal to  $10^{-3}$  in dose range  $[0, 1.0]$ . In the Ex-MCP-Mod method, ADTAR functions and derived weights do not affect numerical positive definiteness of the computed un-weighted correlation matrices; the Ex-MCP-Mod method using an ADTAR function is valid if the same conditions as above hold. When formulating an ADTAR function, it seems to be an unavoidable constraint that at least three treatment groups must be used in the second stage of a two-stage design. However, from the practical stand point, this constraint would not be so critical, because according to the second condition the levels of doses are allowed to be relatively close. Thus, one can pre-specify flexible adaptations using an ADTAR function without selecting dosage levels that are too high and toxic for the second stage.

If in the simulation study, Emax, quadratic, and logistic models are pre-specified as the candidate dose-response models in the MCP-Mod, it is preferable to distribute potential doses evenly between placebo and the highest dose chosen for the trial. Our algorithm to search for appropriate dose candidates is applicable to other candidate dose-response models for the MCP-Mod/Ex-MCP-Mod methods. Level plots are especially useful to visualize the appropriate dose combinations. Our findings and proposed algorithm should aid the design of a dose-response study when one wishes to employ the MCP-Mod/Ex-MCP-Mod methods.

## 5.0 CONCLUSION

In this dissertation, we proposed to generalize one previously developed method for two-stage adaptive designs to accommodate situations where both adding and dropping treatment arms are possible between stages. Our procedure was based on using the data set in its entirety and would control for the familywise error rates. Specifically, we proposed

1. to develop an efficient two-stage study design where an adaptation rule adding and/or dropping treatment arms, which we call ADTAR, is pre-specified;
2. to use our adaptive design to establish evidence of a global dose-response relationship or Proof of Concept (PoC) for a treatment; and
3. to show that the two-stage design with the pre-specified adaptation rule can be used to improve statistical power in establishing global PoC.

In Chapter 3, we introduced a pre-specified adaptation rule that adds and/or drops treatment arms between stages, and we called this rule ADTAR. We aimed to follow the FDA's draft guidance on adaptive designs, which advises that clinical experimenters detail adaptation rules in protocols as much as possible. For establishing PoC in drug development, we showed that using an ADTAR function in adaptive two-stage designs adds flexibility over conventional designs and fixed two-stage designs. In addition, the pre-specification of adaptation rules allowed us to increase power and prevents over-controlling the FWER by limiting the adaptation choices compared to two-stage designs without pre-specified rules. In our simulation studies, we showed that redistributing the sample sizes of the unused arms to the retained arms in Stage 2 could allow the ADTAR approach to preserve and potentially

increase power in detecting evidence of dose-response from fixed two-stage designs of the same total sample sizes.

In Chapter 4, practical considerations for dose pre-specification for Bretz et al.'s MCP-Mod method and our Ex-MCP-Mod method were described. In particular, we established lower bounds for the number and levels of doses for each method in a simulation study. These constraints were imposed by both the formula of the correlation matrix and floating-point arithmetic. We showed that as long as the computed un-weighted correlation matrices were numerically positive definite, the Ex-MCP-Mod method did not impose any further constraints on the adaptive two-stage design with the same dose pre-specification. Our findings and proposed algorithm should aid the design of a dose-response study when one wishes to employ the MCP-Mod/Ex-MCP-Mod methods.

In summary, we developed a novel type of pre-specified adaptation rule, ADTAR, for use in adaptive two-stage designs in order to comply with the recommendation in the FDA's draft guidance. Using simulation studies, we showed that

1. the FWER is controlled in establishing global PoC in our design;
2. in our design, statistical power for PoC tests is robust and potentially superior to conventional and fixed two-stage designs; and
3. there are constraints for pre-specifying the number and levels of doses in clinical trials when we use the MCP-Mod method. Our Ex-MCP-Mod method does not impose further constraints over the original MCP-Mod method.

The results of this thesis suggest further investigation of 1) identifying the best candidate dose-response model, and 2) determining a target treatment dose in the proposed design method. Due to logistic considerations in drug development, effective sample allocation and the relationship between sample allocation and PoA are also topics for further investigation. Finally, one open question remains about how to find the optimal ADTAR, which depends on previously obtained pre-clinical and clinical data (including toxicity and efficacy) for the respective investigational drug.

## APPENDIX

### R PROGRAMS FOR SIMULATION STUDIES

#### A.1 PROOF OF ACTIVITY (POA) OF *ADTAR I* (CHAPTER 3)

```
library(mvtnorm)

#####
### Note: This program performs Establishment of PoC.
###     Equal sample size per dose group in Stage 1 is set to be 75.
###     This number must be replaced to 10, 25, or 50 for the other simulations.
###     Equal sample sizes in Stage 2 must be replaced accordingly.
###     All PoC tests are one-sided.
###     Used a critical value from MVT for individual tests.
###     Non-weighting in Stage 2 is also evaluated.

### Function to get rid of NAs from a matrix for Stage 2.
fun = function(MATRIX) {
  LOG = is.na(MATRIX)
  if(sum(LOG) == 0) {
    new = MATRIX
  }
  else {
    P = nrow(MATRIX);
    Q = ncol(MATRIX)
    NA.row = rep(NA,P);
    NA.col = rep(NA,Q)
    for (i in 1:P) {
      if (sum(LOG[i,]) == Q) {
        NA.row[i] = -i
      }
    }
    for (j in 1:Q) {
      if (sum(LOG[,j]) == P) {
        NA.col[j] = -j}
    }
  }
}
```

```

        new = MATRIX[na.exclude(NA.row), na.exclude(NA.col)]
    }
    return(new)
}

### Function to get rid of all-zero rows.
fun2 = function(MATRIX) {
  P = nrow(MATRIX);
  NA.row = rep(NA,P);
  for (i in 1:P) {
    if (sum(MATRIX[i,]) == 0) {
      NA.row[i] = -i
    }
  }
  if(length(na.exclude(NA.row)) >= 1) {
    new = MATRIX[na.exclude(NA.row), ]
  }
  else {
    new=MATRIX
  }
  return(new)
}

### Function to compute a component of weight matrix.
W_c <- function(x,y) {
  W_g<-(adpt.a[x]) * tZ[,x,drop=F]%%J[y,]
  return(W_g)
}

#####
# I. planning a trial      #
#####

# Number of simulated trials.
L <-100

# number of the candidate D-R models.
m<-3

# number of adaptation choices in Stage 2.
adpt.L<-4

# (adpt.L choices x k+1 doses) design matrix for dose adaptation in Stage 2.
Z<-matrix(c(1,1,1,1,0,1,1,1,1,1,1,1,1,0,1,1,1,1,0,1),nrow=adpt.L)
tZ<-t(Z)

# Matrix for associating PoC results in Stage 1 with adaptations in Stage 2.
J<-matrix(c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1),ncol=m)

# g function: g* is adaptation number out of adpt.L choices.
# adpt.a = the number of preliminary PoC results associated with the selected adaptation.

```

```

adpt.a<-c(rep(0,adpt.L))
for(z in 1:8){
  if (z <= 2){
    adpt.a[1]=2
  }
  if (z == 4|z == 6){
    adpt.a[2]=2
  }
  if (z == 7){
    adpt.a[3]=1
  }
  if (z == 3|z == 5|z == 8){
    adpt.a[4]=3
  }
}

# -- weight specification. [(k+1) x m] matrix.
W.denom<-W_c(1,1)+W_c(2,4)+W_c(3,7)+W_c(4,3)

W_g1<-W_c(1,1)*(1/W.denom)
  W_g1<-fun2(W_g1)
W_g2<-W_c(2,4)*(1/W.denom)
  W_g2<-fun2(W_g2)
W_g3<-W_c(3,7)*(1/W.denom)
  W_g3<-fun2(W_g3)
W_g4<-W_c(4,3)*(1/W.denom)
  W_g4<-fun2(W_g4)

### Model Specification, Standardizing Models, and Empty Matrices for Evaluation ###

## 1. Obtain optimal regression coefficients (c_opt). ##

# maximize the power of the test == non-centrality parameter with regulatory condition.
# Using ragranges, we solve the formula and obtain c_opt vector.

# Maximum effect size.
delta.max<-0.6

# Dose to produce the half of the maximum effect.
ED50<-0.2
alpha<-0.025

compute_emax_c1_opt_m1 = function(kandP, n_stg, doses_stg) {
  # Identity matrix of the number of dose size.
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  # Identity matrix of the number of patients/group size.
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### I. Standardized Emax Model.

  # First, write the standadized Emax model.
  f_m1_0<-function(d){
    d/(ED50+d)
  }
}

```

```

}
mu_m1.s1<-matrix(rep(0,kandP),ncol=1)
for(k in 1:kandP){
  d=doses_stg[k]
  mu_m1.s1[k]=f_m1_0(d)
}

# Compute c1_opt for model I in Stage 1.
# Define functions.
mubar.m.s1<-function(mu){
  (sum(mu)/kandP)*Ident1.s1
}
mudiff.m.s1<-function(mu,mubar.m.s1){
  mu-mubar.m.s1
}
c1_opt_m<-function(mudiff.m.s1){
  mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1)%*%mudiff.m.s1)))
}
mubar.m1.s1<-mubar.m.s1(mu_m1.s1)
mudiff.m1.s1<-mudiff.m.s1(mu_m1.s1,mubar.m1.s1)
c1_opt_m1<-c1_opt_m(mudiff.m1.s1)
return(c1_opt_m1)
}

compute_quad_c1_opt_m2 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### II. Standardized quadratic Model.

  # The standadized quadratic model.
  f_m2_0<-function(d){
    d-delta.max*d^2
  }
  mu_m2.s1<-rep(0,kandP)
  for(k in 1:kandP){
    d=doses_stg[k]
    mu_m2.s1[k]=f_m2_0(d)
  }

  mubar.m.s1<-function(mu){
    (sum(mu)/kandP)*Ident1.s1
  }
  mudiff.m.s1<-function(mu,mubar.m.s1){
    mu-mubar.m.s1
  }
  c1_opt_m<-function(mudiff.m.s1){
    mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1)%*%mudiff.m.s1)))
  }

  # Compute c1_opt for model II in Stage 1.
  mubar.m2.s1<-mubar.m.s1(mu_m2.s1)
  mudiff.m2.s1<-mudiff.m.s1(mu_m2.s1,mubar.m2.s1)
}

```

```

c1_opt_m2<-c1_opt_m(mudiff.m2.s1)

return(c1_opt_m2)
}

compute_logis_c1_opt_m3 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### III. Standardized Logistic Model.

  # The standardized logistic model.
  f_m3_0<-function(d){
    1/(1+exp((ED50-d)/delta.max))
  }

  mu_m3.s1<-rep(0,kandP)
  for(k in 1:kandP){
    d=doses_stg[k]
    mu_m3.s1[k]=f_m3_0(d)
  }

  mubar.m.s1<-function(mu){
    (sum(mu)/kandP)*Ident1.s1
  }
  mudiff.m.s1<-function(mu,mubar.m.s1){
    mu-mubar.m.s1
  }
  c1_opt_m<-function(mudiff.m.s1){
    mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1))*%*%mudiff.m.s1)))
  }

  # Compute c_ppt for model III in Stage 1.
  mubar.m3.s1<-mubar.m.s1(mu_m3.s1)
  mudiff.m3.s1<-mudiff.m.s1(mu_m3.s1,mubar.m3.s1)
  c1_opt_m3<-c1_opt_m(mudiff.m3.s1)
  return(c1_opt_m3)
}

compute_cr1 = function(doses_stg, samples){
  # k + 1 in Stage 1 (# of treatment doses and the placebo).
  k1andP=length(doses_stg)
  n_stg1<-replicate(k1andP, samples)
  N1=sum(n_stg1)
  c1_opt_m1<-compute_emax_c1_opt_m1(k1andP, n_stg1, doses_stg)
  c1_opt_m2<-compute_quad_c1_opt_m2(k1andP, n_stg1, doses_stg)
  c1_opt_m3<-compute_logis_c1_opt_m3(k1andP, n_stg1, doses_stg)
  c1_opt_stg1<-t(matrix(c(c1_opt_m1, c1_opt_m2, c1_opt_m3),nrow=k1andP))

  ## 2. Find the critical value for PoC test H0 ###
  # Refer to 'UsingMvtPackage.pdf'.
  # The decision rule: use the best contrasts for computing the test statistic (Tmax).

