Optimal Physical Experiments and Surrogates for Complex Computer Models

by

Jeevan Prakash Jankar

(Under the Direction of Abhyuday Mandal)

Abstract

This dissertation addresses two fundamental areas in statistics: the design of optimal physical experiments and the development of surrogate models for complex computer experiments, with a focus on feature importance and uncertainty quantification.

In the first part, we investigate locally D-optimal crossover designs for generalized linear models. Model parameters and their variances are estimated using generalized estimating equations (GEEs). We identify optimal allocations of experimental units across treatment sequences and demonstrate through simulations that these allocations are reasonably robust to various choices of the correlation structure. Furthermore, we show that a two-stage design—employing our locally D-optimal design in the second stage—yields greater efficiency than a uniform design, particularly in the presence of intra-subject correlation.

The second part of the dissertation extends the principles of Design of Experiments (DoE) to improve reinforcement learning techniques for computer experiments (CEs), which are essential tools for studying phenomena where physical experimentation is infeasible, such as the spread of COVID-19. Building accurate computer models often involves a high-dimensional input space, making the identification of active variables critical. We propose a novel variable selection approach integrated with Active Learning for Lasso Regression, using a weighted distance function to sequentially guide variable selection. Additionally, we introduce a multi-objective optimization framework to construct efficient Sequential MaxPro Designs and Sequential Orthogonal-MaxPro Designs. Finally, we explore an extension of this reinforcement learning framework to Deep Gaussian Process (DGP) models, enabling more flexible modeling and a deeper understanding of feature importance under uncertainty.

INDEX WORDS: crossover Designs, *D*-optimality, feature-importance, uncertainty quantification, active learning, deep Gaussian process

Optimal Physical Experiments and Surrogates for Complex Computer Models

by

Jeevan Prakash Jankar

M.S., University of Georgia, 2022

A Dissertation Submitted to the Graduate Faculty of the
University of Georgia in Partial Fulfillment of the Requirements for the
Degree.

Doctor of Philosophy

Athens, Georgia

2025

Optimal Physical Experiments and Surrogates for Complex Computer Models

by

Jeevan Prakash Jankar

Major Professor:	Abhyuday Mandal
Committee:	Qian Xiao
	Gauri Datta
	Yuan Ke

Electronic Version Approved:

Ron Walcott
Dean of the Graduate School
The University of Georgia
May 2025

DEDICATION

To all the students who had to discontinue their PhD because of toxic work environments or other current issues, and to all the kind and humble researchers who are striving to make academia a better place.

Acknowledgments

I have been unimaginably fortunate, perhaps even unfairly so, to have had the support of many friends, colleagues, mentors, and role models throughout my PhD. What follows is a deeply abridged acknowledgment of their invaluable roles, both in shaping this thesis and in shaping me as a researcher and person.

When I joined Dr. Abhyuday Mandal's group, like many new PhD students, I was filled with both excitement and self-doubt. But Abhyuday Da welcomed me with warmth and optimism, and together we worked toward finding a research direction I genuinely enjoyed. Our first collaboration opened my eyes to just how thrilling research can be, and the memories of that experience have carried me through the many ups and downs of graduate school. At the end of my second year, we began collaborating with Dr. Qian Xiao, who introduced me to the world of computer experiments and uncertainty quantification. His guidance was pivotal in shaping my theoretical interests while grounding them in practical relevance. Both of my advisors have invested immense time and effort in teaching me how to organize ideas and communicate them with clarity. Thanks to their mentorship, I've come to find joy not just in doing research, but also in teaching and in making technical ideas accessible to others.

This thesis would not have been possible without the valuable feedback, insightful suggestions, and genuine enthusiasm of my other two committee members, Dr. Gauri Datta and Dr. Yuan Ke. I'm truly grateful to have had the pleasure of working with both of them on my committee.

Next, I would like to thank all my collaborators, who have broadened my perspective on how to choose research problems and made the research process far more lively and enjoyable. My heartfelt thanks go first to Dr. Jie Yang, with whom I had the pleasure of working on my first two research projects during my PhD. I'm also grateful to my group members, Hongzhi Wang and Xiaotian Zhang, for their helpful discussions and valuable research ideas. I feel incredibly fortunate to have learned from Abhyuday Da, Dr. Gauri Datta, Dr. Ahn, Dr. Cheolwoo Park, Dr. Rai Bai, and Dr. Pengsheng Ji. Their well-designed and

thoughtfully taught courses in statistics and mathematics laid a strong foundation for all of my research at UGA.

Many other researchers have generously taken the time to offer valuable advice that has helped shape my research career, especially during the final stages of my PhD. I would like to extend my heartfelt thanks to all of them, including Dr. T.N. Sriram, Dr. Liang Liu, Dr. Ting Zhang, Dr. Robert B. Gramacy, and Dr. Aaditya Ramdas (with sincere apologies for this inevitably incomplete list). I am especially grateful to Dr. Sriram and Dr. Liu for their consistent support and guidance throughout my PhD.

Let me now rewind a bit further. In the early years of my PhD, I deeply benefited from the guidance and shared experiences of fellow researchers, especially students, who generously offered their time and advice. Will Cranford, Subham Das, Jialin Yang, Hongzhi Wang, Jiacheng Li, and Greg Ellison all played a part in helping me navigate some of the toughest moments of my journey as a junior PhD student. I'm especially indebted to them for their moral support and encouragement. A special thanks to Subham for his guidance during some of the challenges I faced within the department.

This acknowledgment would be incomplete without expressing my sincerest thanks to my undergraduate thesis advisor, Saugata Bandyopadhyay Sir, for introducing me to research and encouraging me to apply to graduate school. I had the privilege of taking several of his courses in measure theory and functional analysis, which instilled in me a lasting fascination with the role of measure theory in statistics. He also introduced me to theoretical problems that continue to inspire my research in statistical learning theory to this day.

If you have come this far, thanks to you too! Remember to hydrate yourself.

On a more personal note, I've been incredibly fortunate to be surrounded by a wonderful group of thoughtful and talented friends in Athens who have become my family and my fish bowl over the years: Ananta Da, Aditi & Sean, Buket, Freeman, Hari & Maddy (H&M), Kuhelika, Prachi, Samyam, Sneha, Subham, and Twinkle. You can call it an aquarium, honestly. Their presence has made this journey lighter, brighter, and far more joyful.

I've had countless philosophical debates and conversations with them that have helped me discover and shape my beliefs and values over the years. These exchanges have also sharpened my ability to articulate thoughts clearly, something that has, unsurprisingly, been invaluable in research. My friends have always had their doors open (quite literally), even during the most ungodly hours, whenever I needed a break, a distraction, or simply someone to pour my heart out to.

I'm deeply indebted to Hari and Maddy for the many, many times they invited me over, or graciously allowed me to invite myself, especially during some of the roughest patches of my PhD. Special thanks to Maddy for introducing me to running and helping me complete my first half and full marathons. I'm grateful for our deep conversations about life, the much-needed yoga sessions and coffee breaks, and of course, for keeping me well-fed with her expertly baked breads, desserts, and cakes. Thanks to Hari for expanding my musical tastes and for making me watch all those hilariously ridiculous movies I never would have discovered on my own. I want to thank Aditi for fueling me with gallons of chai, for all the poha sessions, and for making vada-pav whenever I had a craving. I'm especially grateful to her for introducing me to climbing, which has since become an integral part of my life. Our thought-provoking conversations have helped me learn so much about myself, and her unwavering enthusiasm in listening has meant more to me than words can express. Special thanks to her for organizing Diwali celebrations every year and being my family whenever I needed one.

Thanks to Ananta, who has been like a big brother to me. Without him, I would have been both homeless and malnourished during the early years of my PhD. I'm especially thankful for all the delicious Bengali food and for deepening my love for fish. I'll always appreciate his tireless driving and for taking me on so many memorable hikes and road trips across the East Coast. And thank you to Twinkle, one of my closest friends since day one in Athens, for all the weekend nights we stayed up partying our hearts out. Lastly, a big thank-you to both Ananta and Twinkle for being such cooperative, caring, and responsible roommates. I couldn't have asked for better.

Thanks to Freeman for entertaining us with qawwali and guitar sessions, for inviting me to his concerts, and for enriching my life with his eclectic sense of humor. Thanks to Sean for introducing me to disc golf, for all the pool games, and for taking me to some of the best taco spots in Athens, not to mention providing a steady stream of free pop culture education. I'm grateful to Kuhelika for patiently listening to my endless complaints, always offering practical and honest advice. She has continually challenged me to be a more socially conscious person, and her care, insight, and relatability have meant a lot. And yes, thank you for all the spontaneous samosa runs and for making chai whenever I needed it most. Thanks to Samyam for all the tasty momo nights and for helping me train for marathons.

There are friends outside of this group who have played equally important roles along the way. I want to begin by expressing my deepest gratitude to Onkar and Prakash for their unwavering effort to stay in touch, often through sponta-

neous, hour-long calls that became a lifeline throughout my PhD. Their emotional and intellectual support, along with their thoughtful feedback, helped me navigate some of the most important decisions of this journey. Thanks to Alex for her constant support, love, especially during the final month of my PhD. I'm grateful for our honest and open friendship, where we could be both kids and adults at the same time. Despite the short time we have known each other, we have created many wonderful memories. Thank you also for your insightful feedback on this thesis, for always being there to talk to, and for all the hilarious side conversations. And thank you to Tim for being my partner in crime and for all the incredible memories we created during that unforgettable month at IMSI in Chicago.

Words cannot adequately capture the depth of my gratitude to my fiancée, Malavika (an incredible researcher in her own right, and someone I'm immensely proud of) for her unwavering companionship. She has brought the calm, maturity, and level-headedness I needed to navigate the challenges of a PhD and of life itself. Throughout this journey, she has shown extraordinary patience and kindness. This thesis quite literally would not have been possible without her tireless care, meticulously looking after me as if I were a fragile houseplant. Thank you also for taking care of our fur baby, Queso, whenever I couldn't. Together, they have given me more love and emotional support than I could ever express, something that sustained me through every step of this PhD.

Finally, I want to thank my family, my mom, dad, and brother, for their unwavering love and blessings. I'm especially grateful to my parents for recognizing the value of education and doing everything they could to provide us with access to quality learning, even in the face of financial constraints. Their belief in me, their constant encouragement, and their steadfast support have carried me through every step of this journey.

As I bring this section to a close, I must take a moment to thank myself for completing this thesis within such a short timeframe, despite going through one of the most challenging and unfortunate periods of my life. I must also acknowledge that this is far from an exhaustive account of the many ways kind and generous people have helped me along the way. Each of them has inspired me, and I can only hope to pay forward even a small fraction of the kindness I've received.

Contents

Acknowledgments **v**

List of Figures **x**

List of Tables **xii**

1 Optimal Design of Experiments **1**
- 1.1 Overview: Design of Experiments 1
- 1.2 Crossover Designs for GLM 5
- 1.3 Optimal Designs for Two-treatment Crossover Trials 15
- 1.4 Optimal Design for Multiple-treatment Crossover Trials . . . 23
- 1.5 Discussion 29

2 Optimal Crossover Designs for GLMs: An Application to Work Environment Experiment **31**
- 2.1 The Work Environment Experiment 31
- 2.2 Poisson Regression 33
- 2.3 Beta Regression 36
- 2.4 Gamma Regression 40
- 2.5 Discussion 44

3 A General Equivalence Theorem for Crossover Designs under GLMs **45**
- 3.1 Overview 45
- 3.2 Notation and Preliminaries 46
- 3.3 Equivalence Theorems for Crossover Designs 48
- 3.4 Real Example 57
- 3.5 Discussion 60

4 Overview: Computer Experiments **61**
- 4.1 Surrogates 61

4.2 Review: Space-Filling and Orthogonal Designs 63

5 Lasso Surrogate for Complex Computer Experiments 71
5.1 Overview 71
5.2 Weighted MaxPro Criterion 73
5.3 Orthogonal Weighted MaxPro Criterion 75
5.4 Sequential Variable Selection Algorithm 76
5.5 Simulation Study 78
5.6 Discussion 81

6 Deep Gaussian Process Surrogate for Complex Computer Experiments 82
6.1 Overview 82
6.2 Shallow Gaussian Process 84
6.3 Deep Gaussian Process 89
6.4 Variable Selection Algorithm 95
6.5 Discussion 99

7 Extension and Conclusion 100

Appendices 103
A.1 Appendix A: Optimal Crossover Designs 103
A.2 Appendix B: A General Equivalence Theorem for Crossover Designs 110

Bibliography 111

List of Figures

1.1 Optimal proportions for $p = 2$ case under θ_1. 17
1.2 Optimal proportions for $p = 2$ case under θ_2. 17
1.3 Optimal proportions for $p = 3$ case with two-treatment sequences under θ_1 18
1.4 Optimal proportions for $p = 3$ case with two-treatment sequences under θ_2 18
1.5 Optimal proportions for $p = 3$ case with four-treatment sequences under θ_1. 18
1.6 Optimal proportions for $p = 3$ case with four-treatment sequences: θ_2. 20
1.7 Optimal proportions for $p = 4$ case under θ_1 21
1.8 Optimal proportions for $p = 4$ case under θ_2 22
1.9 Performance of the locally optimal designs 27
1.10 Simulation Results: Ratios of the MSEs of the Uniform versus optimal deigns, for different values of ρ, for each of the two cases. 30

2.1 Uniform optimal proportions for Poisson response under θ_1 . 35
2.2 Non-uniform optimal proportions for Poisson response under θ_2 35
2.3 Uniform optimal proportions for beta response ($logit$ link) under θ_1 37
2.4 Non-uniform optimal proportions for beta response ($logit$ link) under θ_2 38
2.5 Uniform optimal proportions for beta response (log link) under θ_1 39
2.6 Non-uniform optimal proportions for beta response (log link) under θ_2 39
2.7 Uniform optimal proportions for gamma response (log link) under θ_1 41

2.8 Non-uniform optimal proportions for gamma response (log link) under θ_2 42
2.9 Uniform optimal proportions for gamma response (inv link) under θ_1 43
2.10 Non-uniform optimal proportions for gamma response (inv link) under θ_2 43

3.1 Objective function and its directional derivative for designs with two treatment sequences. 56
3.2 Objective function and its directional derivative for designs with two treatment sequences. 57

5.1 Illustration of the proposed design's nearly orthogonal (above) and space-filling (below) properties. 80

6.1 An example of a Bayesian framework for human pose tracking 83
6.2 Computer simulation of two black holes colliding. 84
6.3 Two Layer Deep Gaussian Process 92
6.4 Three Layer Deep Gaussian Process 93

List of Tables

1.1 Optimal proportions for $p = 2$ case. 16
1.2 Optimal proportions for $p = 3$ case for designs with two treatment sequences. 19
1.3 Optimal proportions for $p = 3$ case for designs with four treatment sequences. 19
1.4 Optimal proportions for $p = 4$ case. 21
1.5 Optimal Proportions when response is Poisson 23
1.6 Binary data from a four-period crossover trial. 24
1.7 Optimal proportions for different correlation matrices 25
1.8 Assumed values for model parameters 26
1.9 Simulation Results 29

2.1 Latin square design 32
2.2 Optimal proportions in case of Poisson response. 34
2.3 Optimal proportions in case of beta response ($logit$ link). . . . 37
2.4 Optimal proportions in case of beta response (log link). . . . 38
2.5 Optimal proportions in case of gamma response (log link). . . 41
2.6 Optimal proportions in case of gamma response ($inverse$ link). 42

5.1 Discovery rate of active and weakly active features under two threshold levels. 79

1 Optimal Design under Variance Misspecification 103
2 Optimal design considering 24 sequences under θ_1 104
3 Optimal design considering 24 sequences under θ_2. 105

Chapter 1

Optimal Design of Experiments

1.1 Overview: Design of Experiments

1.1.1 Industrial Experiments

Experimentation is a crucial method for gaining insights into processes and products, both in industrial settings and in research. However, conducting experiments in industry is often costly. As a result, alternative approaches, such as analyzing historical process data or consulting with process experts, are typically considered first, as they may provide the necessary information at a lower cost. Nevertheless, there are situations where experimentation remains the most effective—or even the only viable—way to acquire new knowledge or confirm suspected behaviors within a process.

An experiment involves conducting one or more tests in which deliberate changes are made to the input variables of a process or system to observe and understand how these changes affect the output response (Montgomery, 2009). The system under study may have a single response or multiple responses (denoted as Y). The main objective is to evaluate how the controllable input variables (X) influence the outcome. However, in many practical situations, there are other variables, known as disturbance or noise factors (Z), that also impact the response but cannot be easily controlled due to practical or cost-related limitations.

According to (Cox & Reid, 2000), experiments are generally performed in controlled environments where the experimenter determines the key characteristics of the materials, the type of manipulations applied, and the methods used for measurement. In contrast, observational studies do not offer full con-

trol over these elements, even if they share the same research goals. Therefore, experiments provide a stronger basis for establishing causal links between experimental factors and responses, something that is often more challenging to achieve through observational studies.

1.1.2 Design of Experiments

Given the high costs often associated with experimentation, it is important to obtain as much useful information as possible while using minimal resources. (C. J. Wu & Hamada, 2011) define Design of Experiments (DoE) as a set of principles and techniques that help researchers design more effective experiments, analyze data efficiently, and connect the results to the study's original objectives. DoE is particularly valuable for those seeking to gain insights into and improve a product or process in a structured and efficient manner.

The foundations of DoE were laid by Ronald A. Fisher and Frank Yates, who tackled agricultural and biological research challenges at the Rothamsted Experimental Station in the 1920s and 1930s (Box, 1980). Fisher's key contributions include advocating for the randomization of experimental treatments, introducing Analysis of Variance (ANOVA) to assess the significance of effects, and developing factorial designs. Factorial designs allow researchers to examine several experimental factors simultaneously, rather than studying them one at a time (Fisher, 1925).

Since the 1930s, the field of Design of Experiments (DoE) has undergone substantial evolution. (Steinberg & and, 1984) provide a detailed account of its progress up to the mid-1980s, with additional historical insights available in (Montgomery, 2009) and (C. J. Wu & Hamada, 2011). A major advancement was the development of fractional factorial designs (Finney, 1945). After World War II, DoE gained prominence as it was adapted to solve challenges in industrial settings, particularly within the chemical industry. Notable contributions came from George E. P. Box, who played a key role in introducing response surface methodology and sequential experimentation to optimize processes. During the 1970s, topics such as optimal design, computer-assisted design tools, and mixture designs attracted growing interest (Steinberg & and, 1984). In the 1980s, G. Taguchi's influential and often debated work shifted attention toward designing experiments aimed at reducing variability in products and processes. According to (Montgomery, 2009), the focus on quality improvement in Western manufacturing, combined with Taguchi's approaches, contributed to the broader adoption of DoE, especially in industries like automotive and

electronics. Today, DoE is widely used across numerous fields of science and engineering, far beyond its original applications in agriculture.