```

```

# compute the critical value for the H0 test.
# Note: the linear contrast matrix under H0 is C1.
V1<-diag(1/n_stg1)
df1<-N1-k1andP
# [m x (k+1)] matrix.
C1<-matrix(c1_opt_stg1, nrow=m)
cv1<-C1%*%V1%*%t(C1)
dv1<-t(1/sqrt(diag(cv1)))
# this is the correlation matrix under H0.
cr1<-cv1*(t(dv1)%*%dv1)
return(cr1)
}

compute_cr2 = function(doses_stg, samples, W_a){
  # k + 1 in Stage 2 (# of treatment doses and the placebo).
  k2andP=length(doses_stg)
  n_stg2<-replicate(k2andP, samples)
  N2=sum(n_stg2)
  c2_opt_m1<-compute_emax_c1_opt_m1(k2andP, n_stg2, doses_stg)
  c2_opt_m2<-compute_quad_c1_opt_m2(k2andP, n_stg2, doses_stg)
  c2_opt_m3<-compute_logis_c1_opt_m3(k2andP, n_stg2, doses_stg)
  Wc2_opt_m1=W_a[,1]*c2_opt_m1
  Wc2_opt_m2=W_a[,2]*c2_opt_m2
  Wc2_opt_m3=W_a[,3]*c2_opt_m3
  c2_opt_stg2<-t(matrix(c(Wc2_opt_m1, Wc2_opt_m2, Wc2_opt_m3),nrow=k2andP))

  ## 2. Find the critical value for PoC test H0 ###
  V2<-diag(1/n_stg2)
  df2<-N2-k2andP
  C2<-matrix(c2_opt_stg2, nrow=m)
  cv2<-C2%*%V2%*%t(C2)
  dv2<-t(1/sqrt(diag(cv2)))
  cr2<-cv2*(t(dv2)%*%dv2)
  return(cr2)
}

### Common Setting ###
d0<-0
d1<-0.05
d2<-0.2
d3<-0.6
d4<-1.0

### Stage 1 setting ###

n.s1<-75
doses_stg1<-c(d0,d1,d2,d3)

k1andP=length(doses_stg1)
n_stg1=c(rep(n.s1,k1andP))
N1<-sum(n_stg1)
df1<-N1-k1andP
cr1<-compute_cr1(doses_stg1, n.s1)

```

```

c1_opt_m1<-compute_emax_c1_opt_m1(k1andP, n_stg1, doses_stg1)
c1_opt_m2<-compute_quad_c1_opt_m2(k1andP, n_stg1, doses_stg1)
c1_opt_m3<-compute_logis_c1_opt_m3(k1andP, n_stg1, doses_stg1)
c1_opt_stg1<-t(matrix(c(c1_opt_m1, c1_opt_m2, c1_opt_m3),nrow=k1andP))
Ident1.s1<-matrix(rep(1,k1andP),ncol=1)
Ident2.s1<-matrix(rep(1,n_stg1[1]),ncol=1)

### Stage 2 setting ###

# sample size per group for each adaptation choice.
n.g1.s2<-75
n.g2.s2<-75
n.g3.s2<-75
n.g4.s2<-60

doses_stg2_g1<-c(d0,d2,d3,d4)
doses_stg2_g2<-c(d0,d1,d2,d4)
doses_stg2_g3<-c(d0,d1,d2,d3)
doses_stg2_g4<-c(d0,d1,d2,d3,d4)

n_stg2_g1<-c(rep(n.g1.s2,length(doses_stg2_g1)))
n_stg2_g2<-c(rep(n.g2.s2,length(doses_stg2_g2)))
n_stg2_g3<-c(rep(n.g3.s2,length(doses_stg2_g3)))
n_stg2_g4<-c(rep(n.g4.s2,length(doses_stg2_g4)))

compute_wt_ct<-function(c2_opt,W_a){
  Wc2_opt_m1=W_a[,1]*c2_opt[1,]
  Wc2_opt_m2=W_a[,2]*c2_opt[2,]
  Wc2_opt_m3=W_a[,3]*c2_opt[3,]
  c2_opt_stg2<-t(matrix(c(Wc2_opt_m1, Wc2_opt_m2, Wc2_opt_m3),nrow=k2andP))
  return(c2_opt_stg2)
}

#=====
# Find the critical value for testing Tmax (one-sided).
delta<-rep(0,m)
q_value_stg1<-qmvtn(0.975,df=df1,delta=delta,corr=cr1,
  abseps=1e-04,maxpts=1e+05,tail="lower")
q_stg1<-q_value_stg1$quantile

# Compute Critical Values for Global PoC test.#
chi_df4_p975L<-qchisq(p=0.975, df=4, ncp=0, lower.tail = TRUE, log.p = FALSE)
GH_Mi_critValp975L<-exp(-(chi_df4_p975L)/2)

### A. Constant Data-Generating Function #####

#####
# II. Stage 1: Data Generating #
#####

```

```

### Prepare the empty matrices to fill the results for evaluation (see #IV#) ###
# for Stage 1
ps_stg1 = array(c(0),dim=c(4,L))
# for Stage 2
ps_stg2 = array(c(0,0,0,0),dim=c(4,L))
  # column 1: M1 p-value for H(s,1) using a univ. T-dist,
  # column 2: M2 p-value for H(s,2) using a univ. T-dist,
  # column 3: M3 p-value for H(s,3) using a univ. T-dist,
  # column 4: mvt.prob1.
  # Note: s=1, 2 (stage).
ps_stg2.noWt = array(c(0,0,0,0),dim=c(4,L))

# T-stats for model 1, model 2, model 3 and Tmax.
Ts_stg1 = array(c(0,0,0,0),dim=c(4,L))
Ts_stg2 = array(c(0,0,0,0),dim=c(4,L))
Ts_stg2.noWt = array(c(0,0,0,0),dim=c(4,L))

# for Stage 2: for model 1, model 2, and model 3.
m.s2 = array(c(0,0,0),dim=c(3,L))

PoC_Gps = array(c(0,0,0,0),dim=c(4,L)) # for Global Tests.
  # column 1: product of two (pVal from MVT).
PoC_Gps.noWt = array(c(0,0,0,0),dim=c(4,L)) # for Global Tests.

# Results of L trials for Stage 1 and adaptive treatment decision "a" or "g".
H0_result1<-matrix(rep(0,L*3),ncol=3)

# Results of L trials for Stage 2, the number of models in Stage 2.
H0_result2<-matrix(rep(0,L*1),ncol=1)
H0_result2.noWt<-matrix(rep(0,L*1),ncol=1)

# Results of L trials for Global Tests.
H0_resultG<-matrix(rep(0,L*6),ncol=6)
  # column 1: overall global test based on MVT.
  # column 2: global test on M1.
  # column 3: global test on M2.
  # column 4: global test on M3.
  # column 5: Model # of the best model (the most significant model among M1,2,and 3).
  # column 6: 1 if the best Model is significant.
H0_resultG.noWt<-matrix(rep(0,L*6),ncol=6)

### Dataset 'trial' ###

# 1. Empty vectors& columns for dataset and specification.

# stage 1 #

simul.time<-
system.time(for(l in 1:L){

  # Empty data matrix to fill samples using the data generating function.
  mydata1<-matrix(0,nrow=n_stg1[1],ncol=k1andP)

```

```

# Fill 1 if PoC is rejected for the model with the smallest p-value in the lth trial.
# To fill mean value mu0 of response Y for each dose group.
mu0<-c(rep(0,k1andP))
# To fill std.dv value sd0 of response Y for each dose group.
sd0<-c(rep(0,k1andP))

## 3. Draw sample size of nj for dose group j in the lth trial.##
for(j in 1:k1andP){
  # std.dv of Y for dose j group are equally sd0=1.478.
  sd0[j]<-1.478
  # the true mean response mu0 at dose j group is defined by a constant function.
  mu0[j]<-0.2
  mydata1[,j]<-rnorm(n_stg1[j], mu0[j], sd0[j])
}

mymean1<-matrix(rep(0,k1andP),ncol=1)
for(j in 1:k1andP){
  mymean1[j]<-mean(mydata1[,j])
}
mymean1.bar<-mean(mymean1)
resid.ind1<-rep(0,k1andP)
for(j in 1:k1andP){
  resid.ind1[j]<-t(mydata1[,j]-mymean1[j]*
    Ident2.s1)%*(mydata1[,j]-mymean1[j]*Ident2.s1)
}
# Find SSE.
s1<-sqrt(sum(resid.ind1)/(N1-k1andP))

#####
# III. Stage 1: Run the MCP-Mod #
#####

# analyzing a trial

# stage 1 #

## 4. Obtain the test stat w/ the best contrasts (c1_opt_m) 'Tmax_m'##

# Test statistic function. #
T1<-function(c){t(c)%*mymean1/(s1*sqrt(t(c)%*(solve(diag(n_stg1)))%*c))}

# Test statistic T_m vector.#
T_m_stg1<-matrix(rep(0,m),nrow=1)
for(j in 1:m){
# For example, T_m1<-T(c1_opt_m1) is one element.
  T_m_stg1[j]<-T1(c1_opt_stg1[j,])
}

attributes(T_m_stg1[1])<-NULL
T_M1_stg1<-T_m_stg1[1]
attributes(T_m_stg1[2])<-NULL
T_M2_stg1<-T_m_stg1[2]
attributes(T_m_stg1[3])<-NULL

```

```

T_M3_stg1<-T_m_stg1[3]

T_m_stg1<-c(T_M1_stg1,T_M2_stg1,T_M3_stg1)
T_max1<-max(T_m_stg1)

#####
# IV. Evaluation of the MCP-Mod at every trial in Stage 1 #
#####

## 5. Test H0: the dose-response curve is flat ###
### Overall PoC test using MVT of m models in Stage 1.
### Note: this is using the critical value q_stg1.

mvt.prob1 <- pmvt(lower=T_m_stg1, delta=delta, df=df1, corr=cr1)
attributes(mvt.prob1)<-NULL
if(mvt.prob1 < 0.025){
  H0_result1[l,1]=1
}

### Individual models are tested based on a univariate t-distribution.
p_M1_stg1<-pt(T_M1_stg1, df1, lower.tail = FALSE, log.p = FALSE)
p_M2_stg1<-pt(T_M2_stg1, df1, lower.tail = FALSE, log.p = FALSE)
p_M3_stg1<-pt(T_M3_stg1, df1, lower.tail = FALSE, log.p = FALSE)

ps_stg1[,l]= c(p_M1_stg1,p_M2_stg1,p_M3_stg1,mvt.prob1)
Ts_stg1[,l] = c(T_m_stg1,T_max1) # T-stats for model 1, model 2, model 3 and Tmax.

#####
# V. Use ADTAR~I between Stages #
#####

# Check the result of Stage 1 to decide doses in Stage 2 based on ADTAR~I.

### 'a1' uses the result of individual D-R testings.

if((T_M1_stg1 > q_stg1)&(T_M2_stg1 > q_stg1)&(T_M3_stg1 > q_stg1)){
  a1=1
  g=1
}
if((T_M1_stg1 > q_stg1)&(T_M2_stg1 > q_stg1)&(T_M3_stg1 <= q_stg1)){
  a1=2
  g=1}
if((T_M1_stg1 > q_stg1)&(T_M2_stg1 <= q_stg1)&(T_M3_stg1 > q_stg1)){
  a1=3
  g=4
}
if((T_M1_stg1 <= q_stg1)&(T_M2_stg1 > q_stg1)&(T_M3_stg1 > q_stg1)){
  a1=4
  g=2
}

```

```

if((T_M1_stg1 > q_stg1)&(T_M2_stg1 <= q_stg1)&(T_M3_stg1 <= q_stg1)){
  a1=5
  g=4
}
if((T_M1_stg1 <= q_stg1)&(T_M2_stg1 > q_stg1)&(T_M3_stg1 <= q_stg1)){
  a1=6
  g=2
}
if((T_M1_stg1 <= q_stg1)&(T_M2_stg1 <= q_stg1)&(T_M3_stg1 > q_stg1)){
  a1=7
  g=3
}
if((T_M1_stg1 <= q_stg1)&(T_M2_stg1 <= q_stg1)&(T_M3_stg1 <= q_stg1)){
  a1=8
  g=4
}

# The result of Stage 1 into H0_result1 matrix.
H0_result1[1,2]=a1
H0_result1[1,3]=g

if (a1 <= 2){
  doses_stg2=doses_stg2_g1
}
if (a1 == 4|a1 == 6){
  doses_stg2=doses_stg2_g2
}
if (a1 == 7){
  doses_stg2=doses_stg2_g3
}
if (a1 == 3|a1 == 5|a1 == 8){
  doses_stg2=doses_stg2_g4
}

if (a1 <= 2){
  n_stg2=n_stg2_g1
}
if (a1 == 4|a1 == 6){
  n_stg2=n_stg2_g2
}
if (a1 == 7){
  n_stg2=n_stg2_g3
}
if (a1 == 3|a1 == 5|a1 == 8){
  n_stg2=n_stg2_g4
}

N2=sum(n_stg2)
k2andP=length(doses_stg2) # k + 1 in Stage 2 (# of treatment doses and the placebo).

#####
# VI. Stage 2: Contrast Vectors and Data Generating #

```

```
#####

Ident1.s2<-matrix(rep(1,k2andP),ncol=1)
Ident2.s2<-matrix(rep(1,n_stg2[1]),ncol=1)

# Weight the optimal contrast vector in Stage 2(plM_stg2$contMat).

if (a1 <= 2){
  W_a=W_g1
}
if (a1 == 4|a1 == 6){
  W_a=W_g2
}
if (a1 == 7){
  W_a=W_g3
}
if (a1 == 3|a1 == 5|a1 == 8){
  W_a=W_g4
}

#element-by-element operations on the weight and contrast vectors.
df2<-N2-k2andP
cr2<-compute_cr2(doses_stg2, n_stg2[1], W_a)

# correlation and optimal contrasts without weights in Stage 2.
cr2.noWt<-compute_cr1(doses_stg2, n_stg2[1])
c2_opt_m1<-compute_emax_c1_opt_m1(k2andP, n_stg2, doses_stg2)
c2_opt_m2<-compute_quad_c1_opt_m2(k2andP, n_stg2, doses_stg2)
c2_opt_m3<-compute_logis_c1_opt_m3(k2andP, n_stg2, doses_stg2)
c2_opt_org<-t(matrix(c(c2_opt_m1, c2_opt_m2, c2_opt_m3),nrow=k2andP))

# weighted optimal contrasts in Stage 2.
c2_opt_stg2<-compute_wt_ct(c2_opt_org,W_a)
Wc2_opt_m1<-c2_opt_stg2[1,]
Wc2_opt_m2<-c2_opt_stg2[2,]
Wc2_opt_m3<-c2_opt_stg2[3,]

# get rid of NA values from cr2.
cr2.sub<-fun(cr2)

### Dataset 'trial' ###

mydata2<-matrix(0,nrow=n_stg2[1],ncol=k2andP)
mu0_stg2<-c(rep(0,k2andP))
sd0_stg2<-c(rep(0,k2andP))

# 3. Fill the data from a data-generating function in Stage 2.
for(j in 1:k2andP){
  #for(l in 1:n_stg2[j]){
    # std.dv of Y for dose j group are equally sd0_stg2=1.478.
    sd0_stg2[j]<- 1.478
    mu0_stg2[j]<-0.2
  #}
}
```

```

mydata2[,j]<-rnorm(n_stg2[j], mu0_stg2[j], sd0_stg2[j])
}

mymean2<-matrix(rep(0,k2andP),ncol=1)
for(j in 1:k2andP){
  mymean2[j]<-mean(mydata2[,j])
}

mymean2.bar<-mean(mymean2)

resid.ind2<-rep(0,k2andP)
for(j in 1:k2andP){
  resid.ind2[j]<-t(mydata2[,j]-mymean2[j]*
  Ident2.s2)%*(mydata2[,j]-mymean2[j]*Ident2.s2)
}
s2<-sqrt(sum(resid.ind2)/(N2-k2andP))

#####
# VII. Stage 2: Run the Extended MCP-Mod      #
#####

# analyzing a trial

# stage 2 #

## 4. Obtain the test stat w/ the best contrasts (c2_opt_m) 'Tmax_m'##

# Test statistic function. #
T2<-function(c){t(c)%*mymean2/(s2*sqrt(t(c)%*(solve(diag(n_stg2))))%*c)}}

# Test statistic T_m vector.#
T_m_stg2<-matrix(rep(0,m),nrow=1)
T_m_stg2.noWt<-matrix(rep(0,m),nrow=1)

for(j in 1:m){
  T_m_stg2[j]<-T2(c2_opt_stg2[j,])
  T_m_stg2.noWt[j]<-T2(c2_opt_org[j,])
}

attributes(T_m_stg2[1])<-NULL
T_M1_stg2<-T_m_stg2[1]
attributes(T_m_stg2[2])<-NULL
T_M2_stg2<-T_m_stg2[2]
attributes(T_m_stg2[3])<-NULL
T_M3_stg2<-T_m_stg2[3]

T_m_stg2<-c(T_M1_stg2,T_M2_stg2,T_M3_stg2)

sum.m2<-sum(m.s2[,1])
# the number of retained models in Stage 2.
m2<-m-sum.m2
lower2<-rep(-Inf, m2)
subT.s2<-na.omit(T_m_stg2*diag(cr2))

```

```

attributes(subT.s2)<-NULL
T_max2<-ifelse(length(subT.s2)>1,max(subT.s2),subT.s2)

# Non weighted T stats in Stage 2.

attributes(T_m_stg2.noWt[1])<-NULL
T_M1_stg2.noWt<-T_m_stg2.noWt[1]
attributes(T_m_stg2.noWt[2])<-NULL
T_M2_stg2.noWt<-T_m_stg2.noWt[2]
attributes(T_m_stg2.noWt[3])<-NULL
T_M3_stg2.noWt<-T_m_stg2.noWt[3]

T_m_stg2.noWt<-c(T_M1_stg2.noWt,T_M2_stg2.noWt,T_M3_stg2.noWt)

subT.s2.noWt<-na.omit(T_m_stg2.noWt*diag(cr2.noWt))
attributes(subT.s2.noWt)<-NULL
T_max2.noWt<-ifelse(length(subT.s2.noWt)>1,max(subT.s2.noWt),subT.s2.noWt)

#####
# VIII. Evaluation of the MCP-Mod at every trial in Stage 2 #
#####

## 5. Test H0: the dose-response curve is flat ##

### Overall PoC test using MVT of m models in Stage 2.
### Note: this is using the critical value q_stg2.

delta.s2<-rep(0,length(subT.s2))
mvt.prob2 <- pmvt(lower=subT.s2, delta=delta.s2, df=df2, corr=cr2.sub)
attributes(mvt.prob2)<-NULL
if(mvt.prob2 < 0.025){
  H0_result2[1,1]=1
}

### Individual models are tested based on a univariate t-distribution.
p_M1_stg2<-pt(T_M1_stg2, df2, lower.tail = FALSE, log.p = FALSE)
p_M2_stg2<-pt(T_M2_stg2, df2, lower.tail = FALSE, log.p = FALSE)
p_M3_stg2<-pt(T_M3_stg2, df2, lower.tail = FALSE, log.p = FALSE)

ps_stg2[,1]= c(p_M1_stg2,p_M2_stg2,p_M3_stg2,mvt.prob2)
Ts_stg2[,1] = c(T_m_stg2,T_max2)

### Non weighted results using the critical value q_stg2.

delta.s2.noWt<-rep(0,length(subT.s2.noWt))
mvt.prob2.noWt <- pmvt(lower=subT.s2.noWt, delta=delta.s2.noWt,
  df=df2, corr=cr2.noWt)
attributes(mvt.prob2.noWt)<-NULL
if(mvt.prob2.noWt < 0.025){
  H0_result2.noWt[1,1]=1
}

```

```

### Individual models are tested based on a univariate t-distribution.
p_M1_stg2.noWt<-pt(T_M1_stg2.noWt, df2, lower.tail = FALSE, log.p = FALSE)
p_M2_stg2.noWt<-pt(T_M2_stg2.noWt, df2, lower.tail = FALSE, log.p = FALSE)
p_M3_stg2.noWt<-pt(T_M3_stg2.noWt, df2, lower.tail = FALSE, log.p = FALSE)

ps_stg2.noWt[,1]= c(p_M1_stg2.noWt,p_M2_stg2.noWt,p_M3_stg2.noWt,mvt.prob2.noWt)
Ts_stg2.noWt[,1] = c(T_m_stg2.noWt,T_max2.noWt)

#####
# IX. Evaluation of the MCP-Mod at every trial #
#####

# 1. Evaluate PoC establishment (PoA evaluation)

## Global Tests ##
# -> Here I used the decision rule by the CEF.

# 1. combine p-values #
PoC_Gps[2,1]<-p_M1_stg1*p_M1_stg2
PoC_Gps[3,1]<-p_M2_stg1*p_M2_stg2
PoC_Gps[4,1]<-p_M3_stg1*p_M3_stg2
PoC_Gps[1,1]<-mvt.prob1*mvt.prob2

# 2. test global H0 by Fisher's comb. test of CEF. assign "1" if rejected. #
H0_resultG[1,2]<-(ifelse(PoC_Gps[2,1] < GH_Mi_critValp975L,1,0))
H0_resultG[1,3]<-(ifelse(PoC_Gps[3,1] < GH_Mi_critValp975L,1,0))
H0_resultG[1,4]<-(ifelse(PoC_Gps[4,1] < GH_Mi_critValp975L,1,0))
H0_resultG[1,1]<-(ifelse(PoC_Gps[1,1] < GH_Mi_critValp975L,1,0))

## Global Tests without Weights ##

# 1. combine p-values #
PoC_Gps.noWt[2,1]<-p_M1_stg1*p_M1_stg2.noWt
PoC_Gps.noWt[3,1]<-p_M2_stg1*p_M2_stg2.noWt
PoC_Gps.noWt[4,1]<-p_M3_stg1*p_M3_stg2.noWt
PoC_Gps.noWt[1,1]<-mvt.prob1*mvt.prob2.noWt

# 2. test global H0 by Fisher's comb. test of CEF. assign "1" if rejected. #
H0_resultG.noWt[1,2]<-(ifelse(PoC_Gps.noWt[2,1] < GH_Mi_critValp975L,1,0))
H0_resultG.noWt[1,3]<-(ifelse(PoC_Gps.noWt[3,1] < GH_Mi_critValp975L,1,0))
H0_resultG.noWt[1,4]<-(ifelse(PoC_Gps.noWt[4,1] < GH_Mi_critValp975L,1,0))
H0_resultG.noWt[1,1]<-(ifelse(PoC_Gps.noWt[1,1] < GH_Mi_critValp975L,1,0))

minP_G<-c(PoC_Gps[2,1],PoC_Gps[3,1],PoC_Gps[4,1])
subminP<-na.omit(minP_G)
attributes(subminP)<-NULL
H0_resultG[1,5] = which(minP_G==min(subminP))
minP_G=H0_resultG[1,5]+1

```

```

# 1 if the model with the smallest p-product is significant by the CEF.
if(HO_resultG[1,minP_G] == 1){
  HO_resultG[1,6]=1
}

# Repeat for non weighting.
minP_G.noWt<-c(PoC_Gps.noWt[2,1],PoC_Gps.noWt[3,1],PoC_Gps.noWt[4,1])
subminP.noWt<-na.omit(minP_G.noWt)
attributes(subminP.noWt)<-NULL
HO_resultG.noWt[1,5] = which(minP_G.noWt==min(subminP.noWt))
minP_G.noWt=HO_resultG.noWt[1,5]+1
if(HO_resultG.noWt[1,minP_G.noWt] == 1){
  HO_resultG.noWt[1,6]=1
}

} # This is the end of i loop (see #1#).
)

#####
# X. Evaluation summary of L trials. #
#####

### MCP-Mod Hypothesis: no-response curve.
### =>If rejected, at least one of (M1, M2 M3) is significant.
### MVT based ###
## Stage 1 ##
PoA_stg1_MCPMod.MVT<-mean(HO_result1[,1])
PoA_stg1_MCPMod.MVT

## Stage 2 ##
PoA_stg2_ExMCPMod.MVT<-mean(HO_result2[,1])
PoA_stg2_ExMCPMod.MVT
## Global Test by Fisher's Combination Test ##
PoA_Global_ExMCPMod.MVT<-mean(HO_resultG[,1])
PoA_Global_ExMCPMod.MVT

## Stage 2 without Weights ##
PoA_stg2_ExMCPMod.MVT.noWt<-mean(HO_result2.noWt[,1])
PoA_stg2_ExMCPMod.MVT.noWt
## Global Test by Fisher's Combination Test ##
PoA_Global_ExMCPMod.MVT.noWt<-mean(HO_resultG.noWt[,1])
PoA_Global_ExMCPMod.MVT.noWt

### Summary of Results of MCP-Mod, the null hypotheses ###
PoA_result<-matrix(c(PoA_stg1_MCPMod.MVT,PoA_stg2_ExMCPMod.MVT,PoA_Global_ExMCPMod.MVT,
  PoA_stg2_ExMCPMod.MVT.noWt,PoA_Global_ExMCPMod.MVT.noWt),nrow=5)

PoA_result

user.cpu<-simul.time[1]
system.cpu<-simul.time[2]
elapsed.time<-simul.time[3]
print(cbind(user.cpu,system.cpu,elapsed.time))

```

Note: From here, we only show the different data-generation code. The rest of this program is omitted as the same coding is used for the other Emax, quadratic, and logistic data.

```

### B. Emax Data-Generating Function #####
### Dataset 'trial' ###
# stage 1 #
  for(j in 1:k1andP){
    sd0[j]<-1.478
    # the true mean response mu0 at dose j group is defined by a Emax [m=1] function.
    mu0[j]<-0.2+0.7*doses_stg1[j]/(0.2+doses_stg1[j])#
    mydata1[,j]<-rnorm(n_stg1[j], mu0[j], sd0[j])
  }

### Dataset 'trial' ###
# stage 2 #
  for(j in 1:k2andP){
    sd0_stg2[j]<- 1.478 #1.478
    mu0_stg2[j]<-0.2+0.7*doses_stg2[j]/(0.2+doses_stg2[j])# Emax [m=1]
    mydata2[,j]<-rnorm(n_stg2[j], mu0_stg2[j], sd0_stg2[j])
  }

### C. Quadratic Data-Generating Function #####
### Dataset 'trial' ###
# stage 1 #
  for(j in 1:k1andP){
    sd0[j]<-1.478
    # the true mean response mu0 at dose j gp is defined by a Quadratic [m=2] funct.
    mu0[j]<-0.2+2.049*doses_stg1[j]-1.749*(doses_stg1[j])^2
    mydata1[,j]<-rnorm(n_stg1[j], mu0[j], sd0[j])
  }

### Dataset 'trial' ###
# stage 2 #
  for(j in 1:k2andP){
    sd0_stg2[j]<- 1.478
    mu0_stg2[j]<-0.2+2.049*doses_stg2[j]-1.749*(doses_stg2[j])^2
    mydata2[,j]<-rnorm(n_stg2[j], mu0_stg2[j], sd0_stg2[j])
  }

### D. Logistic Data-Generating Function #####

```

```

### Dataset 'trial' ###

# stage 1 #

for(j in 1:k1andP){
  sd0[j]<-1.478
  # the true mean response mu0 at dose j gp is defined by a Logistic [m=3] funct.
  mu0[j]<-0.193+0.607/(1+exp(10*log(3)*(0.4-doses_stg1[j])))
  mydata1[,j]<-rnorm(n_stg1[j], mu0[j], sd0[j])
}

### Dataset 'trial' ###

# stage 2 #

for(j in 1:k2andP){
  sd0_stg2[j]<- 1.478
  mu0_stg2[j]<-0.193+0.607/(1+exp(10*log(3)*(0.4-doses_stg2[j])))
  mydata2[,j]<-rnorm(n_stg2[j], mu0_stg2[j], sd0_stg2[j])
}

```

## A.2 PROOF OF ACTIVITY (POA) OF *ONE-STAGE DESIGN* (CHAPTER 3)

```

library(mvtnorm)

#####
### Note: This program performs Establishment of PoC.
###       The total sample size of the one-stage design is set to be 80 (16 x 5 groups).
###       This number must be replaced to 200, 400, or 600 for the other simulations.

#####
# I. planning a trial      #
#####

L <-10000
m<-3
### Stage 1 setting ###

n0_stg1 <- 16    # Input sample size for Placebo
n1_stg1 <- 16    # Input sample size for Treatment 1 (d=0.05)
n2_stg1 <- 16    # Input sample size for treatment 2 (d=0.2)
n3_stg1 <- 16    # Input sample size for treatment 3 (d=0.6)
n4_stg1 <- 16    # Input sample size for treatment 4 (d=1.0)

```

```

n_stg1<-c(n0_stg1,n1_stg1,n2_stg1,n3_stg1,n4_stg1)
doses_stg1<-c(0,0.05,0.2,0.6,1.0)
N1=sum(n_stg1)
# k + 1 in Stage 1 (# of treatment doses and the placebo).
k1andP=length(doses_stg1)

### Model Specification, Standardizing Models ###

## 1. Get optimal regression coefficients (c_opt). ##
delta.max<-0.6
ED50<-0.2
alpha<-0.025
Ident1.s1<-matrix(rep(1,k1andP),ncol=1)
Ident2.s1<-matrix(rep(1,n_stg1[1]),ncol=1)

### I. Standardized Emax Model.

# First, write the standardized Emax model.
f_m1_0<-function(d){
  d/(ED50+d)
}
mu_m1.s1<-matrix(rep(0,k1andP),ncol=1)
for(k in 1:k1andP){
  d=doses_stg1[k]
  mu_m1.s1[k]=f_m1_0(d)
}

# Then, compute c1_opt for model I in Stage 1.
mubar.m.s1<-function(mu){(sum(mu)/k1andP)*Ident1.s1}
mudiff.m.s1<-function(mu,mubar.m.s1){mu-mubar.m.s1}
c1_opt_m<-function(mudiff.m.s1){mudiff.m.s1
  %*(solve(sqrt(t(mudiff.m.s1)*%*%mudiff.m.s1)))}
mubar.m1.s1<-mubar.m.s1(mu_m1.s1)
mudiff.m1.s1<-mudiff.m.s1(mu_m1.s1,mubar.m1.s1)
c1_opt_m1<-c1_opt_m(mudiff.m1.s1)

### II. Standardized quadratic Model.
# Write the standadized quadratic model.
f_m2_0<-function(d){
  d-delta.max*d^2
}

mu_m2.s1<-rep(0,k1andP)
for(k in 1:k1andP){
  d=doses_stg1[k]
  mu_m2.s1[k]=f_m2_0(d)
}

# Compute c1_opt for model II in Stage 1.
mubar.m2.s1<-mubar.m.s1(mu_m2.s1)
mudiff.m2.s1<-mudiff.m.s1(mu_m2.s1,mubar.m2.s1)