DoE involves numerous statistical techniques, making statistical knowledge essential for understanding how these methods operate. DoE, along with Statistical Process Control (SPC), was among the first tools adopted by the quality movement and is often seen as a key component of quality management practices. Today, the ongoing development of DoE methods is frequently featured in journals associated with the American Society for Quality and other publications focused on quality-related topics. Since improving quality typically involves minimizing variation in processes and products, there is a strong connection between statistical thinking and efforts to enhance quality (Snee, 1990).

As mentioned, statistics is a central part of DoE, but when statistical methods are applied, it is important not to forget about non-statistical knowledge. (Box et al., 2005) claim that "statistical techniques are useless unless combined with appropriate subject matter knowledge and experience." Thus, both statistical skills and process knowledge are needed to successfully design, conduct and analyze an experiment. In this thesis, we focus on one such special class of designs called crossover designs and work on developing statistical methodology and applications for crossover trials with non-normal responses.

1.1.3 Crossover Designs

Pharmaceutical companies frequently conduct clinical trials where the outcome is either the success or failure of a particular therapy. Crossover designs, also known as *repeated measurements designs* or *change-over designs*, have been used extensively in pharmaceutical research. There is a rich body of literature on optimal crossover designs when the response can be adequately modeled by normal distributions. However, for a binary outcome, where the response needs to be described using generalized linear models (GLMs), limited results are known. Consequently, these trials are usually designed using the guidelines of traditional crossover designs obtained using the theory of linear models. However, these designs can be quite inefficient for GLMs. Our goal is to bridge this gap in the literature and determine efficient designs specifically for crossover experiments with responses under univariate GLMs, including binary, binomial, Poisson, gamma, inverse Gaussian responses, etc.

Among different types of experiments that are available for treatment comparisons with multiple periods, the crossover designs are among the most important ones. In these experiments, every subject is exposed to a sequence of

treatments over different time periods, i.e., subjects crossover from one treatment to another. One of the most important aspects of crossover designs is that we can get the same number of observations as other designs but with less number of subjects. This is an important consideration since human participants are often scarce in clinical trials. The order in which treatments are applied to subjects is known as a *sequence* and the time at which these sequences are applied is known as a *period*. In most of cases, the main aim of such experiments is to compare t treatments over p periods. In each period, each subject receives a treatment, and the corresponding response is recorded. In different periods, a subject may receive different treatments, but the treatment may also be repeated on the same subject. Naturally, crossover designs also provide within-subject information about treatment differences.

Most of the research in the crossover design literature dealt with continuous response variables (see, for example, (Carriere & Huang, 2000; Kershner & Federer, 1981; Laska & Meisner, 1985; Matthews, 1987) and the references therein. The problem of determining optimal crossover designs for continuous responses has been studied extensively (see, for example, (Bose & Dey, 2009), for a review of results). For examples of practical cases where the responses are discrete in nature, such as binary responses, one may refer to (Kenward & Jones, 2014) and (Senn, 2003).

Among many fixed effects models proposed in the literature, the following linear model is used extensively to formulate crossover designs.

$$Y_{ij} = \lambda + \beta_i + \alpha_j + \tau_{d(i,j)} + \rho_{d(i-1,j)} + \epsilon_{ij}, \tag{1.1}$$

where Y_{ij} is the observation from the jth subject in the ith time period, with $i = 1, \ldots, p$ and $j = 1, \ldots, n$. Here $d(i, j)$ stands for the treatment assignment to the jth subject at time period i and $\lambda, \beta_i, \alpha_j, \tau_{d(i,j)}, \rho_{d(i-1,j)}$ are the corresponding overall mean, the ith period effect, the jth subject effect, the direct treatment effect and the carryover treatment effect respectively. Here ϵ_{ij}'s are the uncorrelated error terms which follow a normal distribution with zero mean and constant variance. Model (1.1) is sometimes referred to as the traditional model due to its extensive use in the literature.

As all the effects are fixed, for the linear model (1.1), the Fisher information matrix is independent of model parameters. Various optimality criteria such as A-, D-, E-optimality depend on this information matrix (see, for example, (Pukelsheim, 1993)). Numerous results corresponding to the optimality of crossover designs for linear models are available in the literature. (Cheng & Wu, 1980; Hedayat & Afsarinejad, 1975; Kunert, 1984) studied the optimality

of balanced, uniform designs. (Cheng & Wu, 1980) formulated theorems for optimality of strongly balanced design. (Kunert, 1983) produced results for the optimality of designs which are neither balanced nor strongly balanced. (Dey et al., 1983) were among the first ones to provide results for the optimality of designs when $p \leq t$. Considering arbitrary p and t with both $p \leq t$ and $p \geq t$, (Kushner, 1997) obtained conditions for universal optimality through approximate theory. Such results cannot be readily extended for binary responses since the Fisher information matrix for GLMs depends on the model parameters (McCullagh & Nelder, 1989; Stufken & Yang, 2012). In this thesis, we focus on local optimality to circumvent this problem (Khuri et al., 2006).

1.2 Crossover Designs for GLM

Although there is a rich body of literature on optimal crossover designs for linear models, the results on crossover designs under GLMs are meager. Before identifying optimal crossover designs, we first formally introduce the GLM and the associated optimal crossover designs.

This chapter is organized as follows: we describe a preliminary setup of a model for crossover designs for GLMs in Section 1.2.1 and then discuss Generalized Estimating Equations in Section 1.2.2. We propose different correlation structures in Section 1.2.3 and formulate locally optimal crossover designs along with an algorithm for obtaining such designs, in Section 1.2.4. In Sections 1.3 we provide examples of optimal design for two-treatment crossover trials. We calculate optimal designs for examples with a binary response in Section 1.3.1 and for examples with a Poisson response in Section 1.3.2. In Section 1.4.1, we provide examples of optimal designs for multi-treatment crossover trials, where we use the Latin square design. Sensitivity study and Relative D-efficiency are presented in Section 1.4.2. Simulation studies are presented in Section 1.4.3. The chapter concludes with comments in Section 1.5. Some technical details and additional results are presented in Appendix A.1.

1.2.1 Preliminary Setup

We consider a crossover trial with t treatments, n subjects, and p periods. The responses obtained from these n subjects are denoted as $Y_1, \ldots, Y_n$, where the response from the jth subject is $Y_j = (Y_{1j}, \ldots, Y_{pj})'$. As discussed above, we use a GLM to describe the marginal distribution of Y_{ij} as in (Liang & Zeger, 1986). Let μ_{ij} denote the mean of a binary response Y_{ij}. To fix ideas, first we consider logistic regression, which models the marginal mean μ_{ij} for the

crossover trial as

$$\text{logit}(\mu_{ij}) = \log\left(\frac{\mu_{ij}}{1-\mu_{ij}}\right) = \eta_{ij} = \lambda + \beta_i + \tau_{d(i,j)} + \rho_{d(i-1,j)}, \qquad (1.2)$$

where $i = 1, \ldots, p$; $j = 1, \ldots, n$; λ is the overall mean, β_i represents the effect of the ith period, τ_s is the direct effect due to treatment s and ρ_s is the carryover effect due to treatment s, where $s = 1, \ldots, t$.

Remark: Unlike model 1.1, model 1.2 does not contain a subject effect term α_j. Note that the response here is described by a GLM, where the Fisher information matrix depends on model parameters. In this thesis, we consider the local optimality approach of (Chernoff, 1953), in which the parameters are replaced by assumed values. In the linear model, the subject effect can be estimated from the data, but for our local optimality approach for the GLM, an educated guess for the subject effect is needed. It would be reasonable to guess the fixed treatment effects from prior knowledge, while from a design point of view, the subject effect, if included, has to be treated as random. Instead of incorporating a random effects term, in this thesis, the mean response is modeled through the logit link function in equation 1.2 with an extra assumption that the responses from a particular subject are mutually correlated, while the responses from different subjects are uncorrelated. In the case of GLMs, only the mean response is modeled through the link function, and hence we are free to choose a variance-covariance matrix as long as that is positive definite. So, in this thesis, we use this opportunity to choose the covariance matrix and capture the subject effect by putting different meaningful structures on this matrix and studying the robustness of the design. In this way, we can exclude a random subject effect from the model and calculate optimal designs more easily.

As the main interest is in estimating the treatment effects and variance of its estimator, carryover effects are treated as nuisance parameters. To ensure estimability of the model parameters, we set the baseline constraints as $\beta_1 = \tau_1 = \rho_1 = 0$. Consider $\beta = (\beta_2, \ldots, \beta_p)'$, $\tau = (\tau_2, \ldots, \tau_t)'$ and $\rho = (\rho_2, \ldots, \rho_t)'$, which define the parameter vector $\theta = (\lambda, \beta, \tau, \rho)'$. Then the linear predictor corresponding to the jth subject, $\eta_j = (\eta_{1j}, \ldots, \eta_{pj})'$ can be written as

$$\eta_j \;=\; X_j\theta.$$

The corresponding design matrix X_j can be written as $X_j = [1_p, P_j, T_j, F_j]$, where P_j is $p \times (p-1)$ such that $P_j = [0_{(p-1)1}, I_{p-1}]'$; where T_j is a $p \times (t-1)$ matrix with its $(i,s)^{\text{th}}$ entry equal to 1 if subject j receives the direct effect of

the treatment s in the i^{th} period and zero otherwise; where F_j is a $p \times (t-1)$ matrix with its $(i,s)^{\text{th}}$ entry equal to 1 if subject j receives the carryover effect of the treatment s in the i^{th} period and zero otherwise, where columns of T_j and F_j are indexed by $2, \ldots, t$.

If the number of subjects is fixed to n and the number of periods is p, then we determine the proportion of subjects assigned to a particular treatment sequence. As the number of periods is fixed to p, each treatment sequence will be of length p and a typical sequence can be written as $\omega = (t_1, \ldots, t_p)'$ where $t_i \in \{1, \ldots, t\}$. Now, let Ω be the set of all such sequences and n_ω denote the number of subjects assigned to sequence ω. Then, the total number of subjects n can be written as $n = \Sigma_{\omega \in \Omega} n_\omega$, $n_\omega \geq 0$. A crossover design ζ in approximate theory is specified by the set $\{p_\omega, \omega \in \Omega\}$, where $p_\omega = n_\omega / n$ is the proportion of subjects assigned to treatment sequence ω. Such a crossover design ζ can be denoted as follows:

$$\zeta = \left\{ \begin{array}{cccc} \omega_1 & \omega_2 & \ldots & \omega_k \\ p_{\omega_1} & p_{\omega_2} & \ldots & p_{\omega_k} \end{array} \right\}$$

where k is the number of treatment sequences involved, such that $\sum_{i=1}^{k} p_{\omega_i} = 1,$ for $i = 1, \ldots, k$. From the definitions of matrices T_j and F_j it can be noted that they depend only on the treatment sequence ω that subject j receives. So it can be inferred that $T_j = T_\omega$ and $F_j = F_\omega$. This implies, $X_j = X_\omega$ as $P_j = [0_{(p-1)1}, I_{p-1}]'$.

1.2.2 Generalized Estimating Equations

Generalized Estimating Equations (GEE) are quasi-likelihood equations which allow us to estimate quasi-likelihood estimators. In this thesis, instead of using maximum likelihood estimation (MLE) or ordinary least squares (OLS) to estimate the parameters, we use quasi-likelihood estimation. Earlier, we made one important assumption in crossover trials that observations from each subject are mutually correlated while the observations from different subjects are uncorrelated. This dependency between repeated observations from a subject is modeled using what is called the "working correlation" matrix C. If C is the true correlation matrix of Y_j, then from the definition of covariance we can write

$$Cov(Y_j) \;\; = \;\; D_j^{1/2} C D_j^{1/2},$$

where $D_j = diag\Big(\mu_{1j}(1-\mu_{1j}), \ldots, \mu_{pj}(1-\mu_{pj})\Big)$. Let us denote $Cov(Y_j)$ by W_j. In (Liang et al., 1988) equation (3.1), it has been shown that for the repeated measurement model, the GEE are defined to be

$$\sum_{j=1}^{n} \frac{\partial \mu_j'}{\partial \theta} W_j^{-1} \left(Y_j - \mu_j\right) = 0$$

where $\mu_j = (\mu_{1j}, \ldots, \mu_{pj})'$ and the asymptotic variance for the GEE estimator $\hat{\theta}$ (see Liang et al., 1988, equation (3.2)) is

$$\mathrm{Var}(\hat{\theta}) \quad = \quad \left[\sum_{j=1}^{n} \frac{\partial \mu_j'}{\partial \theta} W_j^{-1} \frac{\partial \mu_j}{\partial \theta}\right]^{-1} \tag{1.3}$$

where $W_j = Cov(Y_j)$. As mentioned by (Singh & Mukhopadhyay, 2016) in the thesis (Liang et al., 1988), equation (3.2) it has also been shown that if the true correlation structure varies from "working correlation" structure, then $\mathrm{Var}(\hat{\theta})$ is given by the sandwich formula

$$\mathrm{Var}(\hat{\theta}) \quad = \quad U^{-1} V U^{-1},$$

where the U and V in the above equation are as follows:

$$U = \sum_{\omega \epsilon \Omega} n p_\omega \frac{\partial \mu_\omega'}{\partial \theta} W_\omega^{-1} \frac{\partial \mu_\omega}{\partial \theta}. \tag{1.4}$$

$$V = \sum_{\omega \epsilon \Omega} n p_\omega \frac{\partial \mu_\omega'}{\partial \theta} W_\omega^{-1} Cov(Y_\omega) W_\omega^{-1} \frac{\partial \mu_\omega}{\partial \theta}. \tag{1.5}$$

So, it is expected that the effect of variance misspecification on the locally optimal designs will be minimal. Table A.1 presented in the Appendix A.1 confirms this.

From above equations 1.3, 1.4 and 1.5, it can be seen that if the true correlation of Y_j is equal to C, then $\mathrm{Var}(\hat{\theta}) = U^{-1}$. We have considered carryover effects to be nuisance parameters, since the main interest usually lies in estimating the direct treatment effect contrasts. So, instead of working with the full variance-covariance matrix of parameter estimator $\hat{\theta}$, we concentrate only on the variance of the estimator of treatment effect $\mathrm{Var}(\hat{\tau})$ where

$$\mathrm{Var}(\hat{\tau}) \quad = \quad H \mathrm{Var}(\hat{\theta}) H', \tag{1.6}$$

H is a $(t-1) \times m$ matrix given by $[0_{(t-1)1}, 0_{(t-1)(p-1)}, I_{t-1}, 0_{(t-1)(t-1)}]$ where $m = p + 2t - 2$ is the total number of parameters in θ and $0_{(t-1)(p-1)}$ is a $(t-1) \times (p-1)$ matrix of zeros.

We calculate optimal proportions such that the variances of the treatment effect estimators are minimized. In this thesis, we focus on D-optimality and use the determinant of $\mathrm{Var}(\hat{\tau})$ as our objective function. Note that other optimality criteria, such as A-,E-optimality, can be applied similarly. Then an optimal design ζ^* minimizes the determinant of $\mathrm{Var}(\hat{\tau})$ in equation (1.6) with respect to p_ω such that $\sum_{w \in \Omega} p_w = 1$. For illustration, we give an explicit expression of the information matrix and present the associated calculations for a crossover design in Appendix A.1.

1.2.3 Proposed Correlation Structures

As mentioned in the above section, to calculate the variance matrix of parameter estimates, a predefined working correlation structure for the responses is needed. Any correlation structure can be assumed for the responses, but if the design is not robust, then the optimal proportions will vary as the correlation structure varies. So, to check the robustness of the design and to make the design more practically acceptable, optimal proportions using different correlation structures are calculated. For the design in equation (2) with two treatments A and B, six different types of correlation structures are proposed, and optimal proportions are calculated. Out of these six correlation structures, the correlation matrices defined by the first three correlation structures are fixed and do not depend on treatment sequence, whereas the correlation matrices of the fourth, fifth and sixth types depend on treatment sequences and vary along with treatment sequences.

The first correlation structure is a compound symmetric correlation structure, i.e.,

$$Corr(1) = (1-\rho)I_p + \rho J_p,$$

where I_p is the identity matrix of order p, and J_p is a $p \times p$ matrix with all elements being unity.

The second correlation structure is the AR(1) correlation structure, i.e.,

$$Corr(2) = \left(\rho^{|i-i'|}\right),$$

so that the correlation between responses decreases as the time gap between responses increases.

The third correlation structure is as follows:

$$Corr(3) = \begin{pmatrix} 1 & \rho & 0 & \dots & 0 & 0 & 0 \\ \rho & 1 & \rho & \dots & 0 & 0 & 0 \\ \vdots & & & \vdots & & & \vdots \\ 0 & 0 & 0 & \dots & \rho & 1 & \rho \\ 0 & 0 & 0 & \dots & 0 & \rho & 1 \end{pmatrix}.$$

For each correlation structure, different correlation matrices using different ρ values are considered.

To understand the other three correlation structures, we denote the correlation coefficient between the response when a subject receives treatment A first and the response when the same subject receives treatment B after as ρ_{AB}, and ρ_{BA} when the subject receives B first and A afterward. Note that in general ρ_{AB} is not necessarily the same as ρ_{BA}. In a similar manner, we define ρ_{AA} and ρ_{BB}. To define the fourth type of correlation structure, we will use the same structure as $Corr(3)$ but with different values of the correlation coefficient for different treatment sequences. For the fourth type of correlation we use $\rho_{AB} = 0.2, \rho_{BA} = 0.5$ and $\rho_{AA} = 0.1, \rho_{BB} = 0.3$.

To define the fifth and sixth types of correlation structures, we use the AR(1) correlation structure with a correlation coefficient depending on the treatment sequence. For the fifth type, we use the same values for ρ_{AB} and ρ_{BA} and for the sixth type of correlation structure, we use different values for ρ_{AB} and ρ_{BA}. For both the fifth and sixth types of correlation structure, we keep $\rho_{AA} = \rho_{BB}$. These values may vary from one example to another and depend on what treatments A and B are. As the entries of the correlation matrix depend on which treatment the subject receives in a particular period, these correlation matrices are different for different treatment sequences. Here, we aim to see how optimal proportions vary as we vary values of ρ_{AB} and ρ_{BA}.