```

```

c1_opt_m2<-c1_opt_m(mudiff.m2.s1)
c1_opt_m2

### III. Standardized Logistic Model.
# Write the standardized logistic model.
  f_m3_0<-function(d){
    1/(1+exp((ED50-d)/delta.max))
  }

mu_m3.s1<-rep(0,k1andP)
for(k in 1:k1andP){
  d=doses_stg1[k]
  mu_m3.s1[k]=f_m3_0(d)
}

# Compute c_ppt for model III in Stage 1.
mubar.m3.s1<-mubar.m.s1(mu_m3.s1)
mudiff.m3.s1<-mudiff.m.s1(mu_m3.s1,mubar.m3.s1)
c1_opt_m3<-c1_opt_m(mudiff.m3.s1)

c1_opt_stg1<-t(matrix(c(c1_opt_m1, c1_opt_m2, c1_opt_m3),nrow=k1andP))

## 2. Find the critical value for PoC test H0 ###
V1<-diag(1/n_stg1)
df1<-N1-k1andP
C1<-matrix(c1_opt_stg1, nrow=m)
cv1<-C1%*%V1%*%t(C1)
dv1<-t(1/sqrt(diag(cv1)))
cr1<-cv1*(t(dv1)%*%dv1)
delta<-rep(0,m)

# Find the critical value to test Tmax (one-sided).
q_value_stg1<-qmvmt(0.975,df=df1, delta=delta, corr=cr1,
  abseps=1e-04, maxpts=1e+05, tail="lower")
q_stg1<-q_value_stg1$quantile

# Compute Critical Values for Global PoC test.#
# critical value at lower 975%tile. #
chi_df4_p975L<-qchisq(p=0.975, df=4, ncp=0, lower.tail = TRUE, log.p = FALSE)
GH_Mi_critValp975L<-exp(-(chi_df4_p975L)/2)

### A. Constant Data-Generating Function #####

#####
# II. Stage 1: Data Generating      #
#####

### Prepare the empty matrices to fill the results###
ps_stg1 = array(c(0,0,0,0,0,0),dim=c(6,L))
Ts_stg1 = array(c(0,0,0,0),dim=c(4,L))
H0_result1<-matrix(rep(0,L*4),ncol=4)

```

```

### Dataset 'trial' ###

# 1. Empty vectors&columns for dataset and specification.

# stage 1 #

simul.time<-
system.time(for(l in 1:L){

  mydata1<-matrix(0,nrow=n_stg1[1],ncol=k1andP)
  mu0<-c(0,0,0,0,0)
  sd0<-c(0,0,0,0,0)

  ## 3. Draw sample size of nj for dose group j in the lth trial.##
  for(j in 1:k1andP){
    sd0[j]<-1.478
    mu0[j]<-0.2
    mydata1[,j]<-rnorm(n_stg1[j], mu0[j], sd0[j])
  }

  mymean1<-matrix(rep(0,k1andP),ncol=1)
  for(j in 1:k1andP){
    mymean1[j]<-mean(mydata1[,j])
  }

  mymean1.bar<-mean(mymean1)

  resid.ind1<-rep(0,k1andP)
  for(j in 1:k1andP){
    resid.ind1[j]<-t(mydata1[,j]-mymean1[j]
      *Ident2.s1)%*(mydata1[,j]-mymean1[j]*Ident2.s1)
  }

  s1<-sqrt(sum(resid.ind1)/(N1-k1andP))

#####
# III. Stage 1: Run the MCP-Mod      #
#####

# analyzing a trial

# stage 1 #

## 4. Obtain the test stat w/ the best contrasts (c1_opt_m) 'Tmax_m'##

# Test statistic function. #
T1<-function(c){t(c)%*mymean1/(s1*sqrt(t(c)%*(solve(diag(n_stg1)))%*c))}

# Test statistic T_m vector.#
T_m_stg1<-matrix(rep(0,m),nrow=1)

```

```

for(j in 1:m){
  T_m_stg1[j]<-T1(c1_opt_stg1[j,])
}

attributes(T_m_stg1[1])<-NULL
T_M1_stg1<-T_m_stg1[1]
attributes(T_m_stg1[2])<-NULL
T_M2_stg1<-T_m_stg1[2]
attributes(T_m_stg1[3])<-NULL
T_M3_stg1<-T_m_stg1[3]

T_m_stg1<-c(T_M1_stg1,T_M2_stg1,T_M3_stg1)

#####
# IV. Evaluation of the MCP-Mod at every trial in Stage 1 #
#####

## 5. Test H0: the dose-response curve is flat ###
### Overall PoC test using MVT of m models in Stage 1.

mvt.prob1 <- pmvt(lower=T_m_stg1, delta=delta, df=df1, corr=cr1)
attributes(mvt.prob1)<-NULL
if(mvt.prob1 < 0.025){H0_result1[1,4]=1}

### Individual models are tested based on a univariate t-distribution.
p_M1_stg1<-pt(T_M1_stg1, df1, lower.tail = FALSE, log.p = FALSE)
p_M2_stg1<-pt(T_M2_stg1, df1, lower.tail = FALSE, log.p = FALSE)
p_M3_stg1<-pt(T_M3_stg1, df1, lower.tail = FALSE, log.p = FALSE)

ps_stg1[,1]= c(p_M1_stg1,p_M2_stg1,p_M3_stg1,SK.p1,MC.p1,mvt.prob1)
Ts_stg1[,1] = c(T_m_stg1,T_max1)

#####
# V. Check the interim result between Stages #
#####

# Check the result of Stage 1 to decide doses in Stage 2 based on ADTAR~I.

if((p_M1_stg1 <= 0.025)&(p_M2_stg1 <= 0.025)&(p_M3_stg1 <= 0.025)){
  a1=1
}
if((p_M1_stg1 <= 0.025)&(p_M2_stg1 <= 0.025)&(p_M3_stg1 > 0.025)){
  a1=2
}
if((p_M1_stg1 <= 0.025)&(p_M2_stg1 > 0.025)&(p_M3_stg1 <= 0.025)){
  a1=3
}
}

```

```

if((p_M1_stg1 > 0.025)&(p_M2_stg1 <= 0.025)&(p_M3_stg1 <= 0.025)){
  a1=4
}
if((p_M1_stg1 <= 0.025)&(p_M2_stg1 > 0.025)&(p_M3_stg1 > 0.025)){
  a1=5
}
if((p_M1_stg1 > 0.025)&(p_M2_stg1 <= 0.025)&(p_M3_stg1 > 0.025)){
  a1=6
}
if((p_M1_stg1 > 0.025)&(p_M2_stg1 > 0.025)&(p_M3_stg1 <= 0.025)){
  a1=7
}
if((p_M1_stg1 > 0.025)&(p_M2_stg1 > 0.025)&(p_M3_stg1 > 0.025)){
  a1=8
}

H0_result1[1,3]=a1

#####
# IX. Evaluation of the MCP-Mod at every trial #
#####

# 1. Evaluate PoC establishment (PoA evaluation)

}      # This is the end of i loop (see #1#).
)

#####
# X. Evaluation summary of L trials.      #
#####

### MCP-Mod Hypothesis: no-response curve.
### MVT based ###
## Stage 1 ##
PoA_stg1_MCPMod.MVT<-mean(HO_result1[,4])
PoA_stg1_MCPMod.MVT

user.cpu<-simul.time[1]
system.cpu<-simul.time[2]
elapsed.time<-simul.time[3]
print(cbind(user.cpu,system.cpu,elapsed.time))

```

Note: The rest of program is omitted as the same coding is used for the other Emax, quadratic, and logistic data. For the different data-generation code, see Appendix [A.1](#).

### A.3 PROOF OF ACTIVITY (POA) OF *FIXED TWO-STAGE DESIGNS* (CHAPTER 3)

```

library(mvtnorm)

#####

### This program performs Establishment of PoC.
### Note: The total sample size in Study 2 are designed to be similar to that in Study 1.
### Equal sample sizes in Study 1 are set to be 75. For the other simulations,
### This number must be replaced to 10, 25, or 50. Equal sample sizes in Study 2
### must be also replaced accordingly.

library(mvtnorm)

#####
# I. planning a trial      #
#####

# Number of simulated trials.
L <-10000

# number of the candidate D-R models.
m<-3

# number of adaptation choices in Stage 2.
adpt.L<-4

# (adpt.L choices x k+1 doses) design matrix for dose adaptation in Stage 2.
Z<-matrix(c(1,1,1,1,0,1,1,1,1,1,1,1,1,0,1,1,1,1,0,1),nrow=adpt.L)
tZ<-t(Z)

# Matrix for associating PoC results in Stage 1 with adaptations in Stage 2.
JK<-matrix(c(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1),nrow=adpt.L)

### Model Specification, Standardizing Models, and Empty Matrices for Evaluation ###

## 1. Obtain optimal regression coefficients (c_opt). ##

# maximize the power of the test == non-centrality parameter with regulatory condition.
# Using ragranges, we solve the formula and obtain c_opt vector.

# Maximum effect size.
delta.max<-0.6

# Dose to produce the half of the maximum effect.
ED50<-0.2
alpha<-0.025

```

```

compute_emax_c1_opt_m1 = function(kandP, n_stg, doses_stg) {
  # Identity matrix of the number of dose size.
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  # Identity matrix of the number of patients/group size.
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### I. Standardized Emax Model.

  # First, write the standadized Emax model.
  f_m1_0<-function(d){
    d/(ED50+d)
  }
  mu_m1.s1<-matrix(rep(0,kandP),ncol=1)
  for(k in 1:kandP){
    d=doses_stg[k]
    mu_m1.s1[k]=f_m1_0(d)
  }

  # Compute c1_opt for model I in Stage 1.
  # Define functions.
  mubar.m.s1<-function(mu){
    (sum(mu)/kandP)*Ident1.s1
  }
  mudiff.m.s1<-function(mu,mubar.m.s1){
    mu-mubar.m.s1
  }
  c1_opt_m<-function(mudiff.m.s1){
    mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1)%*%mudiff.m.s1)))
  }
  mubar.m1.s1<-mubar.m.s1(mu_m1.s1)
  mudiff.m1.s1<-mudiff.m.s1(mu_m1.s1,mubar.m1.s1)
  c1_opt_m1<-c1_opt_m(mudiff.m1.s1)
  return(c1_opt_m1)
}

```

```

compute_quad_c1_opt_m2 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### II. Standardized quadratic Model.

  # The standadized quadratic model.
  f_m2_0<-function(d){
    d-delta.max*d^2
  }
  mu_m2.s1<-rep(0,kandP)
  for(k in 1:kandP){
    d=doses_stg[k]
    mu_m2.s1[k]=f_m2_0(d)
  }

  mubar.m.s1<-function(mu){
    (sum(mu)/kandP)*Ident1.s1
  }

```

```

}
mudiff.m.s1<-function(mu,mubar.m.s1){
  mu-mubar.m.s1
}
c1_opt_m<-function(mudiff.m.s1){
  mudiff.m.s1**%(solve(sqrt(t(mudiff.m.s1)**%mudiff.m.s1)))
}

# Compute c1_opt for model II in Stage 1.
mubar.m2.s1<-mubar.m.s1(mu_m2.s1)
mudiff.m2.s1<-mudiff.m.s1(mu_m2.s1,mubar.m2.s1)
c1_opt_m2<-c1_opt_m(mudiff.m2.s1)

return(c1_opt_m2)
}

compute_logis_c1_opt_m3 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### III. Standardized Logistic Model.

  # The standardized logistic model.
  f_m3_0<-function(d){
    1/(1+exp((ED50-d)/delta.max))
  }

  mu_m3.s1<-rep(0,kandP)
  for(k in 1:kandP){
    d=doses_stg[k]
    mu_m3.s1[k]=f_m3_0(d)
  }

  mubar.m.s1<-function(mu){
    (sum(mu)/kandP)*Ident1.s1
  }
  mudiff.m.s1<-function(mu,mubar.m.s1){
    mu-mubar.m.s1
  }
  c1_opt_m<-function(mudiff.m.s1){
    mudiff.m.s1**%(solve(sqrt(t(mudiff.m.s1)**%mudiff.m.s1)))
  }

  # Compute c_ppt for model III in Stage 1.
  mubar.m3.s1<-mubar.m.s1(mu_m3.s1)
  mudiff.m3.s1<-mudiff.m.s1(mu_m3.s1,mubar.m3.s1)
  c1_opt_m3<-c1_opt_m(mudiff.m3.s1)
  return(c1_opt_m3)
}

compute_cr1 = function(doses_stg, samples){
  # k + 1 in Stage 1 (# of treatment doses and the placebo).

```

```

k1andP=length(doses_stg)
n_stg1<-replicate(k1andP, samples)
N1=sum(n_stg1)
c1_opt_m1<-compute_emax_c1_opt_m1(k1andP, n_stg1, doses_stg)
c1_opt_m2<-compute_quad_c1_opt_m2(k1andP, n_stg1, doses_stg)
c1_opt_m3<-compute_logis_c1_opt_m3(k1andP, n_stg1, doses_stg)
c1_opt_stg1<-t(matrix(c(c1_opt_m1, c1_opt_m2, c1_opt_m3),nrow=k1andP))

## 2. Find the critical value for PoC test H0 ###
# Refer to 'UsingMvtPackage.pdf'.
# The decision rule: use the best contrasts for computing the test statistic (Tmax).
# compute the critical value for the H0 test.
# Note: the linear contrast matrix under H0 is C1.
V1<-diag(1/n_stg1)
df1<-N1-k1andP
# [m x (k+1)] matrix.
C1<-matrix(c1_opt_stg1, nrow=m)
cv1<-C1%*%V1%*%t(C1)
dv1<-t(1/sqrt(diag(cv1)))
# this is the correlation matrix under H0.
cr1<-cv1*(t(dv1)%*%dv1)
return(cr1)
}

### Common Setting ###
d0<-0
d1<-0.05
d2<-0.2
d3<-0.6
d4<-1.0

### Stage 1 setting ###

n.s1<-75
doses_stg1<-c(d0,d1,d2,d3)

k1andP=length(doses_stg1)
n_stg1=c(rep(n.s1,k1andP))
N1<-sum(n_stg1)
df1<-N1-k1andP
cr1<-compute_cr1(doses_stg1, n.s1)
  c1_opt_m1<-compute_emax_c1_opt_m1(k1andP, n_stg1, doses_stg1)
  c1_opt_m2<-compute_quad_c1_opt_m2(k1andP, n_stg1, doses_stg1)
  c1_opt_m3<-compute_logis_c1_opt_m3(k1andP, n_stg1, doses_stg1)
c1_opt_stg1<-t(matrix(c(c1_opt_m1, c1_opt_m2, c1_opt_m3),nrow=k1andP))
Ident1.s1<-matrix(rep(1,k1andP),ncol=1)
Ident2.s1<-matrix(rep(1,n_stg1[1]),ncol=1)

### Stage 2 setting ###

# sample size per group for each adaptation choice.

```

```

# Dose combination is the same as aI in ADTARI #

n0_stg2 <- 75    # Input sample size for Placebo
n2_stg2 <- 75    # Input sample size for treatment 2 (d=0.2)
n3_stg2 <- 75    # Input sample size for treatment 3 (d=0.6)
n4_stg2 <- 75    # Input sample size for treatment 4 (d=1.0)

# set sample sizes for each fixed dose combination in Stage 2.
n_stg2<-c(n0_stg2,n2_stg2,n3_stg2,n4_stg2) # For b_V fixed dose combination.

doses_stg2<-c(0,0.2,0.6,1.0)

#####
# Note: For the other three fixed two-stage designs,
# the following sample sizes and/or doses are used in Stage 2.
#
# <Fixed two-stage design "aII">
# n0_stg2 <- 75    # Input sample size for Placebo
# n1_stg2 <- 75    # Input sample size for treatment 1 (d=0.05)
# n2_stg2 <- 75    # Input sample size for treatment 2 (d=0.2)
# n4_stg2 <- 75    # Input sample size for treatment 4 (d=1.0)
# n_stg2<-c(n0_stg2,n1_stg2,n2_stg2,n4_stg2)
# doses_stg2<-c(0,0.05,0.2,1.0)
#
# <Fixed two-stage design "aIII">
# n0_stg2 <- 75    # Input sample size for Placebo
# n1_stg2 <- 75    # Input sample size for treatment 1 (d=0.05)
# n2_stg2 <- 75    # Input sample size for treatment 2 (d=0.2)
# n3_stg2 <- 75    # Input sample size for treatment 3 (d=0.6)
# n_stg2<-c(n0_stg2,n1_stg2,n2_stg2,n3_stg2)
# doses_stg2<-c(0,0.05,0.2,0.6)
#
# <Fixed two-stage design "aIV">
# n0_stg2 <- 60    # Input sample size for Placebo
# n1_stg2 <- 60    # Input sample size for treatment 1 (d=0.05)
# n2_stg2 <- 60    # Input sample size for treatment 2 (d=0.2)
# n3_stg2 <- 60    # Input sample size for treatment 3 (d=0.6)
# n4_stg2 <- 60    # Input sample size for treatment 4 (d=1.0)
# n_stg2<-c(n0_stg2,n1_stg2,n2_stg2,n3_stg2,n4_stg2)
# doses_stg2<-c(0,0.05,0.2,0.6,1.0)
#
#####

N2=sum(n_stg2)
# k + 1 in Stage 2 (# of treatment doses and the placebo).
k2andP=length(doses_stg2)

#=====
# Find the critical value for testing Tmax (one-sided).
delta<-rep(0,m)
q_value_stg1<-qmvmt(0.975,df=df1,delta=delta,corr=cr1,

```

```

    abseps=1e-04,maxpts=1e+05,tail="lower")
q_stg1<-q_value_stg1$quantile

# Compute Critical Values for Global PoC test.#
chi_df4_p975L<-qchisq(p=0.975, df=4, ncp=0, lower.tail = TRUE, log.p = FALSE)
GH_Mi_critValp975L<-exp(-(chi_df4_p975L)/2)

### A. Constant Data-Generating Function #####

#####
# II. Stage 1: Data Generating      #
#####

### Prepare the empty matrices to fill the results for evaluation (see #IV#) ###
# for Stage 1
ps_stg1 = array(c(0),dim=c(4,L))
# for Stage 2
ps_stg2.noWt = array(c(0,0,0,0),dim=c(4,L))
  # column 1: M1 p-value for H(s,1) using a univ. T-dist,
  # column 2: M2 p-value for H(s,2) using a univ. T-dist,
  # column 3: M3 p-value for H(s,3) using a univ. T-dist,
  # column 4: mvt.prob1.
  # Note: s=1, 2 (stage).

# T-stats for model 1, model 2, model 3 and Tmax.
Ts_stg1 = array(c(0,0,0,0),dim=c(4,L))
Ts_stg2.noWt = array(c(0,0,0,0),dim=c(4,L))

PoC_Gps.noWt = array(c(0,0,0,0),dim=c(4,L)) # for Global Tests.
  # column 1: product of two (pVal from MVT).

# Results of L trials for Stage 1 and adaptive treatment decision "a" or "g".
HO_result1<-matrix(rep(0,L*3),ncol=3)

# Results of L trials for Stage 2,the number of models in Stage 2.
HO_result2.noWt<-matrix(rep(0,L*1),ncol=1)

# Results of L trials for Global Tests.
HO_resultG.noWt<-matrix(rep(0,L*6),ncol=6)
  # column 1: overall global test based on MVT.
  # column 2: global test on M1.
  # column 3: global test on M2.
  # column 4: global test on M3.
  # column 5: Model # of the best model (the most significant model among M1,2,and 3).
  # column 6: 1 if the best Model is significant.

### Dataset 'trial' ###

# 1. Empty vectors& columns for dataset and specification.

```

```

# stage 1 #

simul.time<-
system.time(for(l in 1:L){

  # Empty data matrix to fill samples using the data generating function.
  mydata1<-matrix(0,nrow=n_stg1[1],ncol=k1andP)

  # Fill 1 if PoC is rejected for the model with the smallest p-value in the lth trial.
  # To fill mean value mu0 of response Y for each dose group.
  mu0<-c(rep(0,k1andP))
  # To fill std.dv value sd0 of response Y for each dose group.
  sd0<-c(rep(0,k1andP))

  ## 3. Draw sample size of nj for dose group j in the lth trial.##
  for(j in 1:k1andP){
    # std.dv of Y for dose j group are equally sd0=1.478.
    sd0[j]<-1.478
    # the true mean response mu0 at dose j group is defined by a constant function.
    mu0[j]<-0.2
    mydata1[,j]<-rnorm(n_stg1[j], mu0[j], sd0[j])
  }

  mymean1<-matrix(rep(0,k1andP),ncol=1)
  for(j in 1:k1andP){
    mymean1[j]<-mean(mydata1[,j])
  }
  mymean1.bar<-mean(mymean1)
  resid.ind1<-rep(0,k1andP)
  for(j in 1:k1andP){
    resid.ind1[j]<-t(mydata1[,j]-mymean1[j]*
      Ident2.s1)%*(mydata1[,j]-mymean1[j]*Ident2.s1)
  }
  # Find SSE.
  s1<-sqrt(sum(resid.ind1)/(N1-k1andP))

#####
# III. Stage 1: Run the MCP-Mod #
#####

# analyzing a trial

# stage 1 #

## 4. Obtain the test stat w/ the best contrasts (c1_opt_m) 'Tmax_m'##

# Test statistic function. #
T1<-function(c){t(c)%*mymean1/(s1*sqrt(t(c)%*(solve(diag(n_stg1)))%*c))}

# Test statistic T_m vector.#
T_m_stg1<-matrix(rep(0,m),nrow=1)
for(j in 1:m){
  # For example, T_m1<-T(c1_opt_m1) is one element.

```

```

    T_m_stg1[j]<-T1(c1_opt_stg1[j,])
}

attributes(T_m_stg1[1])<-NULL
T_M1_stg1<-T_m_stg1[1]
attributes(T_m_stg1[2])<-NULL
T_M2_stg1<-T_m_stg1[2]
attributes(T_m_stg1[3])<-NULL
T_M3_stg1<-T_m_stg1[3]

T_m_stg1<-c(T_M1_stg1,T_M2_stg1,T_M3_stg1)
T_max1<-max(T_m_stg1)

#####
# IV. Evaluation of the MCP-Mod at every trial in Stage 1 #
#####

## 5. Test H0: the dose-response curve is flat ###
### Overall PoC test using MVT of m models in Stage 1.
### Note: this is using the critical value q_stg1.

mvt.prob1 <- pmvt(lower=T_m_stg1, delta=delta, df=df1, corr=cr1)
attributes(mvt.prob1)<-NULL
if(mvt.prob1 < 0.025){
  H0_result1[l,1]=1
}

### Individual models are tested based on a univariate t-distribution.
p_M1_stg1<-pt(T_M1_stg1, df1, lower.tail = FALSE, log.p = FALSE)
p_M2_stg1<-pt(T_M2_stg1, df1, lower.tail = FALSE, log.p = FALSE)
p_M3_stg1<-pt(T_M3_stg1, df1, lower.tail = FALSE, log.p = FALSE)

ps_stg1[,l]= c(p_M1_stg1,p_M2_stg1,p_M3_stg1,mvt.prob1)
Ts_stg1[,l] = c(T_m_stg1,T_max1) # T-stats for model 1, model 2, model 3 and Tmax.

#####
# V. Check the results of Stage 1 #
#####

# Check the result of Stage 1.

### 'a1' uses the result of individual D-R testings.

if((T_M1_stg1 > q_stg1)&(T_M2_stg1 > q_stg1)&(T_M3_stg1 > q_stg1)){
  a1=1
}
if((T_M1_stg1 > q_stg1)&(T_M2_stg1 > q_stg1)&(T_M3_stg1 <= q_stg1)){
  a1=2
}
if((T_M1_stg1 > q_stg1)&(T_M2_stg1 <= q_stg1)&(T_M3_stg1 > q_stg1)){

```

```

    a1=3
  }
  if((T_M1_stg1 <= q_stg1)&(T_M2_stg1 > q_stg1)&(T_M3_stg1 > q_stg1)){
    a1=4
  }
  if((T_M1_stg1 > q_stg1)&(T_M2_stg1 <= q_stg1)&(T_M3_stg1 <= q_stg1)){
    a1=5
  }
  if((T_M1_stg1 <= q_stg1)&(T_M2_stg1 > q_stg1)&(T_M3_stg1 <= q_stg1)){
    a1=6
  }
  if((T_M1_stg1 <= q_stg1)&(T_M2_stg1 <= q_stg1)&(T_M3_stg1 > q_stg1)){
    a1=7
  }
  if((T_M1_stg1 <= q_stg1)&(T_M2_stg1 <= q_stg1)&(T_M3_stg1 <= q_stg1)){
    a1=8
  }

# The result of Stage 1 into H0_result1 matrix.
H0_result1[1,2]=a1

#####
# VI. Stage 2: Contrast Vectors and Data Generating      #
#####

Ident1.s2<-matrix(rep(1,k2andP),ncol=1)
Ident2.s2<-matrix(rep(1,n_stg2[1]),ncol=1)

#element-by-element operations on the weight and contrast vectors.
df2<-N2-k2andP

# correlation and optimal contrasts without weights in Stage 2.
cr2.noWt<-compute_cr1(doses_stg2, n_stg2[1])
c2_opt_m1<-compute_emax_c1_opt_m1(k2andP, n_stg2, doses_stg2)
c2_opt_m2<-compute_quad_c1_opt_m2(k2andP, n_stg2, doses_stg2)
c2_opt_m3<-compute_logis_c1_opt_m3(k2andP, n_stg2, doses_stg2)
c2_opt_org<-t(matrix(c(c2_opt_m1, c2_opt_m2, c2_opt_m3),nrow=k2andP))

### Dataset 'trial' ###

mydata2<-matrix(0,nrow=n_stg2[1],ncol=k2andP)
mu0_stg2<-c(rep(0,k2andP))
sd0_stg2<-c(rep(0,k2andP))

# 3. Fill the data from a data-generating function in Stage 2.
for(j in 1:k2andP){
  #for(l in 1:n_stg2[j]){
    # std.dv of Y for dose j group are equally sd0_stg2=1.478.
    sd0_stg2[j]<- 1.478
    mu0_stg2[j]<-0.2
  #}
}

```

```

mydata2[,j]<-rnorm(n_stg2[j], mu0_stg2[j], sd0_stg2[j])
}

mymean2<-matrix(rep(0,k2andP),ncol=1)
for(j in 1:k2andP){
  mymean2[j]<-mean(mydata2[,j])
}

mymean2.bar<-mean(mymean2)

resid.ind2<-rep(0,k2andP)
for(j in 1:k2andP){
  resid.ind2[j]<-t(mydata2[,j]-mymean2[j]*
  Ident2.s2)%*(mydata2[,j]-mymean2[j]*Ident2.s2)
}
s2<-sqrt(sum(resid.ind2)/(N2-k2andP))

#####
# VII. Stage 2: Run the Extended MCP-Mod      #
#####

# analyzing a trial

# stage 2 #

## 4. Obtain the test stat w/ the best contrasts (c2_opt_m) 'Tmax_m'##

# Test statistic function. #
T2<-function(c){t(c)%*mymean2/(s2*sqrt(t(c)%*(solve(diag(n_stg2))))%*c)}}

# Test statistic T_m vector.#
T_m_stg2.noWt<-matrix(rep(0,m),nrow=1)

for(j in 1:m){
  T_m_stg2.noWt[j]<-T2(c2_opt_org[j,])
}

# Non weighted T stats in Stage 2.

attributes(T_m_stg2.noWt[1])<-NULL
T_M1_stg2.noWt<-T_m_stg2.noWt[1]
attributes(T_m_stg2.noWt[2])<-NULL
T_M2_stg2.noWt<-T_m_stg2.noWt[2]
attributes(T_m_stg2.noWt[3])<-NULL
T_M3_stg2.noWt<-T_m_stg2.noWt[3]

T_m_stg2.noWt<-c(T_M1_stg2.noWt,T_M2_stg2.noWt,T_M3_stg2.noWt)

subT.s2.noWt<-na.omit(T_m_stg2.noWt*diag(cr2.noWt))
attributes(subT.s2.noWt)<-NULL
T_max2.noWt<-ifelse(length(subT.s2.noWt)>1,max(subT.s2.noWt),subT.s2.noWt)