As an illustration, we consider $p = 2$ with treatment sequences AB and BA. Then the third type of correlation matrices for both treatment sequences AB and BA will have the same structure as $Corr(1)$. The fourth, fifth and sixth type correlation matrices will have the same structure as follows, with different ρ values,

$$Corr(4/5/6)_{AB} = \begin{pmatrix} 1 & \rho_{AB} \\ \rho_{AB} & 1 \end{pmatrix}, Corr(4/5/6)_{BA} = \begin{pmatrix} 1 & \rho_{BA} \\ \rho_{BA} & 1 \end{pmatrix}.$$

For the $p = 3$ case, consider an example with treatment sequences ABB and BAA. The fourth type of correlation matrix will have values as mentioned

above. The fifth type correlation matrices for both treatment sequences ABB and BAA will be the same if in treatment sequences, A and B are interchangeable and $\rho_{AB} = \rho_{BA}$ along with $\rho_{AA} = \rho_{BB}$. The sixth type correlation matrices for both treatment sequences ABB and BAA will be different as ρ_{AB} and ρ_{BA} are different. We get

$$Corr(4)_{ABB} = \begin{pmatrix} 1 & \rho_{AB} & 0 \\ \rho_{AB} & 1 & \rho_{BB} \\ 0 & \rho_{BB} & 1 \end{pmatrix}, Corr(4)_{BAA} = \begin{pmatrix} 1 & \rho_{BA} & 0 \\ \rho_{BA} & 1 & \rho_{AA} \\ 0 & \rho_{AA} & 1 \end{pmatrix},$$

and

$$Corr(5)_{ABB} = Corr(5)_{BAA} = \begin{pmatrix} 1 & \rho_{AB} & \rho_{AB}^2 \\ \rho_{AB} & 1 & \rho_{BB} \\ \rho_{AB}^2 & \rho_{BB} & 1 \end{pmatrix},$$

and

$$Corr(6)_{ABB} = \begin{pmatrix} 1 & \rho_{AB} & \rho_{AB}^2 \\ \rho_{AB} & 1 & \rho_{BB} \\ \rho_{AB}^2 & \rho_{BB} & 1 \end{pmatrix}, \quad Corr(6)_{BAA} = \begin{pmatrix} 1 & \rho_{BA} & \rho_{BA}^2 \\ \rho_{BA} & 1 & \rho_{AA} \\ \rho_{BA}^2 & \rho_{AA} & 1 \end{pmatrix}.$$

Same as the above two cases, for the $p = 4$ case, we consider an example with treatment sequences $AABB$ and $BBAA$. The fourth type of correlation matrix will be as given below. The fifth type of correlation matrices for both treatment sequences $AABB$ and $BBAA$ will be same because in treatment sequences A, B are interchangeable and $\rho_{AA} = \rho_{BB}$ and $\rho_{AB} = \rho_{BA}$. The sixth type of correlation matrices for both treatment sequences ABB and BAA will be different as ρ_{AB} and ρ_{BA} are different. We get

$$Corr(4)_{AABB} = \begin{pmatrix} 1 & \rho_{AA} & 0 & 0 \\ \rho_{AA} & 1 & \rho_{AB} & 0 \\ 0 & \rho_{AB} & 1 & \rho_{BB} \\ 0 & 0 & \rho_{BB} & 1 \end{pmatrix},$$

$$Corr(4)_{BBAA} = \begin{pmatrix} 1 & \rho_{BB} & 0 & 0 \\ \rho_{BB} & 1 & \rho_{BA} & 0 \\ 0 & \rho_{BA} & 1 & \rho_{AA} \\ 0 & 0 & \rho_{AA} & 1 \end{pmatrix},$$

and

$$Corr(5)_{AABB} = Corr(5)_{BBAA} = \begin{pmatrix} 1 & \rho_{BB} & \rho_{BA}^2 & \rho_{BA}^3 \\ \rho_{BB} & 1 & \rho_{BA} & \rho_{BB}^2 \\ \rho_{BA}^2 & \rho_{BA} & 1 & \rho_{BB} \\ \rho_{BA}^3 & \rho_{BA}^2 & \rho_{BB} & 1 \end{pmatrix},$$

and

$$Corr(6)_{AABB} = \begin{pmatrix} 1 & \rho_{AA} & \rho_{AB}^2 & \rho_{AB}^3 \\ \rho_{AA} & 1 & \rho_{AB} & \rho_{AB}^2 \\ \rho_{AB}^2 & \rho_{AB} & 1 & \rho_{BB} \\ \rho_{AB}^3 & \rho_{AB}^2 & \rho_{BB} & 1 \end{pmatrix},$$

$$Corr(6)_{BBAA} = \begin{pmatrix} 1 & \rho_{BB} & \rho_{BA}^2 & \rho_{BA}^3 \\ \rho_{BB} & 1 & \rho_{BA} & \rho_{BA}^2 \\ \rho_{BA}^2 & \rho_{BA} & 1 & \rho_{AA} \\ \rho_{BA}^3 & \rho_{BA}^2 & \rho_{AA} & 1 \end{pmatrix}.$$

For the $p = 4$ case, we discuss another interesting example with four treatments A, B, C and D. The set of treatment sequences for this example is $\Omega = \{ABCD, BDAC, CADB, DCBA\}$. This experiment will be discussed in detail later in Section 1.4. Note that the treatment sequences are given by a Latin square design shown below, and the treatments are interchangeable.

$$\begin{array}{cccc} A & B & C & D \\ B & D & A & C \\ C & A & D & B \\ D & C & B & A \end{array}$$

For this Latin square example, six different types of correlation matrices are considered. The first three correlation matrices will be the same as above with $\rho = 0.3$, $\rho = 0.2$ and $\rho = 0.1$ respectively. The fourth type of correlation structure will be defined in a similar manner to that discussed above. The fifth type correlation matrix is defined using AR(1) correlation structure with $\rho_{AB} = \rho_{AC} = \rho_{AD} = \rho_{BA} = \rho_{CA} = \rho_{DA} = 0.4$, $\rho_{BC} = \rho_{BD} = \rho_{CB} = \rho_{DB} = 0.3$ and $\rho_{CD} = \rho_{DC} = 0.2$. For the fourth and sixth types of correlation matrix, $\rho_{AB} = \rho_{AC} = \rho_{AD}$ is taken to be 0.4. In a similar manner $\rho_{BA} = \rho_{BC} = \rho_{BD}$ is taken to be 0.3 and $\rho_{CA} = \rho_{CB} = \rho_{CD}$ is taken to be 0.2 and $\rho_{DA} = \rho_{DB} = \rho_{DC}$ taken to be 0.1. As the entries of the correlation matrix depend on which treatment the subject receives in a particular period, these correlation matrices are different for different treatment sequences and are listed as follows:

$$Corr(4)_{ABCD} = \begin{pmatrix} 1 & \rho_{AB} & 0 & 0 \\ \rho_{AB} & 1 & \rho_{BC} & 0 \\ 0 & \rho_{BC} & 1 & \rho_{CD} \\ 0 & 0 & \rho_{CD} & 1 \end{pmatrix},$$

$$Corr(4)_{BDAC} = \begin{pmatrix} 1 & \rho_{BD} & 0 & 0 \\ \rho_{BD} & 1 & \rho_{DA} & 0 \\ 0 & \rho_{DA} & 1 & \rho_{AC} \\ 0 & 0 & \rho_{AC} & 1 \end{pmatrix},$$

$$Corr(4)_{CADB} = \begin{pmatrix} 1 & \rho_{CA} & 0 & 0 \\ \rho_{CA} & 1 & \rho_{AD} & 0 \\ 0 & \rho_{AD} & 1 & \rho_{DB} \\ 0 & 0 & \rho_{DB} & 1 \end{pmatrix},$$

$$Corr(4)_{DCBA} = \begin{pmatrix} 1 & \rho_{DC} & 0 & 0 \\ \rho_{DC} & 1 & \rho_{CB} & 0 \\ 0 & \rho_{CB} & 1 & \rho_{BA} \\ 0 & 0 & \rho_{BA} & 1 \end{pmatrix},$$

and

$$Corr(5/6)_{ABCD} = \begin{pmatrix} 1 & \rho_{AB} & \rho_{AC}^2 & \rho_{AD}^3 \\ \rho_{AB} & 1 & \rho_{BC} & \rho_{BD}^2 \\ \rho_{AC}^2 & \rho_{BC} & 1 & \rho_{CD} \\ \rho_{AD}^3 & \rho_{BD}^2 & \rho_{CD} & 1 \end{pmatrix},$$

$$Corr(5/6)_{BDAC} = \begin{pmatrix} 1 & \rho_{BD} & \rho_{BA}^2 & \rho_{BC}^3 \\ \rho_{BD} & 1 & \rho_{DA} & \rho_{DC}^2 \\ \rho_{BA}^2 & \rho_{DA} & 1 & \rho_{AC} \\ \rho_{BC}^3 & \rho_{DC}^2 & \rho_{AC} & 1 \end{pmatrix},$$

$$Corr(5/6)_{CADB} = \begin{pmatrix} 1 & \rho_{CA} & \rho_{CD}^2 & \rho_{CB}^3 \\ \rho_{CA} & 1 & \rho_{AD} & \rho_{AB}^2 \\ \rho_{CD}^2 & \rho_{AD} & 1 & \rho_{DB} \\ \rho_{CB}^3 & \rho_{AB}^2 & \rho_{DB} & 1 \end{pmatrix},$$

$$Corr(5/6)_{DCBA} = \begin{pmatrix} 1 & \rho_{DC} & \rho_{DB}^2 & \rho_{DA}^3 \\ \rho_{DC} & 1 & \rho_{CB} & \rho_{CA}^2 \\ \rho_{DB}^2 & \rho_{CB} & 1 & \rho_{BA} \\ \rho_{DA}^3 & \rho_{CA}^2 & \rho_{BA} & 1 \end{pmatrix}.$$

In the above section, we only specified the forms of correlation structures. Note that for this particular example, the form of $Corr(5)$ is the same as that of $Corr(6)$ since the treatment sequences are obtained using a Latin square design. In Section 1.4, we will consider the above six types of correlation structures and calculate the corresponding optimal proportions. We will also perform a simulation analysis using this example. For simulation analysis, the AR(1) correlation structure will be considered with different ρ values. We have performed robustness analysis in Appendix A.1 and provided explicit expressions for obtaining the objective function in Supplementary Section A.1.

1.2.4 Algorithm for Locally Optimal Crossover Trials

In this section, we propose an algorithm to find locally optimal designs for crossover trials. Assumed values of the model parameters are obtained from some prior knowledge or pilot studies. To identify the locally optimal crossover design, the major challenge lies is in minimizing the objective function. The complexity of the objective function increases with the increase of t, p and k. We use the `solnp` function in **R** for numerical optimization.

Algorithm : Pseudo-code for finding locally optimal crossover designs.

Given assumed values of the parameters, construct the design matrix, correlation matrix, and the parameter vector.

for
Each subject in each period.

Calculate the mean of the response.

end

for
Each treatment sequence.

Calculate the covariance matrix using the correlation matrix.

Diagonal entries of covariance matrix are variances of observations.

Variance depends on the distribution of the response.

Calculate the inverse of covariance matrix.

end

for
Each treatment sequence.

Calculate the corresponding derivative matrix.

Using calculated matrices and variables corresponding to each treatment sequence, compute the variance matrix of parameter estimates.

Calculate variance matrix of treatment effects. Its determinant is the required objective function.

end

function

Define the objective function along with the constraints, i.e., sum of proportions is equal to one.

end

solnp Using this constraint optimization function, calculate optimal proportions.

1.3 Optimal Designs for Two-treatment Crossover Trials

The crossover designs for which we will calculate the optimal proportions are similar to those discussed by (Laska & Meisner, 1985) and (Carriere & Huang, 2000). Optimal proportions are listed below for $p = 2, 3, 4$ for the binary response and for $p = 2$ for the Poisson response under two sets of parameter estimates. In this section, we consider only two treatments, A and B. Considering our baseline constraint to be $\tau_A = \rho_A = 0$ and $\beta_1 = 0$ we only have $p + 2$ parameters in vector θ. So, when there are only two treatments involved in the crossover trial, the parameter vector θ is $[\lambda, \beta_2, \ldots, \beta_p, \tau_2, \rho_2]$.

Optimal proportions for different crossover designs are calculated with each of the six different correlation structures mentioned above. For each correlation matrix that we consider, an optimal design ζ^* is the one minimizing the determinant of $\mathrm{Var}(\hat{\tau})$ in equation (1.6) with respect to p_ω such that $\sum_{w \in \Omega} p_w = 1$.

We use different colors to represent different correlation structures. The color scheme that we use is as follows:

Correlation Structure	Color
Corr(1) $(1-\rho)I_p + \rho J_p$ with $\rho = 0.1$	
Corr(2) $\rho^{\lvert i-i' \rvert}, i \neq i'$ with $\rho = 0.1$	
Corr(3) with $\rho = 0.1$	
Corr(4) with $\rho_{AB} = 0.2, \rho_{BA} = 0.5$	
Corr(5) with $\rho_{AB} = \rho_{BA} = 0.4$	
Corr(6) with $\rho_{AB} = 0.4, \rho_{BA} = 0.3$	

1.3.1 Optimal Designs when Response is Binary

In case of binary response, we calculate locally optimal designs under model 1.2 for different crossover designs.

We first consider the local optimality approach for the $p = 2$ case. For illustration purpose, we assume that the parameter values are $\theta_1 = [\lambda, \beta_2, \tau_B, \rho_B] = [0.5, -1.0, 4.0, -2.0]$ which gives us non-uniform optimal allocations and $\theta_2 = [\lambda, \beta_2, \tau_B, \rho_B] = [0.5, 0.06, -0.35, 0.73]$ which gives us approximately uniform allocations. Note that we need to know the parameter values before calculating the optimal proportions. If the initial guess for the model parameters changes, the obtained optimal proportions will change as well. For different correlation structures, the optimal designs (proportions) are stated in Table 1.1. The same information is presented in Figure 1.2 and Figure 1.1 as well.

Table 1.1: Optimal proportions for $p = 2$ case.

Design Points	Corr	Optimal proportions under θ_1	Optimal proportions under θ_2
$\{AB, BA\}$	Corr(1)	$\{0.1770, 0.8230\}$	$\{0.5070, 0.4930\}$
	Corr(2)	$\{0.1770, 0.8230\}$	$\{0.5070, 0.4930\}$
	Corr(3)	$\{0.1770, 0.8230\}$	$\{0.5070, 0.4930\}$
	Corr(4)	$\{0.1770, 0.8230\}$	$\{0.5070, 0.4930\}$
	Corr(5)	$\{0.1770, 0.8230\}$	$\{0.5070, 0.4930\}$
	Corr(6)	$\{0.1770, 0.8230\}$	$\{0.5070, 0.4930\}$
$\{AB, BA, AA, BB\}$	Corr(1)	$\{0.0908, 0.5207, 0.0315, 0.3570\}$	$\{0.2633, 0.2425, 0.2722, 0.2220\}$
	Corr(2)	$\{0.0908, 0.5207, 0.0315, 0.3570\}$	$\{0.2633, 0.2425, 0.2722, 0.2220\}$
	Corr(3)	$\{0.0908, 0.5207, 0.0315, 0.3570\}$	$\{0.2633, 0.2425, 0.2722, 0.2220\}$
	Corr(4)	$\{0.0957, 0.4960, 0.0338, 0.3745\}$	$\{0.2534, 0.2393, 0.2661, 0.2412\}$
	Corr(5)	$\{0.1002, 0.4941, 0.0379, 0.3678\}$	$\{0.2496, 0.2359, 0.2801, 0.2344\}$
	Corr(6)	$\{0.0972, 0.5050, 0.0367, 0.3611\}$	$\{0.2502, 0.2400, 0.2808, 0.2290\}$

It can be seen from the graphs in Figure 1.1 and Figure 1.2 that in the case of $p = 2$, the optimal proportions do not vary when the correlation structure changes both under θ_2 and θ_1. Uniform designs (same proportions for each sequence) are often used in practice. Those uniform designs are sub-optimal under θ_1.

For $p = 3$ case, as before suppose our guess for the parameter values are $\theta_1 = [\lambda, \beta_2, \beta_3, \tau_B, \rho_B] = [0.5, -1.0, 2.0, 4.0, -2.0]$ which gives us non-uniform optimal allocations and $\theta_2 = [\lambda, \beta_2, \beta_3, \tau_B, \rho_B] = [0.5, 0.06, -0.53, -0.35, 0.73]$

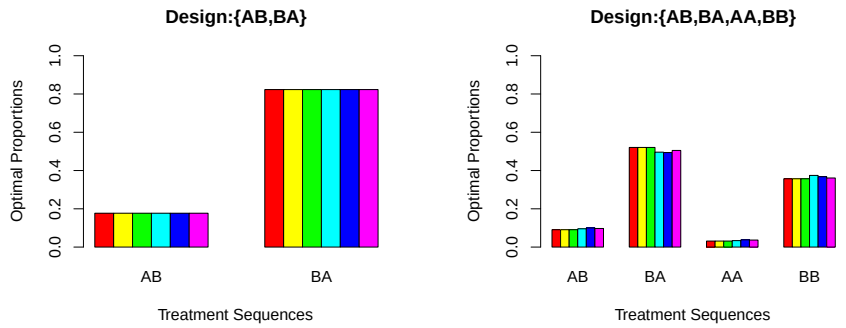


Figure 1.1: Optimal proportions for $p = 2$ case under θ_1.

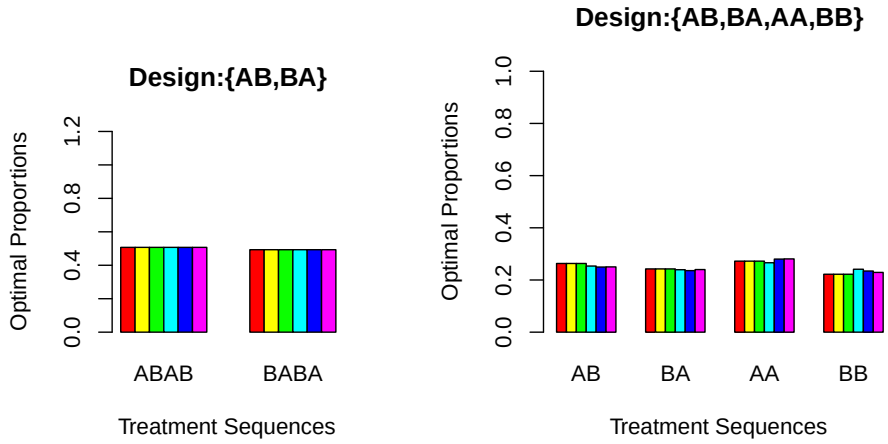


Figure 1.2: Optimal proportions for $p = 2$ case under θ_2.

which gives us approximately uniform optimal allocations. The designs are presented in Table 1.2, Figure 1.3 and Figure 1.4 for the first example, and in Table 1.3, Figure 1.5 and Figure 1.6 for the second example. It can be seen that in the case of $p = 3$, the optimal proportions do not vary much when the correlation structure changes under both θ_1 and θ_2. Similar to the $p = 2$ case, it is clear from the above table that uniform designs are sub-optimal for the $p = 3$ case with two- and four-treatment sequences under θ_1.

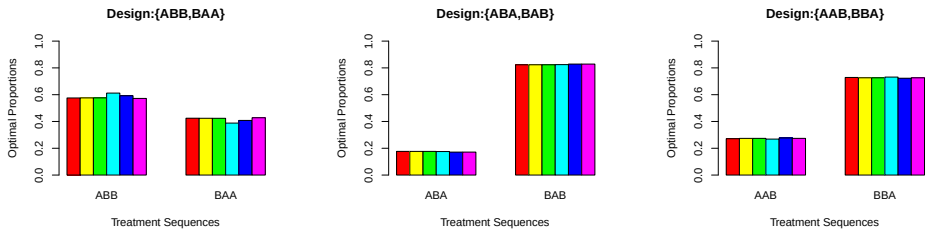


Figure 1.3: Optimal proportions for $p = 3$ case with two-treatment sequences under θ_1

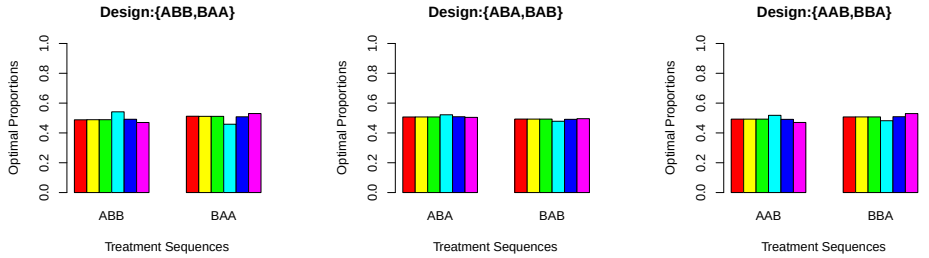


Figure 1.4: Optimal proportions for $p = 3$ case with two-treatment sequences under θ_2

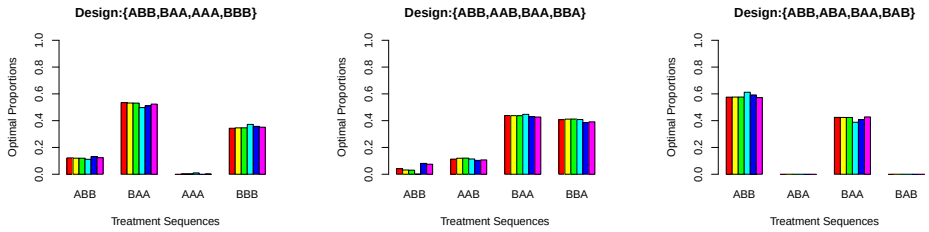


Figure 1.5: Optimal proportions for $p = 3$ case with four-treatment sequences under θ_1.

Table 1.2: Optimal proportions for $p = 3$ case for designs with two treatment sequences.

Design Points	Corr	Optimal proportions under θ_1	Optimal proportions under θ_2
$\{ABB, BAA\}$	Corr(1)	$\{0.5756, 0.4244\}$	$\{0.4880, 0.5120\}$
	Corr(2)	$\{0.5761, 0.4239\}$	$\{0.4887, 0.5113\}$
	Corr(3)	$\{0.5762, 0.4238\}$	$\{0.4888, 0.5112\}$
	Corr(4)	$\{0.6120, 0.3880\}$	$\{0.5416, 0.4584\}$
	Corr(5)	$\{0.5921, 0.4079\}$	$\{0.4917, 0.5083\}$
	Corr(6)	$\{0.5721, 0.4279\}$	$\{0.4700, 0.5300\}$
$\{ABA, BAB\}$	Corr(1)	$\{0.1768, 0.8232\}$	$\{0.5070, 0.4930\}$
	Corr(2)	$\{0.1766, 0.8234\}$	$\{0.5072, 0.4928\}$
	Corr(3)	$\{0.1766, 0.8234\}$	$\{0.5072, 0.4928\}$
	Corr(4)	$\{0.1756, 0.8244\}$	$\{0.5217, 0.4783\}$
	Corr(5)	$\{0.1714, 0.8286\}$	$\{0.5088, 0.4912\}$
	Corr(6)	$\{0.1715, 0.8285\}$	$\{0.5043, 0.4957\}$
$\{AAB, BBA\}$	Corr(1)	$\{0.2713, 0.7287\}$	$\{0.4927, 0.5073\}$
	Corr(2)	$\{0.2738, 0.7262\}$	$\{0.4926, 0.5074\}$
	Corr(3)	$\{0.2740, 0.7260\}$	$\{0.4926, 0.5074\}$
	Corr(4)	$\{0.2685, 0.7315\}$	$\{0.5181, 0.4819\}$
	Corr(5)	$\{0.2771, 0.7229\}$	$\{0.4911, 0.5089\}$
	Corr(6)	$\{0.2740, 0.7260\}$	$\{0.4702, 0.5298\}$

Table 1.3: Optimal proportions for $p = 3$ case for designs with four treatment sequences.

Design Points	Corr	Optimal proportions under θ_1	Optimal proportions under θ_2
$\{ABB, BAA, AAA, BBB\}$	Corr(1)	$\{0.1222, 0.5344, 0.0000, 0.3434\}$	$\{0.4880, 0.5120, 0.0000, 0.0000\}$
	Corr(2)	$\{0.1199, 0.5316, 0.0022, 0.3463\}$	$\{0.4887, 0.5113, 0.0000, 0.0000\}$
	Corr(3)	$\{0.1197, 0.5312, 0.0025, 0.3466\}$	$\{0.4888, 0.5112, 0.0000, 0.0000\}$
	Corr(4)	$\{0.1115, 0.4975, 0.0100, 0.3720\}$	$\{0.5398, 0.4556, 0.0046, 0.0000\}$
	Corr(5)	$\{0.1313, 0.5113, 0.0000, 0.3574\}$	$\{0.4917, 0.5083, 0.0000, 0.0000\}$
	Corr(6)	$\{0.1233, 0.5236, 0.0018, 0.3513\}$	$\{0.4700, 0.5300, 0.0000, 0.0000\}$
$\{ABB, AAB, BAA, BBA\}$	Corr(1)	$\{0.0413, 0.1130, 0.4384, 0.4073\}$	$\{0.3544, 0.1646, 0.3908, 0.0902\}$
	Corr(2)	$\{0.0316, 0.1196, 0.4373, 0.4115\}$	$\{0.4266, 0.0957, 0.4777, 0.0000\}$
	Corr(3)	$\{0.0304, 0.1204, 0.4371, 0.4121\}$	$\{0.4271, 0.0953, 0.4776, 0.0000\}$
	Corr(4)	$\{0.0005, 0.1440, 0.4471, 0.4084\}$	$\{0.1512, 0.3503, 0.1854, 0.3131\}$
	Corr(5)	$\{0.0811, 0.1033, 0.4297, 0.3858\}$	$\{0.4420, 0.0747, 0.4833, 0.0000\}$
	Corr(6)	$\{0.0749, 0.1070, 0.4270, 0.3911\}$	$\{0.4094, 0.0955, 0.4951, 0.0000\}$
$\{ABB, ABA, BAA, BAB\}$	Corr(1)	$\{0.5755, 0.0000, 0.4244, 0.0000\}$	$\{0.4606, 0.0194, 0.4710, 0.0490\}$
	Corr(2)	$\{0.5761, 0.0000, 0.4239, 0.0000\}$	$\{0.4430, 0.0391, 0.4526, 0.0653\}$
	Corr(3)	$\{0.5762, 0.0000, 0.4238, 0.0000\}$	$\{0.4408, 0.0415, 0.4504, 0.0673\}$
	Corr(4)	$\{0.6120, 0.0000, 0.3880, 0.0000\}$	$\{0.4634, 0.1036, 0.4152, 0.0178\}$
	Corr(5)	$\{0.5921, 0.0000, 0.4079, 0.0000\}$	$\{0.4582, 0.0280, 0.4642, 0.0496\}$
	Corr(6)	$\{0.5721, 0.0000, 0.4279, 0.0000\}$	$\{0.4420, 0.0142, 0.4787, 0.0651\}$

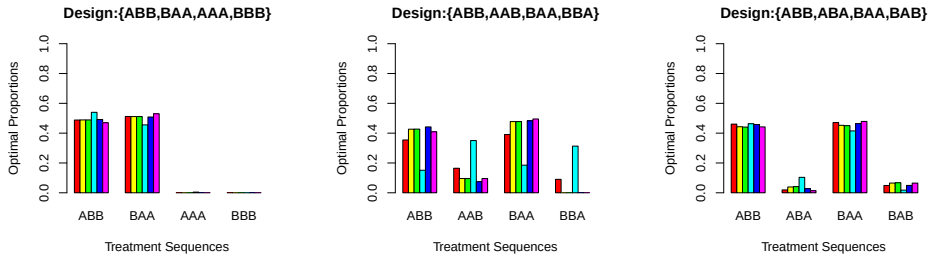


Figure 1.6: Optimal proportions for $p = 3$ case with four-treatment sequences: θ_2.

An interesting thing to observe from Figure 1.5 is that, unlike the previous examples, here under θ_1, the optimal proportions vary a little for different correlation structures. Also, as before, not only is the uniform design sub-optimal here, but the first and third designs have optimal allocations very low for some sequences. In design with $\{ABB, BAA, AAA, BBB\}$ the optimal proportion corresponding to treatment sequence AAA is almost zero. In design with $\{ABB, ABA, BAA, BAB\}$, the optimal proportions corresponding to the treatment sequence ABA and BAB are zero. Also, it can be observed from Figure 1.6 that under θ_2 for different correlation structures, some of the optimal proportions are zero for all three designs. Hence, under θ_2 these designs fail to have uniform allocations.

For $p = 4$ case, in a similar way, we calculate locally optimal designs with nominal parameter values as $\theta_1 = [\lambda, \beta_2, \beta_3, \beta_4, \tau_B, \rho_B] = [0.5, -1.0, 2.0, -1.5, 4.0, -2.0]$ which gives us non-uniform allocations and $\theta_2 = [\lambda, \beta_2, \beta_3, \beta_4, \tau_B, \rho_B] = [0.5, 0.06, -0.53, -0.6, -0.35, 0.73]$ which gives us approximately uniform allocations. From Table 1.4 and Figure 1.8 it is clear that, similar to $p = 2$ and $p = 3$ cases, the uniform designs are sub-optimal for $p = 4$ case under θ_1.

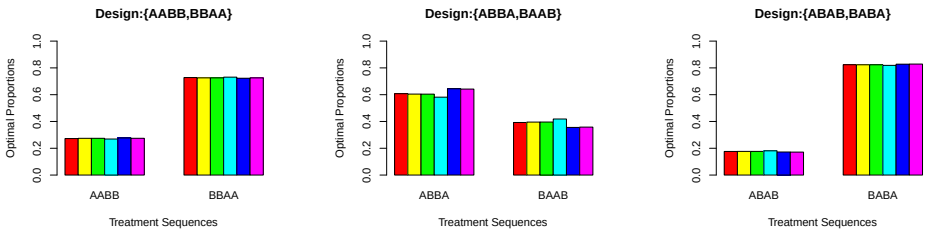


Figure 1.7: Optimal proportions for $p = 4$ case under θ_1

Table 1.4: Optimal proportions for $p = 4$ case.

Design Points	Corr	Optimal proportions under θ_1	Optimal proportions under θ_2
$\{AABB, BBAA\}$	Corr(1)	$\{0.2723, 0.7277\}$	$\{0.4953, 0.5047\}$
	Corr(2)	$\{0.2743, 0.7257\}$	$\{0.4949, 0.5051\}$
	Corr(3)	$\{0.2744, 0.7256\}$	$\{0.4949, 0.5051\}$
	Corr(4)	$\{0.2690, 0.7310\}$	$\{0.5244, 0.4756\}$
	Corr(5)	$\{0.2772, 0.7228\}$	$\{0.4937, 0.5063\}$
	Corr(6)	$\{0.2745, 0.7255\}$	$\{0.4700, 0.5300\}$
$\{ABBA, BAAB\}$	Corr(1)	$\{0.6075, 0.3925\}$	$\{0.4992, 0.5008\}$
	Corr(2)	$\{0.6045, 0.3955\}$	$\{0.4998, 0.5002\}$
	Corr(3)	$\{0.6042, 0.3958\}$	$\{0.4998, 0.5002\}$
	Corr(4)	$\{0.5815, 0.4185\}$	$\{0.4927, 0.5073\}$
	Corr(5)	$\{0.6444, 0.3556\}$	$\{0.5021, 0.4979\}$
	Corr(6)	$\{0.6419, 0.3581\}$	$\{0.5007, 0.4993\}$
$\{ABAB, BABA\}$	Corr(1)	$\{0.1763, 0.8237\}$	$\{0.5071, 0.4929\}$
	Corr(2)	$\{0.1767, 0.8233\}$	$\{0.5071, 0.4929\}$
	Corr(3)	$\{0.1767, 0.8233\}$	$\{0.5071, 0.4929\}$
	Corr(4)	$\{0.1722, 0.8278\}$	$\{0.5086, 0.4914\}$
	Corr(6)	$\{0.1714, 0.8286\}$	$\{0.5031, 0.4969\}$

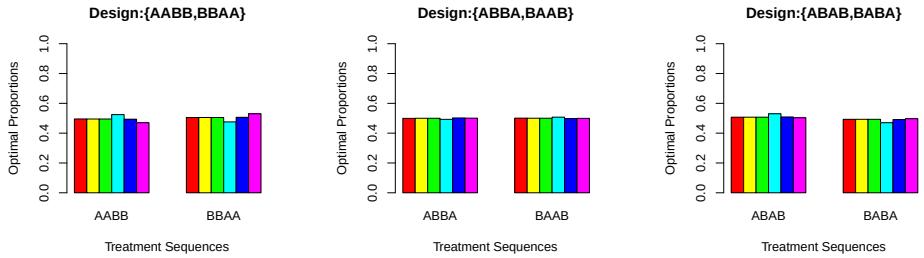


Figure 1.8: Optimal proportions for $p = 4$ case under θ_2

In most cases, we may not have a clear idea about the true correlation structure for responses, and hence we choose a working correlation structure. The results in this section show that no matter what correlation structure we choose or what parameter estimates we choose, the proposed design gives almost similar optimal proportions in each case, which suggests that optimal designs are robust.

1.3.2 Optimal Designs when Response is Poisson

In the case of Poisson response, we calculate the locally optimal design for the following example under the model,

$$\log(\mu_{ij}) = \eta_{ij} = \lambda + \beta_i + \tau_{d(i,j)} + \rho_{d(i-1,j)}, \tag{1.7}$$

where notations have the same meaning as in equation (1.2).

We consider an example described in (Layard & Arvesen, 1978). In a crossover clinical trial to test a standard anti-nausea treatment (drug A) against a proposed treatment (drug B), twenty subjects were tested, ten for each order of administration. The response variable is the number of episodes of nausea suffered by a patient during the first two hours after cancer chemotherapy, and for a given patient is approximately Poisson distributed.

We calculate optimal designs using two values of parameter estimates. $\theta_1 = [0.2, 0.34, -1.60, -1.65]$ represents those parameter estimates that give us non-uniform designs, and $\theta_2 = [-0.223, -0.875, 0.405, -0.105]$ corresponds to parameter estimates guessed from the data presented in the table below.
It can be noted from the above table that when responses are Poisson in nature, the optimal proportions do not vary much when the correlation structure changes under both θ_1 and θ_2. This suggests to us that even when responses

Table 1.5: Optimal Proportions when response is Poisson

Design Points	Correlation Structure	Optimal Design: θ_1
$\{AB, BA\}$	Corr(1) $(1-\rho)I_p + \rho J_p$ with $\rho = 0.1$	$\{0.3632, 0.6368\}$
	Corr(2) $\rho^{\lvert i-i' \rvert}, i \neq i'$ with $\rho = 0.1$	$\{0.3632, 0.6368\}$
	Corr(3) with $\rho = 0.1$	$\{0.3632, 0.6368\}$
	Corr(4) with $\rho_{AB} = 0.2, \rho_{BA} = 0.5$	$\{0.3632, 0.6368\}$
	Corr(5) with $\rho_{AB} = \rho_{BA} = 0.4$	$\{0.3632, 0.6368\}$
	Corr(6) with $\rho_{AB} = 0.4, \rho_{BA} = 0.3$	$\{0.3632, 0.6368\}$
Design Points	Correlation Structure	Optimal Design: θ_2
$\{AB, BA\}$	Corr(1) $(1-\rho)I_p + \rho J_p$ with $\rho = 0.1$	$\{0.5505, 0.4495\}$
	Corr(2) $\rho^{\lvert i-i' \rvert}, i \neq i'$ with $\rho = 0.1$	$\{0.5505, 0.4495\}$
	Corr(3) with $\rho = 0.1$	$\{0.5505, 0.4495\}$
	Corr(4) with $\rho_{AB} = 0.2, \rho_{BA} = 0.5$	$\{0.5505, 0.4495\}$
	Corr(5) with $\rho_{AB} = \rho_{BA} = 0.4$	$\{0.5505, 0.4495\}$
	Corr(6) with $\rho_{AB} = 0.4, \rho_{BA} = 0.3$	$\{0.5505, 0.4495\}$

are Poisson in nature, the proposed design gives almost similar optimal proportions for different choices of correlation matrices. Hence, the obtained optimal designs are robust.

1.4 Optimal Design for Multiple-treatment Crossover Trials

So far, we have considered crossover designs with two treatments only. In this section, we extend our study for multiple treatments. This is motivated by a four-period four treatment trial which was first given in (Kenward & Jones,

1992) and later discussed as Example 6.1 in their book (Kenward & Jones, 2014), *Design and Analysis for Crossover Trials*.

1.4.1 Latin Square Design and Optimal Proportions

In this example, binary responses for a four-period crossover trial were obtained. There were four treatments, and treatment sequences were allocated at random to eighty different subjects at four different periods. At the end of each eriod, efficacy measurement of each subject was recorded as success or failure, which resulted in joint outcome at theend of the trial as shown in Table 1.6. The dataset contains four different treatment sequences which were decided before the trial $\Omega = \{ABCD, BDAC, CADB, DCBA\}$, along with the joint outcome of four different periods from the same subject according to a particular treatment sequence. The numbers below each sequence denote how many subjects received that particular treatment sequence, and the particular response was recorded.

Table 1.6: Binary data from a four-period crossover trial.

Joint Outcome	Frequency of Outcome			
(1=Success, 0=Failure)	$ABCD$	$BDAC$	$CADB$	$DCBA$
$(0,0,0,0)$	1	0	1	1
$(0,0,0,1)$	0	1	1	0
$(0,0,1,0)$	1	1	0	1
$(0,0,1,1)$	1	0	0	0
$(0,1,0,0)$	1	1	1	0
$(0,1,0,1)$	1	1	1	2
$(0,1,1,0)$	1	1	1	2
$(0,1,1,1)$	0	1	1	0
$(1,0,0,0)$	1	0	1	0
$(1,0,0,1)$	1	1	0	0
$(1,0,1,0)$	1	0	1	0
$(1,0,1,1)$	2	0	0	1
$(1,1,0,0)$	1	1	1	0
$(1,1,0,1)$	0	2	2	4
$(1,1,1,0)$	2	3	3	0
$(1,1,1,1)$	4	9	5	10

We use the correlation matrices defined in Section 1.2.3 and calculate the optimal proportions. As mentioned earlier, for estimating parameters, we have

considered the baseline constraints as $\beta_1 = \tau_A = \rho_A = 0$, so that the design matrix has full column rank, and all other parameters are estimable.