```

```

#####
# VIII. Evaluation of the MCP-Mod at every trial in Stage 2 #
#####

## 5. Test H0: the dose-response curve is flat ###

### Non weighted results using the critical value q_stg2.

delta.s2.noWt<-rep(0,length(subT.s2.noWt))
mvt.prob2.noWt <- pmvt(lower=subT.s2.noWt, delta=delta.s2.noWt,
  df=df2, corr=cr2.noWt)
attributes(mvt.prob2.noWt)<-NULL
if(mvt.prob2.noWt < 0.025){
  H0_result2.noWt[1,1]=1
}

### Individual models are tested based on a univariate t-distribution.
p_M1_stg2.noWt<-pt(T_M1_stg2.noWt, df2, lower.tail = FALSE, log.p = FALSE)
p_M2_stg2.noWt<-pt(T_M2_stg2.noWt, df2, lower.tail = FALSE, log.p = FALSE)
p_M3_stg2.noWt<-pt(T_M3_stg2.noWt, df2, lower.tail = FALSE, log.p = FALSE)

ps_stg2.noWt[,1]= c(p_M1_stg2.noWt,p_M2_stg2.noWt,p_M3_stg2.noWt,mvt.prob2.noWt)
Ts_stg2.noWt[,1] = c(T_m_stg2.noWt,T_max2.noWt)

#####
# IX. Evaluation of the MCP-Mod at every trial #
#####

# 1. Evaluate PoC establishment (PoA evaluation)

## Global Tests without Weights ##

# 1. combine p-values #
PoC_Gps.noWt[2,1]<-p_M1_stg1*p_M1_stg2.noWt
PoC_Gps.noWt[3,1]<-p_M2_stg1*p_M2_stg2.noWt
PoC_Gps.noWt[4,1]<-p_M3_stg1*p_M3_stg2.noWt
PoC_Gps.noWt[1,1]<-mvt.prob1*mvt.prob2.noWt

# 2. test global H0 by Fisher's comb. test of CEF. assign "1" if rejected. #
H0_resultG.noWt[1,2]<-(ifelse(PoC_Gps.noWt[2,1] < GH_Mi_critValp975L,1,0))
H0_resultG.noWt[1,3]<-(ifelse(PoC_Gps.noWt[3,1] < GH_Mi_critValp975L,1,0))
H0_resultG.noWt[1,4]<-(ifelse(PoC_Gps.noWt[4,1] < GH_Mi_critValp975L,1,0))
H0_resultG.noWt[1,1]<-(ifelse(PoC_Gps.noWt[1,1] < GH_Mi_critValp975L,1,0))

# Repeat for non weighting.
minP_G.noWt<-c(PoC_Gps.noWt[2,1],PoC_Gps.noWt[3,1],PoC_Gps.noWt[4,1])
subminP.noWt<-na.omit(minP_G.noWt)
attributes(subminP.noWt)<-NULL
H0_resultG.noWt[1,5] = which(minP_G.noWt==min(subminP.noWt))

```

```

    minP_G.noWt=H0_resultG.noWt[1,5]+1
    if(H0_resultG.noWt[1,minP_G.noWt] == 1){
      H0_resultG.noWt[1,6]=1
    }
} # This is the end of i loop (see #1#).
)

#####
# X. Evaluation summary of L trials.      #
#####

### MCP-Mod Hypothesis: no-response curve.
### =>If rejected, at least one of (M1, M2 M3) is significant.
### MVT based ###
## Stage 1 ##
PoA_stg1_MCPMod.MVT<-mean(H0_result1[,1])
PoA_stg1_MCPMod.MVT

## Stage 2 without Weights ##
PoA_stg2_ExMCPMod.MVT.noWt<-mean(H0_result2.noWt[,1])
PoA_stg2_ExMCPMod.MVT.noWt
## Global Test by Fisher's Combination Test ##
PoA_Global_ExMCPMod.MVT.noWt<-mean(H0_resultG.noWt[,1])
PoA_Global_ExMCPMod.MVT.noWt

### Summary of Results of MCP-Mod, the null hypotheses ###
PoA_result<-matrix(c(PoA_stg1_MCPMod.MVT,PoA_stg2_ExMCPMod.MVT.noWt,
  PoA_Global_ExMCPMod.MVT.noWt),nrow=3)

PoA_result

user.cpu<-simul.time[1]
system.cpu<-simul.time[2]
elapsed.time<-simul.time[3]
print(cbind(user.cpu,system.cpu,elapsed.time))

```

Note: The rest of program is omitted as the same coding is used for the other Emax, quadratic, and logistic data. For the different data-generation code, see Appendix [A.1](#).

#### A.4 DOSE CHOICES FOR POSITIVE DEFINITE MATRIX (CHAPTER 4)

```

library(mvtnorm)
require(graphics)
require(utils)

```

```

require(stats)
library(gplots) # for colorpanel()
library(lattice) # for levelplot()

### function to compute minimum eigen value for a placebo + 4 treatment doses.

get_eigen.k4 = function(len, s, sgn) {
  d <- seq(1/(len-1), 1.0, len = len-1)
  Nd<-length(d)
  data<-array(c(sgn*Inf),dim=c(Nd-3, Nd-3, Nd-3, Nd-3))

  for(u in 4:Nd){
    for(i in 3:(u-1)){
      for(j in 2:(i-1)){
        for(k in 1:(j-1)){
          d1<-(c(0, d[k], d[j], d[i], d[u]))
          cr1<-compute_cr1(d1, s)
          data[k, j-1, i-2, u-3]<-min(eigen(cr1)$values)
          #print(c(k, j, i, u))
        }}}
    return(data)
  }

### function to compute minimum eigen value for a placebo + 3 treatment doses.

get_eigen.k3 = function(len, s, sgn) {
  d <- seq(1/(len-1), 1.0, len = len-1)
  Nd<-length(d)
  data<-array(c(sgn*Inf),dim=c(Nd-2, Nd-2, Nd-2))

  for(i in 3:Nd){
    for(j in 2:(i-1)){
      for(k in 1:(j-1)){
        d1<-(c(0, d[k], d[j], d[i]))
        cr1<-compute_cr1(d1, s)
        data[k, j-1, i-2]<-min(eigen(cr1)$values)
        #print(c(k, j, i))
      }}}
    return(data)
  }

get_eigen.k3.log = function(len, s, sgn) {
  d <- seq(1/(len-1), 1.0, len = len-1)
  Nd<-length(d)
  data<-array(c(sgn*Inf),dim=c(Nd, Nd, Nd))

  for(i in 3:Nd){
    for(j in 2:(i-1)){
      for(k in 1:(j-1)){
        d1<-(c(0, d[k], d[j], d[i]))
        cr1<-compute_cr1(d1, s)
        data[k, j, i]<-log(abs(min(eigen(cr1)$values))), base=10)
        #print(c(k, j, i))
      }}}

```

```

    }}
    return(data)
}

### function to compute minimum eigen value for a placebo + 2 treatment doses.

get_eigen.k2 = function(len, s, sgn) {
  d <- seq(1/(len-1), 1.0, len = len-1)
  Nd<-length(d)
  data<-array(c(sgn*Inf),dim=c(Nd-1, Nd-1))

  for(j in 2:Nd){
    for(k in 1:(j-1)){
      d1<-(c(0, d[k], d[j]))
      cr1<-compute_cr1(d1, s)
      data[k, j-1]<-min(eigen(cr1)$values)
      #print(c(k, j))
    }
  }
  return(data)
}

### function to compute minimum eigen value for a placebo + 1 treatment doses.

get_eigen.k1 = function(len, s, sgn) {
  d <- seq(1/(len-1), 1.0, len = len-1)
  Nd<-length(d)
  data<-array(c(sgn*Inf),dim=c(Nd))

  for(k in 1:Nd){
    d1<-(c(0, d[k]))
    cr1<-compute_cr1(d1, s)
    data[k]<-min(eigen(cr1)$values)
    #print(c(k))
  }
  return(data)
}

### Model Specification, Standardizing Models ###

## 1. Get optimal regression coefficients (c_opt). ##
delta.max<-0.6
ED50<-0.2
alpha<-0.025
m<-3

compute_emax_c1_opt_m1 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### I. Standardized Emax Model.

  # First, write the standadized Emax model.
  f_m1_0<-function(d){

```

```

    d/(ED50+d)
  }
mu_m1.s1<-matrix(rep(0,kandP),ncol=1)
for(k in 1:kandP){
  d=doses_stg[k]
  mu_m1.s1[k]=f_m1_0(d)
}

# Compute c1_opt for model I in Stage 1.
mubar.m.s1<-function(mu){(sum(mu)/kandP)*Ident1.s1}
mudiff.m.s1<-function(mu,mubar.m.s1){mu-mubar.m.s1}
c1_opt_m<-function(mudiff.m.s1){
  mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1)%*%mudiff.m.s1)))
}
mubar.m1.s1<-mubar.m.s1(mu_m1.s1)
mudiff.m1.s1<-mudiff.m.s1(mu_m1.s1,mubar.m1.s1)
c1_opt_m1<-c1_opt_m(mudiff.m1.s1)
return(c1_opt_m1)
}

compute_quad_c1_opt_m2 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### II. Standardized Quadratic Model.

  # Write the standadized quadratic model.
  f_m2_0<-function(d){
    d-delta.max*d^2
  }
  mu_m2.s1<-rep(0,kandP)
  for(k in 1:kandP){
    d=doses_stg[k]
    mu_m2.s1[k]=f_m2_0(d)
  }

  # Define functions.
  mubar.m.s1<-function(mu){(sum(mu)/kandP)*Ident1.s1}
  mudiff.m.s1<-function(mu,mubar.m.s1){mu-mubar.m.s1}
  c1_opt_m<-function(mudiff.m.s1){
    mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1)%*%mudiff.m.s1)))
  }

  # Compute c1_opt for model II in Stage 1.
  mubar.m2.s1<-mubar.m.s1(mu_m2.s1)
  mudiff.m2.s1<-mudiff.m.s1(mu_m2.s1,mubar.m2.s1)
  c1_opt_m2<-c1_opt_m(mudiff.m2.s1)
  return(c1_opt_m2)
}

compute_logis_c1_opt_m3 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

```

```

### III. Standardized Logistic Model.

# First, write the standardized logistic model.
f_m3_0<-function(d){1/(1+exp((ED50-d)/delta.max))}

    mu_m3.s1<-rep(0,kandP)
      for(k in 1:kandP){
        d=doses_stg[k]
        mu_m3.s1[k]=f_m3_0(d)}

# Define functions.
mubar.m.s1<-function(mu){(sum(mu)/kandP)*Ident1.s1}
mudiff.m.s1<-function(mu,mubar.m.s1){mu-mubar.m.s1}
c1_opt_m<-function(mudiff.m.s1){mudiff.m.s1
  %*%(solve(sqrt(t(mudiff.m.s1)%*%mudiff.m.s1)))}

# Then compute c_ppt for model III in Stage 1.
mubar.m3.s1<-mubar.m.s1(mu_m3.s1)
mudiff.m3.s1<-mudiff.m.s1(mu_m3.s1,mubar.m3.s1)
c1_opt_m3<-c1_opt_m(mudiff.m3.s1)
return(c1_opt_m3)
}

compute_cr1 = function(doses_stg, samples){
  k1andP=length(doses_stg)
  n_stg1<-replicate(k1andP, samples)
  N1=sum(n_stg1)
  c1_opt_m1<-compute_emax_c1_opt_m1(k1andP, n_stg1, doses_stg)
  c1_opt_m2<-compute_quad_c1_opt_m2(k1andP, n_stg1, doses_stg)
  c1_opt_m3<-compute_logis_c1_opt_m3(k1andP, n_stg1, doses_stg)
  c1_opt_stg1<-t(matrix(c(c1_opt_m1, c1_opt_m2, c1_opt_m3),nrow=k1andP))

  ## 2. Find the critical value for PoC test H0 ###
  V1<-diag(1/n_stg1)
  df1<-N1-k1andP
  C1<-matrix(c1_opt_stg1, nrow=m)
  cv1<-C1%*%V1%*%t(C1)
  dv1<-t(1/sqrt(diag(cv1)))
  cr1<-cv1*(t(dv1)%*%dv1)
  return(cr1)
}

#####
### 1. Specify possible maximum doses ###
len<-10
s<-20

data<-get_eigen.k3.log(len, s, 1)

data<-get_eigen.k3(len, s, 1)
min(data)
ng.data<-get_eigen.k3(len, s, -1)

```

```

max(ng.data)

## build all
s<-20
N<-4
step<-5
min.ev.k1<-c()
max.ev.k1<-c()
min.ev.k2<-c()
max.ev.k2<-c()
min.ev.k3<-c()
max.ev.k3<-c()
min.ev.k4<-c()
max.ev.k4<-c()

for(i in 1:N) {min.ev.k1[i]<-min(get_eigen.k1(i*step, s, 1))}
for(i in 1:N) {max.ev.k1[i]<-max(get_eigen.k1(i*step, s, -1))}

for(i in 1:N) {min.ev.k2[i]<-min(get_eigen.k2(i*step, s, 1))}
for(i in 1:N) {max.ev.k2[i]<-max(get_eigen.k2(i*step, s, -1))}

for(i in 1:N) {min.ev.k3[i]<-min(get_eigen.k3(i*step, s, 1))}
for(i in 1:N) {max.ev.k3[i]<-max(get_eigen.k3(i*step, s, -1))}

for(i in 1:N) {min.ev.k4[i]<-min(get_eigen.k4(i*step, s, 1))}
for(i in 1:N) {max.ev.k4[i]<-max(get_eigen.k4(i*step, s, -1))}

min.ev.k1
max.ev.k1

min.ev.k2
max.ev.k2

min.ev.k3
max.ev.k3

min.ev.k4
max.ev.k4

#####
## counterexample: dose interval is fixed as 1.0/5=0.2 between doses.

# k=1 #
ev<-c()
for(i in 1:10) {
  # step=dose interval.
  step<-10^(-i)
  # placebo and 1 doses. <- A specific dose combination.
  d<-c(0, step)
  cr1<-compute_cr1(d, s)
  ev[i]<-min(eigen(cr1)$values)
}
ev