Using these baseline constraints and the `glm` function in **R**, we fit the model, which gives us parameter estimates for the given data. Then we use these parameter estimates to guess values of unknown parameters. Our nominal guess for the parameter values are $\theta_2 = [0.5, 0.06, -0.53, -0.6, -0.35, 0.025, -0.23, 0.73, 0.23, 0.30]$. Now, we follow the same procedure mentioned in the pseudo code above and calculate the optimal designs for different correlation structures. We also calculate optimal proportions by considering parameter estimates that gives non-uniform designs i.e. $\theta_1 = [-2, 0.25, 0, 0.75, 1, 5, -1.5, -3.5, 2.75, 0.75]$. As seen from Table 1.7, for the Latin square design the optimal proportions that we obtain using θ_1 are non-uniform, and those obtained using θ_2 are nearly uniform.

Table 1.7: Optimal proportions for different correlation matrices

Correlation	θ_1				θ_2			
Structure	$ABCD$	$BDAC$	$CADB$	$DCBA$	$ABCD$	$BDAC$	$CADB$	$DCBA$
$Corr(1)$	0.1725	0.2483	0.2223	0.3569	0.2463	0.2493	0.2504	0.2540
$Corr(2)$	0.1747	0.2490	0.2184	0.3579	0.2461	0.2493	0.2501	0.2546
$Corr(3)$	0.1714	0.2480	0.2236	0.3570	0.2461	0.2492	0.2507	0.2540
$Corr(4)$	0.1788	0.2556	0.2163	0.3493	0.2478	0.2634	0.2334	0.2554
$Corr(5)$	0.1784	0.2465	0.2101	0.3650	0.2480	0.2517	0.2442	0.2561
$Corr(6)$	0.1752	0.2531	0.2170	0.3547	0.2470	0.2656	0.2320	0.2554

We also calculate the optimal design considering all 24 sequences. We consider $Corr(2)$ and calculate optimal proportions for different values of ρ. Please refer to the Appendix A.1 for details. From the tables in the Appendix A.1 it can be noted that corresponding to θ_1 we have non-uniform allocations for the Latin Square design, and almost uniform allocation corresponding to θ_2. In case of non-uniform allocations, although nothing is uniform, the optimal design corresponding to θ_1 has more zeros. Also note that the allocations do not vary a lot as ρ changes, particularly for the sequences where we have zero allocations.

1.4.2 Sensitivity Study and Relative D-efficiency

In this section, we study the performance of the proposed locally optimal designs via a sensitivity study in terms of relative D-efficiencies. Let θ_t be true parameter values and θ_c be assumed parameter values. Then we have corresponding objective functions for these two choices of parameter values i.e $det(var(\hat{\tau}_t))$ and $det(var(\hat{\tau}_c))$ respectively. Hence the relative loss of efficiency of choosing θ_c instead of θ_t can be formulated as

$$S(\tau_t, \tau_c) = \frac{det(var(\hat{\tau}_t))^{(-\frac{1}{k})} - det(var(\hat{\tau}_c))^{(-\frac{1}{k})}}{det(var(\hat{\tau}_t))^{(-\frac{1}{k})}},$$

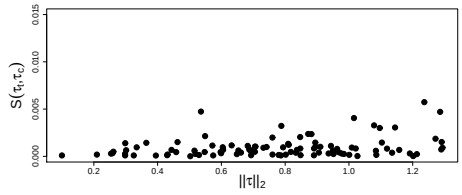
where k is the dimension of τ. Then the relative D-efficiency of the original design ξ compared to the optimal design ξ^* can be computed using the formula:

$$E_\xi = \left[\frac{det(var(\hat{\tau}_c))_{\xi^*}}{det(var(\hat{\tau}_t))_\xi}\right]^{-\frac{1}{k}}.$$

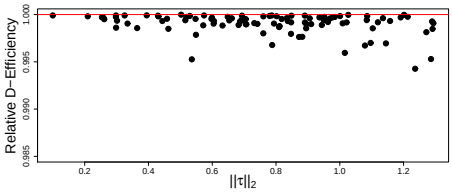
For the Latin square design example, we consider the following two cases of assumed values θ_c for model parameters. For each case the values of parameters are simulated from a uniform distribution. The range of uniform distribution is obtained by ± 1 and ± 2 from true parameter values θ_t for each case, respectively. Here we consider $\theta_t = [0.5, 0.06, -0.53, -0.6, -0.35, 0.025, -0.23, 0.73, 0.23, 0.30]$.

Table 1.8: Assumed values for model parameters

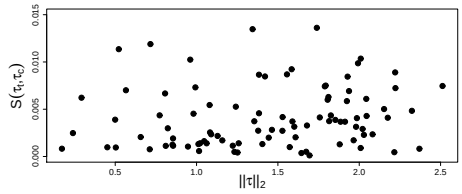
Parameters θ_c	Case 1	Case 2
λ	U$(-0.5, 1.5)$	U$(-1.5, 2.5)$
β_2	U$(-0.04, 0.16)$	U$(-0.14, 0.26)$
β_3	U$(-1.53, 0.47)$	U$(-2.53, 1.47)$
β_4	U$(-1.6, 0.4)$	U$(-2.6, 1.4)$
τ_2	U$(-1.35, 0.65)$	U$(-2.35, 1.65)$
τ_3	U$(-0.075, 0.125)$	U$(-0.175, 0.225)$
τ_4	U$(-1.23, 0.77)$	U$(-2.23, 1.77)$
ρ_2	U$(-0.27, 1.73)$	U$(-1.27, 2.73)$
ρ_3	U$(-0.77, 1.23)$	U$(-1.77, 2.23)$
ρ_4	U$(-0.70, 1.30)$	U$(-1.70, 2.30)$



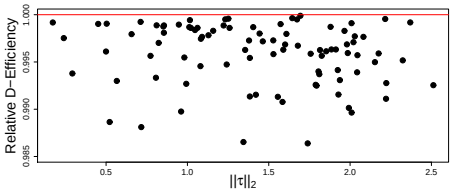
Case 1: Relative loss of efficiency



Case 1: Relative D-efficicency



Case 2: Relative loss of efficiency



Case 2: Relative D-efficicency

Figure 1.9: Performance of the locally optimal designs

1.4.3 Simulation Studies with Two-Stage Designs

As stated earlier the main aim of this thesis is to determine optimal and efficient crossover designs for experiments where the GLMs adequately describes the process under study. Crossover trials are repeated measurement designs, where repeated measurements on the same subject have great advantages, but there are also many potential disadvantages associated with them. Nevertheless, the impact of these disadvantages can be minimized or reduced if we choose a proper design and analysis method. One of the major disadvantages of repeated measurement designs is that the effect of the treatment depends on the subject itself. Stronger subject effects cause more variation in estimated treatment effects.

The simulation studies are motivated by the real-life example of Latin square design mentioned above. Since all the correlation structures mentioned in Section 1.2.3 perform similarly in Table 1.7, we choose $Corr(2)$ for illustration purposes. Note that in $Corr(2)$, we have an AR(1) structure, where the correlation between two responses decreases as the number of periods between responses increases, which makes good practical sense. For these simulation studies, we are considering 400 observations and two different types of initial guesses for θ values. In Case 1 we will use $\theta_2 = [0.5, 0.06, -0.53, -0.6, -0.35, 0.025, -0.23, 0.73, 0.23, 0.30]$ which is obtained from real data. This choice of θ_2 gives optimal allocations as $(0.2460, 0.2495, 0.2500, 0.2545)$, which is approximately uniform. For Case 2 we will use $\theta_1 = [-2, 0.25, 0, 0.75, 1, 5, -1.5, -3.5, 2.75, 0.75]$ and this guess of θ_1 is such that optimal allocations are non-uniform. For

example, for $\rho = 0.1$ the optimal allocations are $(0.172, 0.248, 0.222, 0.358)$. Optimal allocations are similar for other values of ρ.

The simulation process used here has two stages. First for a given parameter θ, we define a design matrix corresponding to each treatment sequence along with a correlation matrix.

- First Stage:

    1. In this stage, we use the `rbin` function in **R** to simulate 30% of observations uniformly over all four treatment sequences. These observations serve as our pilot study. Note that we use a uniform design for a pilot study.
    2. From these observations obtained in the above step, we estimate the correlation coefficient and regression parameters, which are used as the assumed parameter values for the second stage.

- Second Stage:

    1. Based on the assumed parameter values obtained in the first stage and the algorithm described in Section 2.4, we calculate the optimal allocation for the remaining 70% of the subjects.
    2. Using these optimal allocations, we simulate observations for the remaining 70% of subjects according to the assumed parameter values.
    3. In case of uniform design, we simulate a total number of observations uniformly over all treatment sequences, i.e., one-fourth of the total observations correspond to each of the four treatment sequence.

During this process, we calculate the parameter estimates based on the simulated observations and calculate the corresponding Mean Square Error (MSE) from the true parameter values for each simulation. The above simulation procedure is repeated 100 times. Finally, we take the average of those individual MSEs to calculate the overall MSE reported in Table 1.9. We repeat the above simulation process for different correlation coefficients and for two different sets of initial θ's, θ_1 and θ_2. It is clear from Table 1.9 and Figure 10 that if the optimal allocations are non-uniform, then the proposed optimal design has a significant advantage over the traditional uniform designs, for all values of the correlation coefficients. It should be noted that those high values of MSEs for

uniform designs are mostly due to a handful of "bad" datasets. In our experience, the proposed optimal designs never give rise to such data.

Table 1.9: Simulation Results

Correlation Structure Corr(2)	Mean Squared Errors			
	Case 1		Case 2	
ρ	Uniform Design	Optimal Design	Uniform Design	Optimal Design
0.1	0.109	0.108	2.834	0.393
0.2	0.103	0.100	2.718	0.659
0.3	0.101	0.140	4.925	0.490
0.4	0.094	0.127	4.896	0.484
0.5	0.100	0.123	2.596	0.428
0.6	0.088	0.109	2.632	0.469
0.7	0.086	0.095	5.110	0.458
0.8	0.066	0.077	2.705	0.586
0.9	0.050	0.051	2.761	0.559

1.5 Discussion

In practice, it is customary to use uniform designs where the same number of subjects are assigned to each treatment sequence. In the case of linear models, such uniform designs are optimal. However, optimal proportions obtained under GLMs are not uniform. We identified locally optimal designs under different correlation structures. Tables 1.1 to 1.4 and graphs in Figures 1.1 to 1.8 suggest that the optimal proportions do not vary much from one correlation structure to another. These results suggest that the identified designs are robust. Simulation studies and results in Table 1.9 and Figure 1.10 suggest that these designs are more efficient than uniform designs as well.

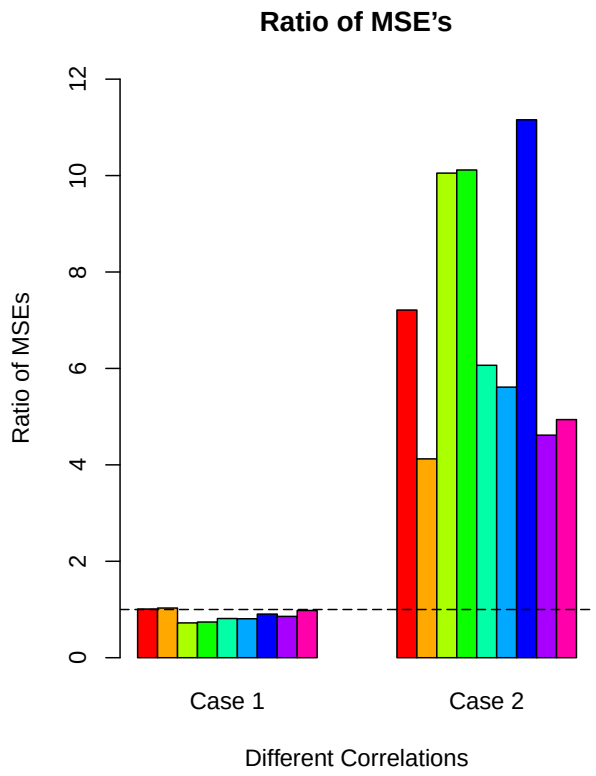


Figure 1.10: Simulation Results: Ratios of the MSEs of the Uniform versus optimal deigns, for different values of ρ, for each of the two cases.

Chapter 2

Optimal Crossover Designs for GLMs: An Application to Work Environment Experiment

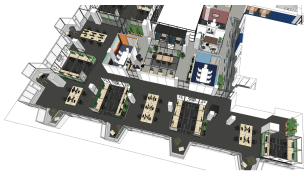
2.1 The Work Environment Experiment

We considered the data obtained from the work environment experiment conducted at Booking.com (Pitchforth et al., 2020). In recent years, most corporate offices and organizations have adopted open office spaces over the traditional cubicle office spaces. Since there were no previous studies to examine the effects of office designs in workspaces, Booking.com conducted an experiment to assess different office spacing efficiency.

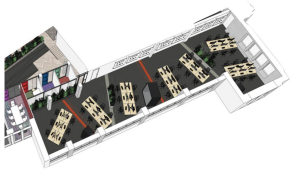
In the work environment experiment, there were a total of $n = 288$ participants. These participants were divided into four groups, G_1, G_2, G_3, G_4, with each group having an equal number (72) of individual participants. This experiment is essentially an uniform crossover design with $p = 4$ periods and $t = 4$ treatments. Periods were named Wave1, Wave2, Wave3 and Wave4, where each Wave had a duration of 2 weeks. The four treatments involved in this experiment are office designs named as A (Activity-Based), B (Open Plan), C (Team Offices), and D (Zoned Open Plan), as shown in the figure below:

$A-$ ACT

$B-$ OPEN



$C-$ TEAM



$D-$ ZONE.

The images are reproduced from the manuscript (Pitchforth et al., 2020), under Creative Commons Attribution license (https://creativecommons.org/licenses/by/4.0/).

During the experiment, each group is exposed to different treatments over different periods depending on the treatment sequence. At a given period, there was no interaction between subjects from different groups. A Latin square design (C. J. Wu & Hamada, 2011) of order four has been used to decide the sequence of exposure so that no group was exposed to the conditions in the same order as any other group. The design is shown below in Table 2.1. A total of $m = 23$ covariates were involved in the experiment, but we consider only the most important ones in our fitted model.

Table 2.1: Latin square design

Groups $\Rightarrow$ Period $\Downarrow$	G_1	G_2	G_3	G_3
Wave 1	OPEN	TEAM	ZONE	ACT
Wave 2	ACT	ZONE	OPEN	TEAM
Wave 3	ZONE	ACT	TEAM	OPEN
Wave 4	TEAM	OPEN	ACT	ZONE

In the following analysis, we consider three different responses that were recorded during the experiment. We discuss these responses in more detail in the following sections. These three responses follow three different types of

distributions. We make an extra assumption that the responses from a particular subject are mutually correlated, while the responses from different subjects are uncorrelated. To capture the dependency among the observations coming from the same subject, we calculate optimal proportions for these different responses using six different correlation structures proposed in Section 2.3 of (Jankar et al., 2020) and shown in the Appendix A.1. For each correlation matrix that we consider, an optimal design ζ^* is the one minimizing the determinant of $\mathrm{Var}(\hat{\tau})$ in equation (1.6) with respect to p_ω such that $\sum_{w\in\Omega} p_w = 1$.

We use different colors to represent different correlation structures. The color scheme that we use is as follows:

Correlation Structure	Color
Corr(1)	
Corr(2)	
Corr(3)	
Corr(4)	
Corr(5)	
Corr(6)	

2.2 Poisson Regression

In the case of Poisson response, we calculate the locally optimal design for the above example under the model,

$$\log(\mu_{ij}) = \eta_{ij} = \lambda + \beta_i + \tau_{d(i,j)} + \rho_{d(i-1,j)}, \tag{2.1}$$

where notations have the same meaning as in equation (1.2). In the above experiment, there were many different types of responses recorded. We consider the response $commit\ count$ to illustrate the optimal crossover design for the Poisson response. The commit counts were the number of commits submitted to the main git repository.

2.2.1 Analysis of data

We consider the three main predictors in the model, which are $area$, $wave$ and $carryover$. Here, $area$ corresponds to the direct treatment effect, $wave$ corresponds to the period effect, and $carryover$ corresponds to the carryover

effect of a treatment given in a previous period. We use different kinds of correlation matrices and calculate the optimal proportions. As mentioned earlier we consider baseline constraints as $\beta_1 = \tau_1 = \rho_1 = 0$, so that all the parameters are estimable.

We fit the Poisson regression model to the commit data by using the `glm` function in **R** and calculate the parameter estimates. We use these parameter estimates to make a guess for values of unknown parameters. Our nominal guess for the parameter values is $\theta_1 = [2, 0.3, 0.8, -0.1, -0.2, 0.04, -0.2, -0.6, 0.15, -0.4]$. It is interesting to note that carryover effects are larger than direct effects, even though θ_1 is calculated using experimental data. Now, we calculate the optimal designs for different correlation structures by minimizing the objective function. We also calculate optimal proportions for another parameter $\theta_2 = [2, 0.3, 0.8, -0.1, -2.0, 0.40, -2.0, -1.0, 0.30, -1.0]$, which is significantly different from θ_1.

2.2.2 Optimal designs

In Table 2.2, we present the optimal proportions corresponding to Poisson response for six different choices of the correlation matrix.

Table 2.2: Optimal proportions in case of Poisson response.

Correlation	θ_1				θ_2			
Structure	$BADC$	$CDAB$	$DBCA$	$ACBD$	$BADC$	$CDAB$	$DBCA$	$ACBD$
$Corr(1)$	0.2500	0.2500	0.2500	0.2500	0.2500	0.2500	0.2500	0.2500
$Corr(2)$	0.2500	0.2500	0.2500	0.2500	0.2747	0.3113	0.1841	0.2299
$Corr(3)$	0.2500	0.2500	0.2500	0.2500	0.2795	0.3074	0.1798	0.2333
$Corr(4)$	0.2500	0.2500	0.2500	0.2500	0.2562	0.3168	0.1860	0.2410
$Corr(5)$	0.2500	0.2500	0.2500	0.2500	0.2736	0.3138	0.1922	0.2204
$Corr(6)$	0.2500	0.2500	0.2500	0.2500	0.2537	0.3190	0.1844	0.2429

As seen from Table 2.2, in the case of Poisson response, the optimal proportions that we obtain using θ_1 are nearly uniform and that using θ_2 are non-uniform.

The plots in Figures 2.1 and 2.2 represent the optimal proportions for Poisson response under θ_1 and θ_2 respectively. It can be seen from these plots that the

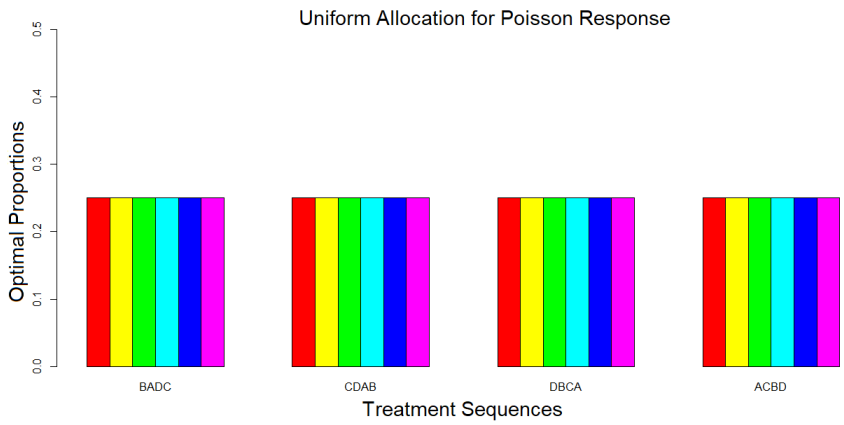


Figure 2.1: Uniform optimal proportions for Poisson response under θ_1

optimal proportions do not vary much when we use different correlation structures under θ_1 and θ_2. In practice, uniform optimal designs (the same proportion for each treatment sequence) are often used. It is clear from the above analysis that those uniform designs are sub-optimal under θ_2.

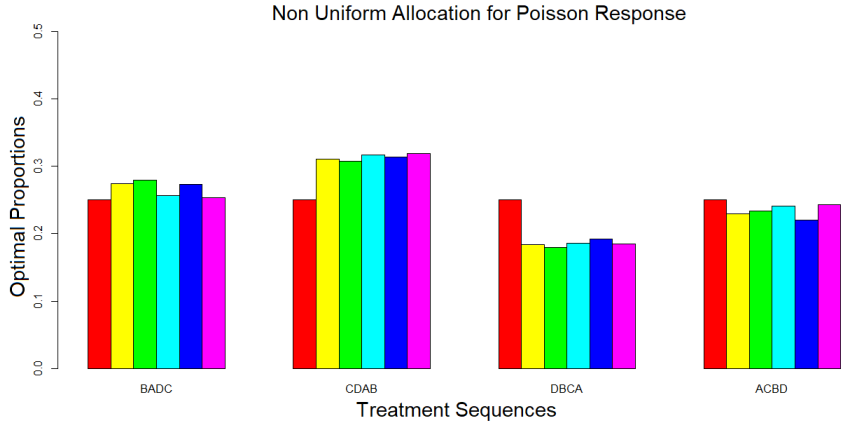


Figure 2.2: Non-uniform optimal proportions for Poisson response under θ_2

2.3 Beta Regression

In the beta response case, we calculate the locally optimal design for the response from the Booking.com example under two different models. We consider two different link functions to model the marginal mean of the response as follows:

$$\text{logit}(\mu_{ij}) = \log(\frac{\mu_{ij}}{1-\mu_{ij}}) = \eta_{ij} = \lambda + \beta_i + \tau_{d(i,j)} + \rho_{d(i-1,j)}, \qquad (2.2)$$

and,

$$\log(\mu_{ij}) = \eta_{ij} = \lambda + \beta_i + \tau_{d(i,j)} + \rho_{d(i-1,j)}, \qquad (2.3)$$

where notations have the same meaning as in equation (1.2).

To illustrate the optimal proportions in the beta response case, we consider the normalized response $engagement$ from the work environment experiment. In the case of this experiment, $engagement$ is a measure of the extent to which participants felt focused on and excited to complete regular work tasks.

2.3.1 Analysis of data

Similar to the Poisson response analysis, we consider three main predictors in the model for a beta response: $area$, $wave$, and $carryover$. As mentioned above, we use six different kinds of correlation matrices and calculate optimal proportions under two different models with different link functions. As mentioned earlier, we consider baseline constraints so that all the parameters are estimable.

We get the initial estimates of parameters by fitting the beta regression model to the response. For two different link functions we need to guess two different sets of parameter values for θ_1 and θ_2. In case of $logit$ link function, our nominal guess for the parameter values is θ_1 = [1.24, -0.035, 0.17, 0.078, -0.2, -0.3, 0.01, -0.35, -0.62, -0.329] and θ_2 = [1.24, -0.035, 0.17, 0.078, -4, -6, 2, -3.5, -3.1, -1.28]. In case of log link function, our nominal guess for the parameter values is θ_1 = [-0.25, -0.01, 0.04, 0.02, -0.05, -0.08, -0.004, -0.088, -0.172, -0.08] and θ_2 = [-0.25, -0.01, 0.04, 0.02, -5, -8, -0.4, -2.2, -4.3, -2]. Note that, as before, θ_1 is an educated guess based on the data at hand, whereas θ_2 has significantly different values for the parameters of interest than those of θ_1.

2.3.2 Optimal designs

In Table 2.3, we present the optimal proportions corresponding to the $logit$ link case for six different choices of correlation matrix. As seen from Table 2.3, in the case of beta response ($logit$ link), the optimal proportions that we obtain using θ_1 are nearly uniform and that using θ_2 are non-uniform.

Table 2.3: Optimal proportions in case of beta response ($logit$ link).

Correlation	θ_1				θ_2			
Structure	$BADC$	$CDAB$	$DBCA$	$ACBD$	$BADC$	$CDAB$	$DBCA$	$ACBD$
$Corr(1)$	0.2518	0.2563	0.2465	0.2454	0.3418	0.2085	0.1643	0.2854
$Corr(2)$	0.2525	0.2572	0.2453	0.2450	0.3316	0.2066	0.1690	0.2928
$Corr(3)$	0.2515	0.2568	0.2462	0.2455	0.3363	0.2058	0.1682	0.2897
$Corr(4)$	0.2405	0.2539	0.2419	0.2637	0.3205	0.2043	0.1739	0.3013
$Corr(5)$	0.2595	0.2542	0.2467	0.2396	0.3250	0.2070	0.1711	0.2969
$Corr(6)$	0.2366	0.2562	0.2423	0.2649	0.3218	0.2088	0.1668	0.3026

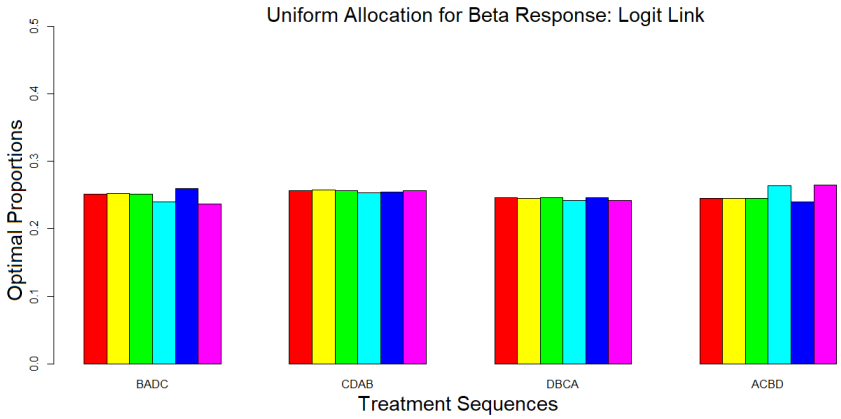


Figure 2.3: Uniform optimal proportions for beta response ($logit$ link) under θ_1

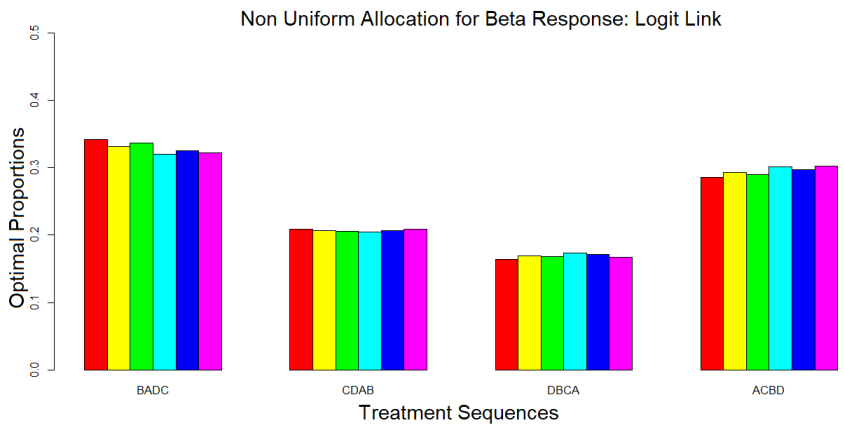


Figure 2.4: Non-uniform optimal proportions for beta response ($logit$ link) under θ_2

In Table 2.4, we present the optimal proportions corresponding to the log link case for six different choices of the correlation matrix. As before, in the beta response (log link) case, the optimal proportions that we obtain using θ_1 are nearly uniform and that using θ_2 are non-uniform.

The plots is Figures 2.3, 2.4 and Figures 2.5, 2.6 represent the optimal proportions for beta response under θ_1 and θ_2 for two different choices of link functions respectively.

Table 2.4: Optimal proportions in case of beta response (log link).

Correlation	θ_1				θ_2			
Structure	$BADC$	$CDAB$	$DBCA$	$ACBD$	$BADC$	$CDAB$	$DBCA$	$ACBD$
$Corr(1)$	0.2522	0.2560	0.2470	0.2448	0.3305	0.1470	0.1930	0.3295
$Corr(2)$	0.2529	0.2569	0.2458	0.2444	0.3270	0.1200	0.2084	0.3446
$Corr(3)$	0.2520	0.2564	0.2466	0.2450	0.3290	0.1210	0.2050	0.3450
$Corr(4)$	0.2410	0.2535	0.2425	0.2630	0.3271	0.1060	0.2137	0.3532
$Corr(5)$	0.2600	0.2540	0.2460	0.2400	0.3245	0.1101	0.2102	0.3552
$Corr(6)$	0.2371	0.2558	0.2428	0.2643	0.3272	0.1096	0.2120	0.3512

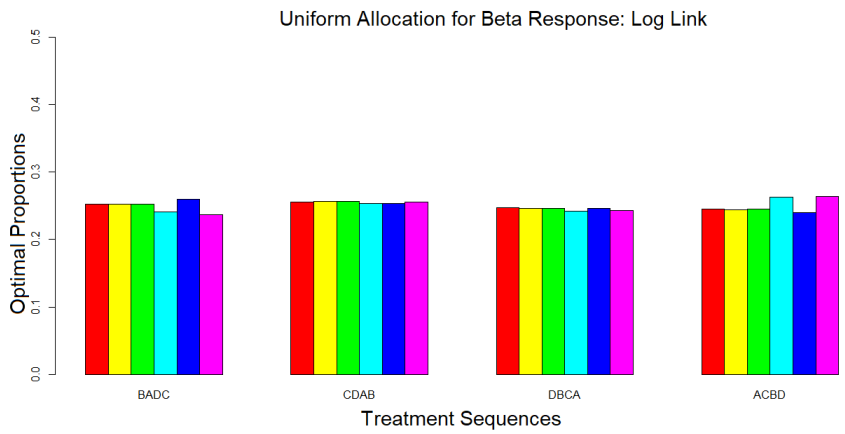


Figure 2.5: Uniform optimal proportions for beta response (*log* link) under θ_1

It can be seen from these plots that optimal proportions do not vary much when we use different correlation structures under θ_1 and θ_2. In most of the situations in practice uniform optimal designs are used. The above analysis shows that those uniform designs are sub-optimal under θ_2 irrespective of what link function is used.

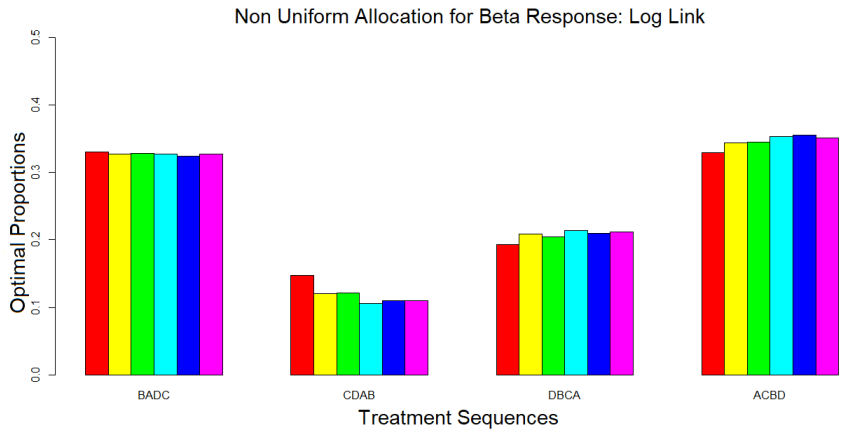


Figure 2.6: Non-uniform optimal proportions for beta response (*log* link) under θ_2

2.4 Gamma Regression

In the case of Gamma response, we calculate locally D-optimal design for the response from the same Booking.com example under two different models. Similar to the beta response, we consider two different link functions to model the marginal mean of the response. We use the log, and $inverse$ link functions, and the two models are as follows:

$$\log(\mu_{ij}) = \eta_{ij} = \lambda + \beta_i + \tau_{d(i,j)} + \rho_{d(i-1,j)},$$

and,

$$inv(\mu_{ij}) = \frac{1}{\mu_{ij}} = \eta_{ij} = \lambda + \beta_i + \tau_{d(i,j)} + \rho_{d(i-1,j)}, \tag{2.4}$$

where, as before, notations have the same meaning as in equation (1.2).

From the work environment experiment, we consider the response *satisfaction*. Satisfaction is an essential concept for organizational and office design research, and it is usually used to measure employees' sentiments. In the work environment experiment, the Leesman satisfaction index was used, which is useful for many benchmark purposes. Since the response is right-skewed, it is safe to assume that the response follows a gamma distribution.

2.4.1 Analysis of data

Similar to the previous two cases, we consider three main predictors in the model for gamma response, which are $area$, $wave$ and $carryover$. As before, we consider six different kinds of correlation matrices and calculate optimal proportions under two different models with different link functions. We consider the same baseline constraints as mentioned earlier. We fit the gamma regression model to the data with $satisfaction$ as response by using the `glm` function in **R** and calculate the parameter estimates.

In case of log link function, our nominal guess for the parameter values is $\theta_1 = [2.1, -0.19, -0.04, -0.04, -0.16, -0.4, -0.06, 0.05, 0.005, -0.05]$ and $\theta_2 = [2.1, -0.19, -0.04, -0.04, -1.6, -4.0, -0.6, 0.5, 0.05, -0.5]$. In case of $inverse$ link function, our nominal guess for the parameter values is $\theta_1 = [0.13, 0.03, 0.003, 0.003, 0.025, 0.07, 0.008, -0.007, -0.0001, -0.01]$ and $\theta_2 = [0.13, 0.03, 0.003, 0.003, 2.5, 7, 0.8, -0.7, -0.01, -1]$. As before,

θ_1 was motivated by the data provided by (Pitchforth et al., 2020) and θ_2 is significantly different from θ_1.

2.4.2 Optimal designs

In the Table 2.5, we present the optimal proportions corresponding to log link case for six different choices of correlation matrix. As seen from Table 2.5, in case of gamma response (log link) the optimal proportions that we obtain using θ_1 are nearly uniform and that using θ_2 are non-uniform.

Table 2.5: Optimal proportions in case of gamma response (log link).

Correlation	θ_1				θ_2			
Structure	$BADC$	$CDAB$	$DBCA$	$ACBD$	$BADC$	$CDAB$	$DBCA$	$ACBD$
$Corr(1)$	0.2500	0.2500	0.2500	0.2500	0.1328	0.2775	0.3336	0.2561
$Corr(2)$	0.2500	0.2500	0.2500	0.2500	0.1248	0.2639	0.3527	0.2586
$Corr(3)$	0.2500	0.2500	0.2500	0.2500	0.1258	0.2582	0.3596	0.2564
$Corr(4)$	0.2500	0.2500	0.2500	0.2500	0.1206	0.2671	0.3451	0.2672
$Corr(5)$	0.2500	0.2500	0.2500	0.2500	0.1225	0.2770	0.3354	0.2656
$Corr(6)$	0.2500	0.2500	0.2500	0.2500	0.1195	0.2685	0.3416	0.2704

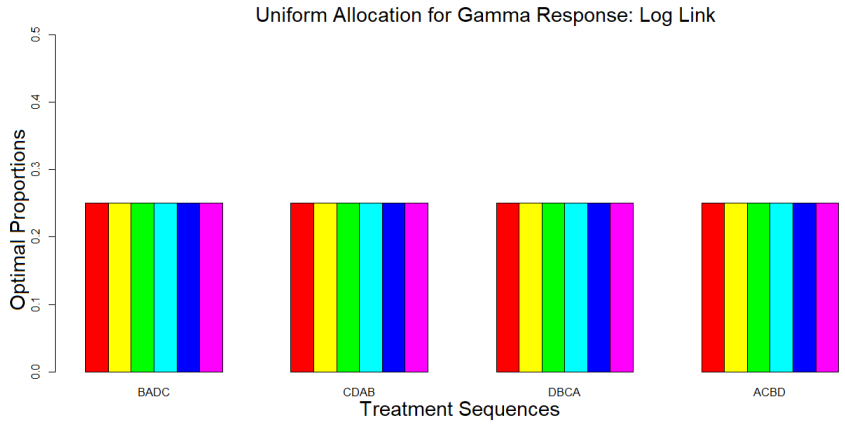


Figure 2.7: Uniform optimal proportions for gamma response (log link) under θ_1

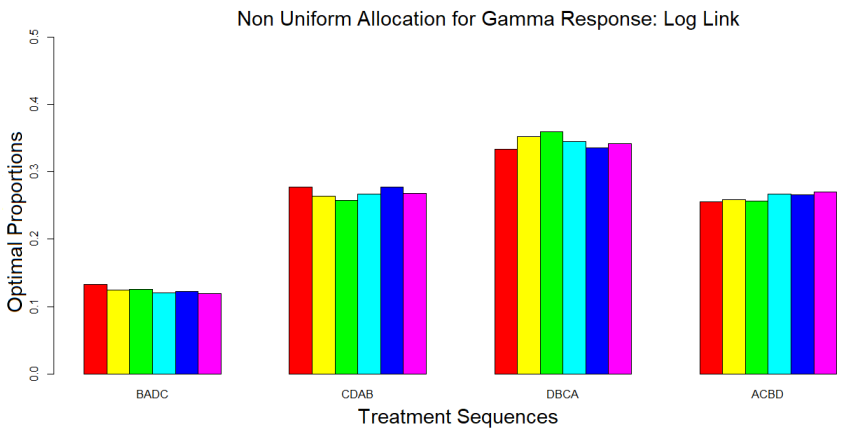


Figure 2.8: Non-uniform optimal proportions for gamma response (log link) under θ_2

In Table 2.6, we present the optimal proportions corresponding to the $inverse$ link case for six different choices of correlation matrix. As before, Table 2.6 indicates that the optimal proportions that we obtain using θ_1 are nearly uniform and that using θ_2 are non-uniform in the case of gamma response ($inverse$ link).