```

```

# k=2 #
ev<-c()
for(i in 1:10) {
  step<-10(-i)
  # placebo and 2 doses. <- A specific dose combination.
  d<-c(0, step, 2*step)
  cr1<-compute_cr1(d, s)
  ev[i]<-min(eigen(cr1)$values)
}
ev

# k=3 #
ev<-c()
for(i in 1:10) {
  step<-10(-i)
  # placebo and 3 doses. <- A specific dose combination.
  d<-c(0, step, 2*step, 3*step)
  cr1<-compute_cr1(d, s)
  ev[i]<-min(eigen(cr1)$values)
}
ev

# k=4 #
ev<-c()
for(i in 1:10) {
  step<-10(-i)
  d<-c(0, step, 2*step, 3*step, 4*step)
  # placebo and 4 doses. <- A specific dose combination.
  cr1<-compute_cr1(d, s)
  ev[i]<-min(eigen(cr1)$values)
}
ev

#### level plots ####

# Example I : Placebo and Three Treatment Doses #
# 20 potential treatment doses from 0.05 to 1.0.

len<-21
s<-20
# data for placebo and 3 doses.
data<-get_eigen.k3.log(len, s, 1)
# for axis labels.
d.level<-c(seq(1/(len-1), 1.0, len = len-1))

#### Each subject has Stage 1, Stage 2, and Global results in the dataset

# 1. Plot when the largest treatment dose, d3==1.0

levelplot(data[, ,20], scales=list(labels=d.level,at=(1:20)),

```

```

#scales=list(tck=0, x=list(rot=90)),
col.regions=colorpanel(20, "lightgray", "black"),
at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
main="Min(Eigenvalue) for Placebo and Three doses, d3=1.0",
sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the smallest e.v. of R for Placebo and 3 trt doses with d3=1.0.

# 2. Plot when the largest treatment dose, d3==0.95

levelplot(data[, ,19], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
col.regions=colorpanel(20, "lightgray", "black"),
at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
main="Min(Eigenvalue) for Placebo and Three doses, d3=0.95",
sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the smallest e.v. of R for Placebo and 3 trt doses with d3=0.95.

# 3. Plot when the largest treatment dose, d3==0.90

levelplot(data[, ,18], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
col.regions=colorpanel(20, "lightgray", "black"),
at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
main="Min(Eigenvalue) for Placebo and Three doses, d3=0.90",
sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the smallest e.v. of R for Placebo and 3 trt. doses with d3=0.90.

# 4. Plot when the largest treatment dose, d3==0.85

levelplot(data[, ,17], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
col.regions=colorpanel(20, "lightgray", "black"),
at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
main="Min(Eigenvalue) for Placebo and Three doses, d3=0.85",
sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.85.

# 5. Plot when the largest treatment dose, d3==0.80

levelplot(data[, ,16], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
col.regions=colorpanel(20, "lightgray", "black"),
at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
main="Min(Eigenvalue) for Placebo and Three doses, d3=0.80",
sub="with log10 scales", xlab="d1", ylab="d2")

```

```

### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.80.

# 6. Plot when the largest treatment dose, d3==0.75

levelplot(data[, ,15], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.75",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.75.

# 7. Plot when the largest treatment dose, d3==0.70

levelplot(data[, ,14], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.70",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.70.

# 8. Plot when the largest treatment dose, d3==0.65

levelplot(data[, ,13], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.65",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.65.

# 9. Plot when the largest treatment dose, d3==0.60

levelplot(data[, ,12], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.60",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.60.

# 10. Plot when the largest treatment dose, d3==0.55

levelplot(data[, ,11], scales=list(labels=d.level,at=(1:20)),

```

```

#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.55",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.55.

# 11. Plot when the largest treatment dose, d3==0.50

levelplot(data[,10], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.50",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.50.

# 12. Plot when the largest treatment dose, d3==0.45

levelplot(data[,9], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.45",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.45.

# 13. Plot when the largest treatment dose, d3==0.40

levelplot(data[,8], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.40",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.40.

# 14. Plot when the largest treatment dose, d3==0.35

levelplot(data[,7], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
      -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.35",
  sub="with log10 scales", xlab="d1", ylab="d2")

```

```

### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.35.

# 15. Plot when the largest treatment dose, d3==0.30
levelplot(data[,6], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
    -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.30",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.30.

# 16. Plot when the largest treatment dose, d3==0.25
levelplot(data[,5], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
    -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.25",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.25.

# 17. Plot when the largest treatment dose, d3==0.20
levelplot(data[,4], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
    -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.20",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.20.

# 18. Plot when the largest treatment dose, d3==0.15
levelplot(data[,3], scales=list(labels=d.level,at=(1:20)),
#scales=list(tck=0, x=list(rot=90)),
  col.regions=colorpanel(20, "lightgray", "black"),
  at=c(-6,-5.75,-5.5,-5.25,-5,-4.75,-4.5,-4.25,-4,-3.75,
    -3.5,-3.25,-3,-2.75,-2.5,-2.25,-2,-1.75,-1.5,-1.25,-1),
  main="Min(Eigenvalue) for Placebo and Three doses, d3=0.15",
  sub="with log10 scales", xlab="d1", ylab="d2")
### Log10 of the Smallest e.v. of R for Placebo and 3 trt. doses with d3=0.15.

```

## A.5 WEIGHTED CORRELATION MATRICES USING DIFFERENT NUMBERS OF DOSES (CHAPTER 4)

```

library(partitions)
library(gregmisc)

#####
# Helper Functions
#####

# function to get rid of all-zero rows.
fun2 = function(MATRIX) {
  P = nrow(MATRIX);
  NA.row = rep(NA,P) ;
  for (i in 1:P) {
    if ((is.na(sum(MATRIX[i,]))) || (sum(MATRIX[i,])) == 0) {
      NA.row[i] = -i}
    }
    if(length(na.exclude(NA.row))>=1){new = MATRIX[na.exclude(NA.row), ]
  }
  else { new=MATRIX }
  return(new)
}

# function to get rid of 0 value in partition candidates.

dropZero<-function(v) {
  r<-c()
  for (i in v)
    if (i!=0)
      r<-c(r,i)
  return(r)
}

# function only to leave non-duplicate partition sets.

uniqueAdd<-function(parts, p) {
  if (length(parts) == 0)
    parts<-p
  else
  {
    pm<-t(matrix(parts, nrow=length(p)))
    found<-FALSE

    for (i in 1:length(pm[,1]))
      if (isTRUE(all.equal(pm[i,]== p, rep(TRUE, length(p)))))
        found<-TRUE
    if (!found)
      parts<-c(parts, p)
  }
}

```

```

    }
    return(parts)
}

# function to simulate all partition candidates given
# 'the number of results in Stage 1' (=val) and
# 'the number of adaptation choices of Z' (=num).

buildParts<-function(val, num) {
  candidates<-parts(val)
  perms<-permutations(n=num, r=num)
  allParts<-c()

  for (i in 1:length(candidates[1,])) {
    part<-dropZero(candidates[1:num, i])
    if ((sum(part) == val) && (length(part) == num)) {
      for (j in 1:length(perms[,1])) {
        u<-c()
        for (k in 1:num) {
          u[k]<-part[perms[j,k]]
        }
        allParts<-uniqueAdd(allParts, u)
      }
    }
  }

  partMat<-matrix(data=allParts, nrow=num)
  return(partMat)
}

# function to make a design matrix Z consisting of L rows
# (=number of adaptation choices g);
# a sub-matrix of the full design matrix full.Z.

subMat<-function(full.Mat, gs) {
  Mat.L<-length(gs)
  Mat<-c() # rth row vector of 'Mat' matrix.
  for (i in 1:Mat.L) {
    Mat<-c(Mat, full.Mat[gs[i],])
  }
  return(t(matrix(Mat, ncol=Mat.L)))
}

# function to build a full design matrix full.Z

build.full.Z<-function(n, p.ds) {
  # number of vector containing all potential doses.
  orig<-1

  # number of vectors allowing 1 dose dropped
  # from the set of potential doses.

```

```

drop.1<-length(combinations(4,1)[,1])

# number of vectors allowing 2 doses dropped
# from the set of potential doses.
drop.2<-length(combinations(4,2)[,1])

if (n==1) {
  full.Z<-matrix(c(1), nrow = orig+drop.1, ncol = p.ds)
  # all possible design matrix Z's of full. Z.
  for (i in 1:drop.1){
    full.Z[orig+i,i+1]=0
  }
}
if (n==2) {
  # how to drop 2 doses out of 4 potentials.
  cdp2<-combinations(4,2)
  full.Z<-matrix(c(1), nrow =orig+drop.1+drop.2, ncol = p.ds)
  for (i in 1:drop.1){
    full.Z[orig+i,i]=0
  }
  for (i in 1:drop.2){
    full.Z[orig+drop.1+i,1+cdp2[i,1]]=0 # 1st dose to drop.
    full.Z[orig+drop.1+i,1+cdp2[i,2]]=0 # 2nd dose to drop.
  }
}
return(full.Z)
}

#####
# End: Helper Functgions
#####

#####
# Models and unweighted contrast vectors
#####

### Model Specification, Standardizing Models, and Empty Matrices for Evaluation ###

## 1. Get optimal regression coefficients (c_opt). ##
delta.max<-0.6
ED50<-0.2
alpha<-0.025
m<-3

compute_emax_c1_opt_m1 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

### I. Standardized Emax Model.
# First, write the standadized Emax model.
f_m1_0<-function(d){d/(ED50+d)}

```

```

mu_m1.s1<-matrix(rep(0,kandP),ncol=1)
for(k in 1:kandP){
  d=doses_stg[k]
  mu_m1.s1[k]=f_m1_0(d)
}
# Compute c1_opt for model I in Stage 1.
# Define functions.
mubar.m.s1<-function(mu){
  (sum(mu)/kandP)*Ident1.s1
}
mudiff.m.s1<-function(mu,mubar.m.s1){
  mu-mubar.m.s1
}
c1_opt_m<-function(mudiff.m.s1){
  mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1)%*%mudiff.m.s1)))}
mubar.m1.s1<-mubar.m.s1(mu_m1.s1)
mudiff.m1.s1<-mudiff.m.s1(mu_m1.s1,mubar.m1.s1)
c1_opt_m1<-c1_opt_m(mudiff.m1.s1)
return(c1_opt_m1)
}

compute_quad_c1_opt_m2 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### II. Standardized quadratic Model.
  # Write the standadized quadratic model.
  f_m2_0<-function(d){
    d-delta.max*d^2
  }
  mu_m2.s1<-rep(0,kandP)
  for(k in 1:kandP){
    d=doses_stg[k]
    mu_m2.s1[k]=f_m2_0(d)
  }

  mubar.m.s1<-function(mu){(sum(mu)/kandP)*Ident1.s1}
  mudiff.m.s1<-function(mu,mubar.m.s1){mu-mubar.m.s1}
  c1_opt_m<-function(mudiff.m.s1){
    mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1)%*%mudiff.m.s1)))
  }
  # Compute c1_opt for model II in Stage 1.
  mubar.m2.s1<-mubar.m.s1(mu_m2.s1)
  mudiff.m2.s1<-mudiff.m.s1(mu_m2.s1,mubar.m2.s1)
  c1_opt_m2<-c1_opt_m(mudiff.m2.s1)
  return(c1_opt_m2)
}

compute_logis_c1_opt_m3 = function(kandP, n_stg, doses_stg) {
  Ident1.s1<-matrix(rep(1,kandP),ncol=1)
  Ident2.s1<-matrix(rep(1,n_stg[1]),ncol=1)

  ### III. Standardized Logistic Model.
  # Write the standardized logistic model.

```

```

f_m3_0<-function(d){
  1/(1+exp((ED50-d)/delta.max))
}
mu_m3.s1<-rep(0,kandP)
for(k in 1:kandP){
  d=doses_stg[k]
  mu_m3.s1[k]=f_m3_0(d)
}
mubar.m.s1<-function(mu){
  (sum(mu)/kandP)*Ident1.s1
}
mudiff.m.s1<-function(mu,mubar.m.s1){
  mu-mubar.m.s1
}
c1_opt_m<-function(mudiff.m.s1){
  mudiff.m.s1%%(solve(sqrt(t(mudiff.m.s1)*%*%mudiff.m.s1)))
}
# Compute c_ppt for model III in Stage 1.
mubar.m3.s1<-mubar.m.s1(mu_m3.s1)
mudiff.m3.s1<-mudiff.m.s1(mu_m3.s1,mubar.m3.s1)
c1_opt_m3<-c1_opt_m(mudiff.m3.s1)
return(c1_opt_m3)
}

#=====
# END: Models and unweighted contrast vectors
#=====

#=====
# Compute weight matrix
#=====
# for given;
# Z: Adaptation design matrix
# adpt.a: Partition vector where the number of PoC results in Stage 1
# are partitioned into L adaptations of Z
# g: Adaptation choice defined in Z (g=1,...,L)

W_a<-function(g, adpt.a, tZ) {
  W_g<-adpt.a[g] * tZ[,g,drop=F] %% c(1,1,1)
  # We fix a1=1 in W_a(g,a1) of this simulation. --
  # all probabilities for testing model m is set to be 1 regardless of its
  # testing result in Stage 1. if that changes, the code needs to be rewritten.
  return(W_g)
}

computeWg<-function(g, adpt.a, Z) {
  tZ<-t(Z)
  gmax<-length(adpt.a) #equals rows(Z)

  #W.denom<-W_a(1,1)+W_a(2,4)+W_a(3,7)+W_a(4,3)
  W.denom<-0
  for (i in 1:gmax) {
    W.denom<-W.denom + W_a(i, adpt.a, tZ)
  }
}

```

```

}
W_g<-W_a(g, adpt.a, tZ)*(1/W.denom)
# W_g<-fun2(W_g)

return (W_g)
}

dropDoses<-function(Wg, d) {
  for (i in 1:length(Wg[,1])) {
    if ((is.na(sum(Wg[i,]))) || (sum(Wg[i,]) == 0)) {
      d[i]<-NA
    }
  }
  return(c(na.exclude(d)))
}

droppedWg<-function(Wg) {
  return(fun2(Wg))
}

#=====
# END: Compute weight matrix
#=====

#=====
# Compute weighted correlation matrix
#=====

#=====
# compute the correlation matrix from
# dose vector: doses_stg
# samples: samples
# weight matrix: W_g

compute_wt_cr2 = function(doses_stg, samples, W_g){
  # k + 1 in Stage 2 (# of treatment doses and the placebo).
  k2andP=length(doses_stg)
  n_stg2<-replicate(k2andP, samples)
  N2=sum(n_stg2)

  # get unweighted contrasts
  c2_opt_m1<-compute_emax_c1_opt_m1(k2andP, n_stg2, doses_stg)
  c2_opt_m2<-compute_quad_c1_opt_m2(k2andP, n_stg2, doses_stg)
  c2_opt_m3<-compute_logis_c1_opt_m3(k2andP, n_stg2, doses_stg)

  # add weighting
  Wc2_opt_m1<-W_g[,1]*c2_opt_m1
  Wc2_opt_m2<-W_g[,2]*c2_opt_m2
  Wc2_opt_m3<-W_g[,3]*c2_opt_m3

  # build matrix
  Wc2_opt<-t(matrix(c(Wc2_opt_m1, Wc2_opt_m2, Wc2_opt_m3),nrow=k2andP))

  ## 2. Find the critical value for PoC test H0 ###