The plots in Figures 2.7, 2.8 and Figures 2.9, 2.10 represent the optimal proportions for gamma response under θ_1 and θ_2 for two different choices of link functions respectively.

Table 2.6: Optimal proportions in case of gamma response ($inverse$ link).

Correlation	θ_1				θ_2			
Structure	$BADC$	$CDAB$	$DBCA$	$ACBD$	$BADC$	$CDAB$	$DBCA$	$ACBD$
$Corr(1)$	0.2500	0.2500	0.2500	0.2500	0.2650	0.3093	0.1828	0.2429
$Corr(2)$	0.2500	0.2500	0.2500	0.2500	0.2486	0.3031	0.1911	0.2572
$Corr(3)$	0.2500	0.2500	0.2500	0.2500	0.2588	0.3051	0.1879	0.2482
$Corr(4)$	0.2500	0.2500	0.2500	0.2500	0.2389	0.3087	0.1784	0.2740
$Corr(5)$	0.2500	0.2500	0.2500	0.2500	0.2406	0.3112	0.1762	0.2720
$Corr(6)$	0.2500	0.2500	0.2500	0.2500	0.2421	0.3146	0.1740	0.2729

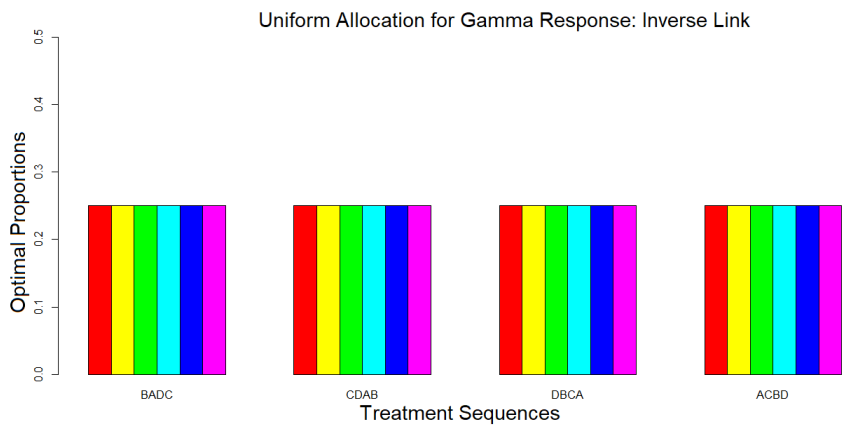


Figure 2.9: Uniform optimal proportions for gamma response (inv link) under θ_1

It can be seen from these plots that optimal proportions do not vary much when we use different correlation structures under θ_1 and θ_2. In most of the situations in practice uniform optimal designs are used. The above analysis shows that those uniform designs are sub-optimal under θ_2 irrespective of what link function is used.

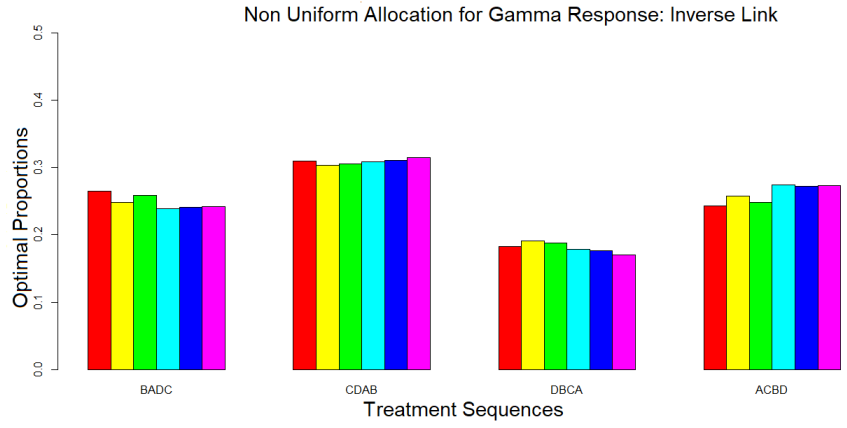


Figure 2.10: Non-uniform optimal proportions for gamma response (inv link) under θ_2

2.5 Discussion

In many experiments in real life, uniform designs are often used. Uniform designs are those in which the same number of subjects is assigned to each treatment sequence. These uniform designs are optimal in the linear model case, *i.e.* when the response is normally distributed. But, in Chapter 1 we show that the obtained optimal proportions are not necessarily uniform, in situations where responses are non-normal. In this chapter, we identify locally optimal designs for responses belonging to Poisson, beta and gamma distributions. Two different link functions were considered in the case of beta and gamma responses. Tables 2.2 to 2.6 and plots in Figures 2.1 to 2.10 suggest that the obtained optimal proportions are robust for different choices of correlation structures. These results also suggest that uniform designs are sub-optimal under θ_2 irrespective of the link function used or the response's distribution. Note that we are using the local optimality approach of (Chernoff, 1953). In real experiments, it is not always possible to guess the values of parameter estimates from prior knowledge. In that case, it is not easy to obtain locally optimal designs. In this thesis, we consider approximate designs in terms of optimal proportions. While conducting real-life experiments, the practitioners must use exact designs where these proportions are to be converted into integers for determining the replication numbers of the sequences. The rounding off error might be insignificant unless the total number of observations is large. The Work Environment Experiment had 288 subjects and hence such issues do not arise.

Chapter 3

A General Equivalence Theorem for Crossover Designs under GLMs

3.1 Overview

Over the years, optimal crossover designs for normal responses have been widely studied in the literature, however, there are several examples in real life where responses are not normal and described by GLMs. In Chapter 1 we provided an algorithm to search locally D-optimal crossover designs in the case of non-normal response, and showed that optimal designs obtained for normal responses can be quite inefficient in the case of GLMs. But there is no guarantee that the designs obtained by the algorithm were indeed optimal. In this chapter, we derive a general equivalence theorem specifically for crossover designs under GLMs, which can be used to verify the optimality of proposed designs. Moreover, it provides an alternative that is faster and numerically more stable than the general algorithm proposed in Chapter 1.

The General Equivalence Theorem is an important tool in optimum experimental designs, which has been widely used for checking the optimality of designs in terms of the Fisher information matrix (Atkinson et al., 2007; Fedorov, 1971, 1972; Fedorov & Leonov, 2014; Fedorov & Malyutov, 1972; Kiefer & Wolfowitz, 1960; Whittle & Malyutov, 1973). Nevertheless, the traditional equivalence theorem does not apply to check the optimality of the obtained crossover designs. The optimal crossover designs under GLMs discussed in Chapter 1 are identified using GEE and are based on the variance matrix of the parameters of interest. Since the variance matrix is asymptotically connected with the inverse of the Fisher information matrix, it is natural to derive a condi-

tion that can be used to check the optimality of designs (see remarks below for more details).

For illustration purposes, we consider two real-life motivating examples. First, we consider an experiment conducted at Booking.com (discussed in Chapter 2) to determine the optimal office design. In addition to that, we discuss another motivating example in an experiment conducted to investigate the effects of various dietary starch levels on milk production in cows. (Kenward & Jones, 1992) discussed this dietary example along with the data set used for analysis (for more details see (Kenward & Jones, 2014)). The design used in both these examples is a 4×4 Latin Square design with four periods and four treatments.

This chapter is organized as follows: to set up ideas, we describe notation and definitions for crossover designs in Section 3.2. In Section 3.3 we propose and derive two different versions of the general equivalence theorem for crossover designs. More specifically, in Section 3.3.1 we use the variance of all parameter estimates as an objective function, and in Section 3.3.2 we use the variance of treatment effects as an objective function to derive two versions of the theorem. We present an illustration in Section 3.3.3 and real-life motivating examples in Section 3.4.

3.2 Notation and Preliminaries

A crossover design ξ in approximate theory is specified by the set $\{p_\omega, \omega \in \mathbf{\Omega}\}$, where $p_\omega = n_\omega/n$ is the proportion of subjects assigned to treatment sequence ω. As denoted by (Silvey, 1980), such a crossover design ξ can be written as follows:

$$\xi = \left\{ \begin{array}{cccc} \omega_1 & \omega_2 & \dots & \omega_k \\ p_1 & p_2 & \dots & p_k \end{array} \right\}, \tag{3.1}$$

where k is the number of treatment sequences involved, ω_i is the i^{th} treatment sequence and p_i is the corresponding proportion of units allocated to that support point, such that $\sum_{i=1}^{k} p_i = 1,$ for $i = 1, \dots, k$. Note in Chapter 1 we observed that, in the case of non-uniform allocations, only a few sequences have non-zero proportions. Hence, in our illustrations, we consider Ω to be the collection of only those sequences that have non-zero allocations.

GEE are quasi-likelihood equations that allow us to estimate quasi-likelihood estimators (Liang et al., 1988; Prentice, 1988). In crossover trials, it is typical to assume that the observations from the same subject are correlated while the

observations from different subjects are independent (Kenward & Jones, 2014). This dependency among repeated observations from the same subject can be modeled by the "working correlation" matrix $\boldsymbol{C_\alpha}$, which is a function of the correlation coefficient α. If $\boldsymbol{C_\alpha}$ is the true correlation matrix of $\boldsymbol{Y_j}$, then from the definition of covariance, we can write

$$Cov(\boldsymbol{Y_j}) \;\; = \;\; \boldsymbol{D_j}^{1/2}\boldsymbol{C_\alpha}\boldsymbol{D_j}^{1/2},$$

where $\boldsymbol{D_j} = diag\Big(Var(Y_{1j}), \ldots, Var(Y_{pj})\Big)$. Let us denote $Cov(\boldsymbol{Y_j})$ by $\boldsymbol{W_j}$. In (Liang et al., 1988) (equation (3.1)) it was shown that for repeated measurement models, the GEE are defined to be

$$\sum_{j=1}^{n} \frac{\partial \boldsymbol{\mu_j}'}{\partial \boldsymbol{\theta}} \boldsymbol{W_j}^{-1} (\boldsymbol{Y_j} - \boldsymbol{\mu_j}) = 0,$$

where $\boldsymbol{\mu_j} = (\mu_{1j}, \ldots, \mu_{pj})'$ and the asymptotic variance for the GEE estimator $\hat{\boldsymbol{\theta}}$ (see (Liang et al., 1988), equation (3.2)) is

$$\mathrm{Var}(\hat{\boldsymbol{\theta}}) \;\; = \;\; \left[\sum_{j=1}^{k} np_j \frac{\partial \boldsymbol{\mu_j}'}{\partial \boldsymbol{\theta}} \boldsymbol{W_j}^{-1} \frac{\partial \boldsymbol{\mu_j}}{\partial \boldsymbol{\theta}}\right]^{-1} = \boldsymbol{M}^{-1}, \qquad (3.2)$$

where $\frac{\partial \boldsymbol{\mu_j}'}{\partial \boldsymbol{\theta}} = \boldsymbol{X_j}'\mathrm{diag}\,\{(g^{-1})'(\eta_{1j}), \ldots, (g^{-1})'(\eta_{pj})\}$ and j stands for the j^{th} treatment sequence. In Section 3.3, we will define $\boldsymbol{M}$ explicitly for crossover designs. Later, we consider the situation where direct treatment effects are studied specifically.

Remark: The general equivalence theorem describes the optimality criteria in terms of the Fisher information matrix. The information matrix for optimal crossover designs under GLMs is defined as the inverse of the variance-covariance matrix of parameters of interest through GEE, which is easier to obtain and works similarly to the Fisher information matrix. Here we assume that the responses from a particular subject are mutually correlated, while the responses from different subjects are uncorrelated. In Chapter 1, we observe that the obtained optimal designs are robust to the choices of such working correlation matrices.

As mentioned in (Atkinson et al., 2007), the general equivalence theorem can be viewed as a consequence of the result that the derivative of a smooth function over an unconstrained region is zero at its minimum. In this thesis, we derive the general equivalence theorem for crossover designs by calculating the directional derivative of an objective function $\Phi(\xi)$ expressed in terms of $\boldsymbol{M}(\xi)$.

Consider $\bar{\xi_i}$ to be the design that puts unit mass at the point x_i, i.e., the design supported only at x_i, where $i = 1, 2, \ldots, k$. Let $\xi_i' = (1-h)\xi + h\bar{\xi_i}$. Then the derivative of $\Phi(\xi)$ in the direction $\bar{\xi_i}$ or x_i in case of D-optimal criterion is

$$\phi(x_i, \xi) = \lim_{h \to 0^+} \frac{1}{h}[\Phi(\xi_i') - \Phi(\xi_i)] = -\lim_{h \to 0^+} \frac{1}{h}[\ln det(\boldsymbol{M}(\xi_i')) - \ln det(\boldsymbol{M}(\xi_i))],$$

and ξ is D-optimal if and only if $min_i \phi(x_i, \xi) = 0$ and $\phi(x_i, \xi) = 0$ if $p_{\omega_i} > 0$, where this minimum is occurring at the points of support of design.

In the case of crossover designs and estimates using GEE, a different approach compared to the one mentioned above is needed, as the design points are finite and pre-specified for crossover designs. We use the technique used in the supplement materials of (J. Yang et al., 2016). Instead of using $\xi_i' = (1-h)\xi + h\bar{\xi_i} = \xi + h(\bar{\xi_i} - \xi)$, they used $\boldsymbol{p_r} + u\boldsymbol{\delta_i^{(r)}}$, where $\boldsymbol{p_r}$ and $\boldsymbol{\delta_i^{(r)}}$ are defined below. Therefore, the directional derivative $\phi(u, \boldsymbol{p_r})$ of the objective function is equal to $\left.\frac{\partial \Phi(\boldsymbol{p_r} + u\boldsymbol{\delta_i^{(r)}})}{\partial u}\right|_{u=0}$.

Here is the outline of the general equivalence theorem in the case of crossover designs. Note that $0 \leq p_i < 1$ for $i = 1, \ldots, k$, and since $\sum_{i=1}^{k} p_i = 1$ we may assume without any loss of generality that $p_k > 0$. Define $\boldsymbol{p_r} = (p_1, \ldots, p_{k-1})'$, and $\Phi(\boldsymbol{p_r}) = -\ln det(\boldsymbol{M}(p_1, \ldots, p_{k-1}, 1 - \sum_{i=1}^{k-1} p_i))$. Let $\boldsymbol{\delta_i^{(r)}} = (-p_1, \ldots, -p_{i-1}, 1 - p_i, -p_{i+1}, \ldots, -p_{k-1})'$ for $i = 1, \ldots, k-1$. $\boldsymbol{\delta_i^{(r)}}$ are defined in such a way that the determinant $|(\boldsymbol{\delta_1^{(r)}}, \ldots, \boldsymbol{\delta_{k-1}^{(r)}})| = p_k \neq 0$. Hence, $\boldsymbol{\delta_1^{(r)}}, \ldots, \boldsymbol{\delta_{k-1}^{(r)}}$ are linearly independent and thus can serve as the new basis of

$$\boldsymbol{S_r} = \{(p_1, \ldots, p_{k-1})' | \sum_{i=1}^{k-1} p_i < 1, \text{and } p_i \geq 0, i = 1, \ldots, k-1\}.$$

Note that negative $\ln det$ is a convex function on a set of positive definite matrices. Hence, $\boldsymbol{p_r}$ minimizes $\Phi(\boldsymbol{p_r})$ if and only if along each direction $\boldsymbol{\delta_i^r}$,

$$\left.\frac{\partial \Phi(\boldsymbol{p_r} + u\boldsymbol{\delta_i^{(r)}})}{\partial u}\right|_{u=0} \begin{cases} = 0 \text{ if } p_i > 0 \\ \geq 0 \text{ if } p_i = 0 \end{cases}$$

3.3 Equivalence Theorems for Crossover Designs

As defined earlier, $\boldsymbol{C_\alpha}$ is the working correlation matrix and hence is a positive definite and symmetric matrix. So, there exists a square matrix $\boldsymbol{R}$ such that

$\boldsymbol{C_\alpha}^{-1} = \boldsymbol{R}^T\boldsymbol{R}$. Then the inverse of the variance of the parameter estimates through GEE is as follows:

$$\boldsymbol{M} = \sum_{j=1}^{k} np_j \frac{\partial \boldsymbol{\mu_j}'}{\partial \boldsymbol{\theta}} \boldsymbol{W_j}^{-1} \frac{\partial \boldsymbol{\mu_j}}{\partial \boldsymbol{\theta}}$$

$$= \sum_{j=1}^{k} np_j \boldsymbol{X_j}^T \boldsymbol{G_j} \boldsymbol{D_j}^{-\frac{1}{2}} \boldsymbol{C_\alpha}^{-1} \boldsymbol{D_j}^{-\frac{1}{2}} \boldsymbol{G_j} \boldsymbol{X_j} \tag{3.3}$$

where $\boldsymbol{G_j} = \text{diag}\left\{(g^{-1})'(\eta_{1j}), \ldots, (g^{-1})'(\eta_{pj})\right\}$. Equation (3.3) can be further simplified as,

$$\boldsymbol{M} = \sum_{j=1}^{k} np_j (\boldsymbol{X_j}^*)^T (\boldsymbol{X_j}^*),$$

where $\boldsymbol{X_j}^* = \boldsymbol{R}\boldsymbol{D_j}^{-\frac{1}{2}} \boldsymbol{G_j} \boldsymbol{X_j}$.

3.3.1 Equivalence Theorem when Objective Function is Variance of Parameter Estimates

In this section, we present the equivalence theorem for crossover design when the objective function is a determinant of the variance of parameter estimates. We also present a special case of the theorem when there are only two treatment sequences involved in the design.

Theorem 1 ***(General Equivalence Theorem for Crossover Design when the objective function is $\lvert \boldsymbol{Var}(\hat{\boldsymbol{\theta}}) \rvert$):*** *Consider the design ξ with k treatment sequences as defined in equation* (3.1)*. Then,*

(a) The set of optimal designs is convex.
(b) The design ξ is D-optimal if and only if

$$\textit{trace}\left(\boldsymbol{X_i}^* \boldsymbol{M}(\xi)^{-1} \boldsymbol{X_i}^{*T}\right) \begin{cases} = m \textit{ if } p_i > 0 \\ \leq m \textit{ if } p_i = 0 \end{cases},$$

for each $p_i \in [0, 1]$, where p_i is the allocation corresponding to point ω_i of design ξ for all $i = 1, 2, \ldots, k$, and m is the number of parameters in $\boldsymbol{\theta}$.

Proof of Theorem 3.3.1:

Let k be the number of treatment sequences involved in the experiment and ξ be any design, then $\Phi(\boldsymbol{M}(\xi)) = -\ln det(\boldsymbol{M}(\xi))$.