```

```

V2<-diag(1/n_stg2)
df2<-N2-k2andP
C2<-matrix(Wc2_opt, nrow=m)# under H0.
cv2<-C2%*%V2%*%t(C2)
dv2<-t(1/sqrt(diag(cv2)))
cr2<-cv2*(t(dv2)%*%dv2) # this is the correlation matrix in H0.

return(cr2)
}

#=====
# END: Compute weighted correlatrion matrix
#=====
#=====
# SCRIPT: Going through (traverse) space and evaluate
#=====

#=====
# compute weight matrices for all choices

step<-0.05
doses<-c(0, step, 2*step, 3*step, 4*step)
#doses<-c(0, 0.2, 0.5, 0.8, 1.0)
samples<-20
# the number of PoC results in Stage 1 given the number of candidate dose-response
# models is m.
a.stg1<-2^m

# placebo and candidate doses.
poten.doses<-5
max.L<-5 # the number of all gs of full.Z.

# full Z matrix.
full.Z<-matrix(c(1), nrow = max.L, ncol = poten.doses)
for (i in 2:max.L){
  full.Z[i,i]=0
}

#=====

#simple test case first
#fullWg<-computeWg(g, adpt.a, Z)
#doses2<-dropDoses(fullWg, doses)
#Wg<-droppedWg(fullWg)
#Wcr2<-compute_wt_cr2(doses2, samples, Wg)
#mev2<-min(eigen(Wcr2)$values)
#sum.fullWg<-matrix(c(0), ncol=3 ,nrow=adpt.L)

#now the full experiment.

getZandEvs<-function(doses){
  ZandMEv<-list()
  minEv<-c()

```

```

#for (num.g in 1:max.L) {
for (num.g in 1:min(max.L,a.stg1)) {
  combs<-combinations(max.L, num.g)
  # the number of k dose combinations where 'k = # of doses to use'.
  num.comb.g<-length(combs[,1])
  mevs2<-c()
  for (j in 1:num.comb.g) {
    Z<-subMat(full.Z, combs[j,])
    # all partitions (related to ADTAR rule mapping result a1
    # (in a.stg1) to g (in adpt.L of design matrix Z))
    adpt.L<-num.g
    p<-buildParts(a.stg1, adpt.L)
    for (i in 1:length(p[1,])) {
      adpt.a<-p[,i]
      for (g in 1:length(adpt.a)) {
        fullWg<-computeWg(g, adpt.a, Z)
        # sum.fullWg<-sum + fullWg
        doses2<-dropDoses(fullWg, doses)
        Wg<-droppedWg(fullWg)
        Wcr2<-compute_wt_cr2(doses2, samples, Wg)
        mev2<-min(eigen(Wcr2)$values)
        mevs2<-c(mevs2, mev2)
      }
    }
  }
  ZandMEv[[num.g]]<-mevs2
  minEv<-c(minEv,mevs2)
}
ZandMEv$minMEv<-min(minEv)
return(ZandMEv)
}

# one trial with the following set of doses.
#doses<-c(0, 0.2,0.5,0.7,1.0)
#print(doses)
#ZandMEv<-getZandEvs(doses)

# the minimum of min(eigenvalue) of WtR matrices
# over all possible Z matrices given a set (doses).
#print(ZandMEv$minMEv)

# list the minimums of min(eigenvalue) of WtR matrices for each length L
# of design matrix Z_(L x k2andP) given a set of potential doses (doses).
#for(num.g in 1:max.L){
#  print(num.g)
#  print(min(ZandMEv[[num.g]]))
#}

# all min(eigenvalue) of WtR matrices sorted by the length L of design
# matrix Z and min.minEV given a set (doses).
#print(ZandMEv)

#sum.fullWg

```

```

### compute minimum eigen value of weighted Rs for various sets of
# placebo + 4 potential doses.

all_doses.k4 = function(len) {
  len<-len+1
  d <- seq(1/(len-1), 1.0, len = len-1)
  print(c(0,d))
  Nd<-length(d)
  # data<-array(c(Inf),dim=c(Nd-3, Nd-3, Nd-3, Nd-3))
  data<-array(c(Inf),dim=c(Nd, Nd, Nd, Nd))

  for(u in 4:Nd){
    for(i in 3:(u-1)){
      for(j in 2:(i-1)){
        for(k in 1:(j-1)){
          #print(c(k, j, i, u))
          doses<-(c(0, d[k], d[j], d[i], d[u]))
          print(doses)
          ZandMEv<-getZandEvs(doses)
          minEv<-ZandMEv$minMEv
          print(minEv)
          data[k, j, i, u]<-minEv
          #data[k, j-1, i-2, u-3]<-minEv
        }}}
    return(data)
  }

# min.eigens allowing 1 dose to be dropped (dose step size=5).
result.stp5.drp1<-all_doses.k4(5)
min(result.stp5.drp1)
save(full.Z, result.stp5.drp1,
      file = "//10.0.0.3/yoko's data/R_Code/2ndPaperR/MinEvs_ADTAR_Drp1d_05Jul10.Rdata")

# min.eigens allowing 1 dose to be dropped (dose step size=10).
result.stp10.drp1<-all_doses.k4(10)
min(result.stp10.drp1)
save(full.Z, result.stp5.drp1, result.stp10.drp1,
      file = "//10.0.0.3/yoko's data/R_Code/2ndPaperR/MinEvs_ADTAR_Drp1d_05Jul10.Rdata")

# min.eigens allowing 1 dose to be dropped (dose step size=20).
result.stp20.drp1<-all_doses.k4(20)
min(result.stp20.drp1)

# Store the results #
save(full.Z, result.stp5.drp1, result.stp10.drp1, result.stp20.drp1,
      file = "//10.0.0.3/yoko's data/R_Code/2ndPaperR/MinEvs_ADTAR_Drp1d_05Jul10.Rdata")

#=====
### NEXT EXPERIMENT: allows 1 or 2 doses excluded by ADTAR.

#simple test case first.

# one row drops two doses in the current full.Z (row length =5).

```

```

full.Z[2,3]=0

# min.eigens allowing 1 or 2 doses to be dropped (dose step size=10).
case1.stp10.drp2<-all_doses.k4(10)
min(case1.stp10.drp2)

# Store the results #
save(full.Z, case1.stp10.drp2,
file = "//10.0.0.3/yoko's data/R_Code/2ndPaperR/MinEvs_ADTAR_Drp2d_Case1_05Jul10.Rdata")

#now the full experiment.

# make a new full.Z.
# placebo and candidate doses (potential doses).
poten.doses<-5
# contains all the sets of doses where 'up to 2 doses' are dropped
# from 5 potentials.
full.Z<-build.full.Z(2,poten.doses)
# the number of all gs of full.Z.
max.L<-length(full.Z[,1]) # max.L<-orig+drop.1+drop.2

# I. generate all possible design matrix Z's for the new full Z and
# get min.eigens allowing 1 dose or 2 doses to be dropped (dose step size=5).
result.stp5.drp2<-all_doses.k4(5)
min(result.stp5.drp2)

# Store the results #
save(full.Z, result.stp5.drp2,
file = "//10.0.0.3/yoko's data/R_Code/2ndPaperR/MinEvs_ADTAR_Drp2d_11Jul10.Rdata")

# II. generate all possible design matrix Z's for the new full Z and
# get min.eigens allowing 1 dose or 2 doses to be dropped (dose step size=10).
result.stp10.drp2<-all_doses.k4(10)
min(result.stp10.drp2)

# Store the results #
save(full.Z, result.stp5.drp2, result.stp10.drp2,
file = "//10.0.0.3/yoko's data/R_Code/2ndPaperR/MinEvs_ADTAR_Drp2d_11Jul10.Rdata")

# III. generate all possible design matrix Z's for the new full Z and
# get min.eigens allowing 1 dose or 2 doses to be dropped (dose step size=20).
result.stp20.drp2<-all_doses.k4(20)
min(result.stp20.drp2)

# Store the results #
save(full.Z, result.stp5.drp2, result.stp10.drp2, result.stp20.drp2,
file = "//10.0.0.3/yoko's data/R_Code/2ndPaperR/MinEvs_ADTAR_Drp2d_11Jul10.Rdata")

```

## BIBLIOGRAPHY

- [1] Bauer, P. and Kieser, M. Combining different phases in the development of medical treatments within a single trial. *Statistics in Medicine*, 18:1833-1848, 1999.
- [2] Bauer, P. and Köhne, K. Evaluation of experiments with adaptive interim analysis. *Biometrics*, 50:1029-1041, 1994.
- [3] Berry, DA., Müller, P., Grieve, AP., Smith, M., Parke, T., Balazek, R., Mitchard, N., and Krams, M. Adaptive Bayesian designs for dose-ranging trials. In Gatsonis C., Kass, RE., Carlin B, Carriquiry A., Gelman A., Verdinelli, I., and West, M. *Case Studies in Bayesian Statistics V*. Springer, Verlag, New York, 2002.
- [4] Bornkamp, B., Bretz, F., Dmitrienko, A., Enas, G., Gaydos, B, Hsu, C., König, F., Kranms, M., Liu, Q., Neuenschwander, B., Parke, T., and Pinheiro, J. Innovative approaches for designing and analyzing adaptive dose-ranging trials. *Journal of Biopharmaceutical Statistics*, 17:969-995, 2007.
- [5] Branson, M., Pinheiro, J., and Bretz, F. Searching for an adequate dose: combining multiple comparisons and modeling techniques in dose-response studies. *Novartis Biometric Technical Report*, Number 2003-08-20, 2003.
- [6] Bretz, F., Pinheiro, J. C., and Branson, M. Combining multiple comparisons and modeling techniques in dose-response studies. *Biometrics*, 61:738-748, 2005.
- [7] Bretz, F., Schmidli, H., König, F., Racine, A., and Maurer, W. Confirmatory seamless phase II/III clinical trials with hypotheses selection at interim: general concepts. *Biometrical Journal*, 48(4):623-634, 2006.
- [8] Chang, M., Kenley, S., Bull, J., Chiu, Y., Wang, W., Wakeford, C., and McCarthy, K. Innovative approaches in drug development. *Journal of Biopharmaceutical Statistics*, 17:775-789, 2007.
- [9] Derendorf, H. and Meibohm, B. Modeling of pharmacokinetic/pharmacodynamic (PK/PD) relationships: concepts and perspectives. *Pharmaceutical Research*, 16(2):10, 1999.

- [10] Dmitrienko, A., Gaydos, B., and Dragalin, V. Fall 2007 web-based training series. Webinar 1. Adaptive designs in clinical trials: Introduction, terminology, and classification. *Biopharmaceutical Section of ASA*, 2007.
- [11] Dragalin, V. Adaptive designs: terminology and classification. *Drug Information Journal*, 40:425-435, 2006.
- [12] Dragalin, V. and Fedorov, V. Adaptive designs for dose-finding based on efficacy-toxicity response. *Journal of Statistical Planning and Inference*, 136:1800-1823, 2006.
- [13] FDA. *Guidance for Industry: Adaptive Design Clinical Trials for Drugs and Biologics (Draft Guideline)*, 2010.
- [14] Gallo, P., Chuang-Stein, C., Dragalin, V., Gaydos, B., Krams, M., and Pinheiro, J. Adaptive designs in clinical drug development - an executive summary of the PhRMA working group. *Journal of Biopharmaceutical Statistics*, 16:275-283, 2006.
- [15] Gaydos, B., Dragalin, V., and Coffey, C. Fall 2007 web-based training series. Webinar 2. Adaptive designs in dose-finding studies. *Biopharmaceutical Section of ASA*, 2007.
- [16] Golub, H. L. The need for more efficient trial designs. *Statistics in Medicine*, 25:3231-3235, 2006.
- [17] Gilbert, G.T. Positive definite matrices and Sylvester's criterion *The American Mathematical Monthly* 98(1):44-46, 1991.
- [18] Hommel, G. Adaptive modifications of hypotheses after an interim analysis. *Biometrical Journal*, 5:581-589, 2001.
- [19] Hung, H. M. J., O'Neill, R. T., Wang, S-J., and Lawrence, J. A regulatory view on adaptive/flexible clinical trial design. *Biometrical Journal*, 48(4):565-573, 2006.
- [20] Hung, H. M., Wang, S., and O'Neill, R. Methodological issues with adaptation of clinical trial design. *Pharmaceutical Statistics*, 5:99-107, 2006.
- [21] ICH. Dose-reponse information to support drug registration, Topic E4. *ICH Harmonized Tripartite Guideline*, 1994.
- [22] ICH. Guideline for good clinical practices, Topic E6. *ICH Harmonized Tripartite Guideline*, 1996.
- [23] IEEE. *The IEEE Standard for Floating-Point Arithmetic (IEEE 754)*, 2008.
- [24] Koyama, T., Sampson, A., and Gleser, L. J. A calculus for design of two-stage adaptive procedure. *Journal of the American Statistical Association*, 100(469):197-203, 2005.
- [25] Little, R.J.A. and Rubin, D.B. *Statistical Analysis with Missing Data*, pages 41-53, Wiley Series in Probability and Statistics, Hoboken, New Jersey, 2nd edition, 2002.

- [26] Marcus, M. and Minc, H. *Introduction to Linear Algebra*, 182, Dover Publications Inc., Mineola, New York, 1988.
- [27] Meibohm, B. and Derendorf, D. Basic concepts of pharmacokinetic/pharmacodynamic (PK/PD) modeling. *International Journal of Clinical Pharmacology and Therapeutics*, 35(10):401-413, 1997.
- [28] Phillips, A. J. and Keene, O. N. Adaptive designs for pivotal trials: discussion points from the PSI adaptive design expert group. *Pharmaceutical Statistics*, 5:61-66, 2006.
- [29] Pinheiro, J., Bornkamp, B., and Bretz, F. Design and analysis of dose-finding studies combining multiple comparisons and modeling procedures. *Journal of Biopharmaceutical Statistics*, 16:639-656, 2006.
- [30] Ruberg, S. J. Dose response studies. I. Some design considerations. *Journal of Biopharmaceutical Statistics*, 5(1):1-14, 1995.