Proof of (a):

Let ξ_1^* and ξ_2^* be optimal designs i.e.,

$$\Phi[\boldsymbol{M}(\xi_1^*)] = \Phi[\boldsymbol{M}(\xi_2^*)] = \min_\xi \Phi[\boldsymbol{M}(\xi)]$$

and let $\xi^* = (1-\gamma)\xi_1^* + \gamma\xi_2^*$, for $0 \leq \gamma \leq 1$. $\Phi[\boldsymbol{M}(\xi)] = -\ln det(\boldsymbol{M}(\xi))$ is convex on set of positive definite matrices (Boyd & Vandenberghe, 2004). Therefore,

$$\Phi[\boldsymbol{M}(\xi^*)] \leq (1-\gamma)\Phi[\boldsymbol{M}(\xi_1^*)] + \gamma\Phi[\boldsymbol{M}(\xi_2^*)] = \min_\xi \Phi[\boldsymbol{M}(\xi)],$$

which proves the optimality of ξ^*.

Proof of (b):

We have $\boldsymbol{p_r} = (p_1, p_2, \ldots, p_{k-1})'$ and $\boldsymbol{\delta_1^{(r)}} = (1-p_1, -p_2, \ldots, -p_{k-1})'$,
$\boldsymbol{\delta_2^{(r)}} = (-p_1, 1-p_2, \ldots, -p_{k-1})', \ldots, \boldsymbol{\delta_{k-1}^{(r)}} = (-p_1, -p_2, \ldots, 1-p_{k-1})'$.

Hence, $\boldsymbol{p_r} + u\boldsymbol{\delta_1^{(r)}} = (p_1 + u(1-p_1), (1-u)p_2, \ldots, (1-u)p_{k-1})'$,
$\boldsymbol{p_r} + u\boldsymbol{\delta_2^{(r)}} = ((1-u)p_1, p_2 + u(1-p_2), \ldots, (1-u)p_{k-1})', \ldots,$
$\boldsymbol{p_r} + u\boldsymbol{\delta_{k-1}^{(r)}} = ((1-u)p_1, (1-u)p_2, \ldots, p_{k-1} + u(1-p_{k-1}))'$.

Determinant of $(\boldsymbol{\delta_1^{(r)}}, \cdots, \boldsymbol{\delta_{k-1}^{(r)}}) = 1 - (p_1 + p_2 + \cdots + p_{k-1}) = p_k$.

Then for design with k treatment sequences we can write $\boldsymbol{M}$ as,

$$\boldsymbol{M}(\boldsymbol{p_r}) = \sum_{j=1}^{k} np_j(\boldsymbol{X_j}^*)^T(\boldsymbol{X_j}^*) = np_1(\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*) + np_2(\boldsymbol{X_2}^*)^T(\boldsymbol{X_2}^*) + \cdots$$
$$+ np_{k-1}(\boldsymbol{X_{k-1}}^*)^T(\boldsymbol{X_{k-1}}^*) + n\left(1 - (p_1 + p_2 + \cdots + p_{k-1})\right)(\boldsymbol{X_k}^*)^T(\boldsymbol{X_k}^*)$$

For illustration purpose consider the direction $\boldsymbol{\delta_1^{(r)}}$, and calculations for other directions can be done similarly,

$$\Phi(\boldsymbol{p_r} + u\boldsymbol{\delta_1^{(r)}}) = -\ln det\left[\boldsymbol{M}\left(\{p_1 + u(1-p_1), (1-u)p_2, \ldots, (1-u)p_{k-1}\}'\right)\right]$$

$$= -\ln det\Bigg[n\left\{p_1 + u(1-p_1)\right\}(\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)$$
$$+n\left\{(1-u)p_2\right\}(\boldsymbol{X_2}^*)^T(\boldsymbol{X_2}^*)+\cdots+n\left\{(1-u)p_{k-1}\right\}(\boldsymbol{X_{k-1}}^*)^T(\boldsymbol{X_{k-1}}^*)$$
$$+n(1-u)\left\{1-(p_1+p_2+\cdots+p_{k-1})\right\}(\boldsymbol{X_k}^*)^T(\boldsymbol{X_k}^*)\Bigg]$$

$$= -m\ln n - \ln det[\boldsymbol{M}(u,\boldsymbol{p_r})] = -m\ln n + \Phi^{(r)}(u),$$

where $\boldsymbol{M}(u,\boldsymbol{p_r}) = \frac{\boldsymbol{M}(\boldsymbol{p_r}+u\boldsymbol{\delta_1^{(r)}})}{n}$, and $\Phi^{(r)}(u) = -\ln det[\boldsymbol{M}(u,\boldsymbol{p_r})]$.

The directional derivative of the above objective function along one specific direction for a design with k treatment sequences can be calculated as follows:

$$\phi(u,\boldsymbol{p_r}) = \frac{\partial\Phi(\boldsymbol{p_r}+u\boldsymbol{\delta_1^{(r)}})}{\partial u} = \lim_{h\to 0}\frac{1}{h}\left[\Phi^{(r)}(u+h) - \Phi^{(r)}(u)\right]$$

$$= -\lim_{h\to 0}\frac{1}{h}\Bigg\{\ln det\left[\boldsymbol{M}(u+h,\boldsymbol{p_r})\right] - \ln det\left[\boldsymbol{M}(u,\boldsymbol{p_r})\right]\Bigg\}$$

$$= -\lim_{h\to 0}\frac{1}{h}\Bigg\{\ln det\Bigg[\boldsymbol{M}(u,\boldsymbol{p_r})+h(1-p_1)\boldsymbol{X_1}^{*T}\boldsymbol{X_1}^* - hp_2\boldsymbol{X_2}^{*T}\boldsymbol{X_2}^* - \cdots$$
$$-hp_{k-1}\boldsymbol{X_{k-1}}^{*T}\boldsymbol{X_{k-1}}^* - h\left(1-(p_1\cdots+p_{k-1})\right)\boldsymbol{X_k}^{*T}\boldsymbol{X_k}^*\Bigg]det\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\Bigg\}$$

$$= -\lim_{h\to 0}\frac{1}{h}\Bigg\{\ln det\Bigg[\boldsymbol{M}(u,\boldsymbol{p_r})\boldsymbol{M}(u,\boldsymbol{p_r})^{-1} + h\left\{\boldsymbol{X_1}^{*T}\boldsymbol{X_1}^* - \boldsymbol{M}(\boldsymbol{p_r})\right\}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\Bigg]\Bigg\}$$
$$= -\lim_{h\to 0}\frac{1}{h}\Bigg\{\ln det\Bigg[\boldsymbol{I}_p + h\left\{\boldsymbol{X_1}^{*T}\boldsymbol{X_1}^* - \boldsymbol{M}(\boldsymbol{p_r})\right\}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\Bigg]\Bigg\}$$

Using the approximation of determinant $\det(\boldsymbol{I}+h\boldsymbol{A}) = 1 + h\text{trace}(\boldsymbol{A}) + \mathcal{O}(h^2)$ (Bornemann, 2010) we get,

$$= -\lim_{h\to 0}\frac{1}{h}\Bigg\{\ln\left(1 + h\text{trace}\left[\left\{\boldsymbol{X_1}^{*T}\boldsymbol{X_1}^* - \boldsymbol{M}(\boldsymbol{p_r})\right\}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\right] + \mathcal{O}(h^2)\right)\Bigg\}$$

And using $\ln(1+t) = t + \mathcal{O}(t^2)$ we get,

$$= -\lim_{h\to 0}\frac{1}{h}\left\{h\text{trace}\left[(\boldsymbol{X_1}^{*T}\boldsymbol{X_1}^{*}-\boldsymbol{M}(\boldsymbol{p_r}))\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\right]+\mathcal{O}(h^2)\right\}$$

$$= -\text{trace}\left[(\boldsymbol{X_1}^{*T}\boldsymbol{X_1}^{*}-\boldsymbol{M}(\boldsymbol{p_r}))\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\right]$$

$$= \text{trace}\left(\boldsymbol{M}(\boldsymbol{p_r})\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\right)-\text{trace}\left(\boldsymbol{X_1}^{*}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{X_1}^{*T}\right)$$

$$\left.\frac{\partial\Phi(\boldsymbol{p_r}+u\boldsymbol{\delta_1^{(r)}})}{\partial u}\right|_{u=0} = m-\text{trace}\left(\boldsymbol{X_1}^{*}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{X_1}^{*T}\right) \qquad (3.4)$$

The proof follows by equating the above expression in equation (3.4) to zero.

3.3.2 Equivalence Theorem when Objective Function is Variance of Treatment Effect Estimates

As the main interest usually lies in estimating the direct treatment effect contrasts, instead of working with the full variance-covariance matrix of parameters estimate, in this section, we concentrate only on the variance of the estimator of treatment effects $\text{Var}(\hat{\boldsymbol{\tau}})$ given as

$$\text{Var}(\hat{\boldsymbol{\tau}}) \quad = \quad \boldsymbol{H}\text{Var}(\hat{\boldsymbol{\theta}})\boldsymbol{H}', \qquad (3.5)$$

where $\boldsymbol{H}$ is a $(t-1)\times m$ matrix given by $[\mathbf{0}_{(t-1)1}, \mathbf{0}_{(t-1)(p-1)}, \boldsymbol{I}_{t-1}, \mathbf{0}_{(t-1)(t-1)}]$ and $m = p+2t-2$ is the total number of parameters in $\boldsymbol{\theta}$. Below, we present the equivalence theorem for crossover design when the objective function is a determinant of the variance of treatment effects estimate i.e., the determinant of dispersion matrix.

Lemma 1 *Consider function $f : R^n_{>0} \to R_{>0}$, such that $f(\boldsymbol{x}) = \frac{1}{\prod_{i=1}^n x_i}$ where $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)' \in R^n_{>0}$. Then $f(\boldsymbol{x})$ is a strictly convex function.*

Proof of Lemma 1:
Let H be the Hessian matrix, i.e., the matrix of second-order partial derivatives.

Then $H = f(x)(D+qq')$, where D is the diagonal matrix with elements $1/(x_1)^2, \ldots, 1/(x_n)^2$ and q is the column vector with elements $1/(x_1), \ldots, 1/(x_n)$.

The lemma follows as H is positive definite. An alternative proof is provided in the Appendix A.2.

Theorem 2 ***General Equivalence Theorem for Crossover Design when objective function is*** $|\boldsymbol{Var}(\hat{\boldsymbol{\tau}})|$***:*** *Consider the design* ξ *with* k *treatment sequences as defined in equation* (3.1)*. Then,*

(a) The set of optimal designs is convex.
(b) The design ξ *is D-optimal if and only if*

$$trace\left\{\boldsymbol{A}(\boldsymbol{X_i}^*)^T(\boldsymbol{X_i}^*)\right\}\begin{cases} = t-1 \text{ if } p_i > 0 \\ \leq t-1 \text{ if } p_i = 0 \end{cases}$$

for each $p_i \in [0,1]$*, where* $\boldsymbol{A} = \boldsymbol{M}^{-1}\boldsymbol{H}'\left(\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{H}'\right)^{-1}\boldsymbol{H}\boldsymbol{M}^{-1}$*,* p_i *is the allocation corresponding to point* ω_i *of design* ξ *for all* $i = 1, 2, \ldots, k$*, and* t *is number of treatments.*

Proof of Theorem 3.3.2:

Let k be the number of treatment sequences involved in the experiment and ξ be any design, then $\Phi(\boldsymbol{M}(\xi)) = \ln det(\boldsymbol{H}\boldsymbol{M}(\xi)^{-1}\boldsymbol{H}')$.

Proof of (a):

Let ξ_1^* and ξ_2^* be optimal designs i.e.,

$$\Phi[\boldsymbol{M}(\xi_1^*)] = \Phi[\boldsymbol{M}(\xi_2^*)] = \min_\xi \Phi[\boldsymbol{M}(\xi)]$$

and let $\xi^* = (1-\gamma)\xi_1^* + \gamma\xi_2^*$, for $0 \leq \gamma \leq 1$.

Since we are using the D-optimality criterion, we prove the following equation (3.6) to prove the optimality of ξ^*.

$$|\boldsymbol{H}\boldsymbol{M}(\xi^*)^{-1}\boldsymbol{H}'| \leq (1-\gamma)|\boldsymbol{H}\boldsymbol{M}(\xi_1^*)^{-1}\boldsymbol{H}'| + \gamma|\boldsymbol{H}\boldsymbol{M}(\xi_2^*)^{-1}\boldsymbol{H}'|. \quad (3.6)$$

Since both $\boldsymbol{M}(\xi_1^*)$ and $\boldsymbol{M}(\xi_2^*)$ are positive definite, we can find a non-singular matrix $\boldsymbol{O}^{-1}$ such that $\boldsymbol{M}(\xi_1^*) = \boldsymbol{O}\boldsymbol{O}^T$ and $\boldsymbol{M}(\xi_2^*) = \boldsymbol{O}\boldsymbol{\Lambda}\boldsymbol{O}^T$, where $\boldsymbol{\Lambda} = \mathrm{diag}\{\lambda_1, \ldots, \lambda_m\}$ is a $m \times m$ diagonal matrix (see page 41 Rao, 1973). In this situation, $\boldsymbol{M}(\xi^*) = \boldsymbol{O}((1-\gamma)\boldsymbol{I} + \gamma\boldsymbol{\Lambda})\boldsymbol{O}^T$. Then (3.6) is equivalent to

$$|\boldsymbol{G}((1-\gamma)\boldsymbol{I} + \gamma\boldsymbol{\Lambda})^{-1}\boldsymbol{G}^T| \leq (1-\gamma)|\boldsymbol{G}\boldsymbol{G}^T| + \gamma|\boldsymbol{G}\boldsymbol{\Lambda}^{-1}\boldsymbol{G}^T|, \quad (3.7)$$

where $\boldsymbol{G} = \boldsymbol{H}(\boldsymbol{O}^T)^{-1}$. According to Theorem 1.1.2 in (Fedorov, 1972),

$$|\boldsymbol{G}((1-\gamma)\boldsymbol{I}+\gamma\boldsymbol{\Lambda})^{-1}\boldsymbol{G}^T| = \sum_{1\leq i_1<\cdots<i_q\leq m} |\boldsymbol{G}^T[i_1, \ldots, i_q]|^2 \prod_{l=1}^{q} \frac{1}{(1-\gamma)+\gamma\lambda_{i_l}},$$

where $\boldsymbol{G}^T[i_1,\ldots,i_q]$ is the $q\times q$ sub-matrix of $\boldsymbol{G}^T$ consisting of the $i_1,\ldots,i_q$ rows of $\boldsymbol{G}^T$. Similarly,

$$(1-\gamma)|\boldsymbol{G}\boldsymbol{G}^T|+\gamma|\boldsymbol{G}\boldsymbol{\Lambda}^{-1}\boldsymbol{G}^T| = \sum_{1\le i_1<\cdots<i_q\le m}|\boldsymbol{G}^T[i_1,\ldots,i_q]|^2\left(1-\gamma+\gamma\prod_{l=1}^{q}\frac{1}{\lambda_{i_l}}\right).$$

Then (3.7) is true if

$$\prod_{l=1}^{q}\frac{1}{(1-\gamma)+\gamma\lambda_{i_l}} \le 1-\gamma+\gamma\prod_{l=1}^{q}\frac{1}{\lambda_{i_l}}. \tag{3.8}$$

Since $f(\boldsymbol{x})=\frac{1}{\prod_{i=1}^{q}x_i}$ is convex function (from Lemma 1), we have $f\left((1-\gamma)\mathbf{1}+\gamma\boldsymbol{\lambda}\right)\le (1-\gamma)f(\mathbf{1})+\gamma f(\boldsymbol{\lambda})$, where $\boldsymbol{\lambda}=(\lambda_{i_1},\cdots,\lambda_{i_q})$ and hence the result follows.

Proof of (b):

$$\begin{aligned}\boldsymbol{M}(\boldsymbol{p_r}) &= np_1(\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)+np_2(\boldsymbol{X_2}^*)^T(\boldsymbol{X_2}^*)+\cdots+np_{k-1}(\boldsymbol{X_{k-1}}^*)^T(\boldsymbol{X_{k-1}}^*)\\ &\quad+n\left(1-(p_1+p_2\cdots+p_{k-1})\right)(\boldsymbol{X_k}^*)^T(\boldsymbol{X_{k-1}}^*).\end{aligned}$$

$$\Phi(\boldsymbol{p_r}+u\boldsymbol{\delta_1^{(r)}}) = \Phi\left(\{p_1+u(1-p_1),(1-u)p_2,\ldots,(1-u)p_{k-1}\}'\right)$$

$$= \ln det\left[\boldsymbol{H}\left\{\boldsymbol{M}\left(\{p_1+u(1-p_1),(1-u)p_2,\ldots,(1-u)p_{k-1}\}'\right)\right\}^{-1}\boldsymbol{H}'\right]$$

$$\begin{aligned}&= -(t-1)\ln n+\ln det\Bigg[\boldsymbol{H}\Bigg\{\{p_1+u(1-p_1)\}(\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\\ &+\{(1-u)p_2\}(\boldsymbol{X_2}^*)^T(\boldsymbol{X_2}^*)+\cdots+\{(1-u)p_{k-1}\}(\boldsymbol{X_{k-1}}^*)^T(\boldsymbol{X_{k-1}}^*)\\ &+(1-u)\{1-(p_1+p_2+\cdots+p_{k-1})\}(\boldsymbol{X_k}^*)^T(\boldsymbol{X_k}^*)\Bigg\}^{-1}\boldsymbol{H}'\Bigg]\end{aligned}$$

$$= -(t-1)\ln n+\ln det\left[\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right] = -(t-1)\ln n+\Phi^{(r)}(u),$$

where now $\Phi^{(r)}(u)=\ln det\left[\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]$.

Consider direction $\boldsymbol{\delta_1^{(r)}}$, then the directional derivative of the above objective function for a design with k treatment sequences can be calculated as follows:

$$\phi(u,\boldsymbol{p_r}) \;=\frac{\partial\Phi(\boldsymbol{p_r}+u\boldsymbol{\delta_1^{(r)}})}{\partial u}=\lim_{h\to0}\frac{1}{h}\left[\Phi^{(r)}(u+h)-\Phi^{(r)}(u)\right]$$

$$=\lim_{h\to0}\frac{1}{h}\left\{\ln det\left[\boldsymbol{H}\boldsymbol{M}(u+h,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]-\ln det\left[\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]\right\}$$

$$= \lim_{h\to 0} \frac{1}{h}\Bigg\{ \ln det\left[\boldsymbol{H}\left\{(1-\mu-h)\boldsymbol{M}(\boldsymbol{p_r}) + (\mu+h)(\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\right\}^{-1}\boldsymbol{H}'\right] - \ln det\left[\boldsymbol{H}\boldsymbol{M}(\mu,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]\Bigg\}$$

$$= \lim_{h\to 0} \frac{1}{h}\Bigg\{ \ln det\left[\boldsymbol{H}\left\{\boldsymbol{M}(u,\boldsymbol{p_r}) - h\left(\boldsymbol{M}(\boldsymbol{p_r}) - (\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\right)\right\}^{-1}\boldsymbol{H}'\right] - \ln det\left[\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]\Bigg\}$$

$$= \lim_{h\to 0} \frac{1}{h}\Bigg\{ \ln det\left[\boldsymbol{H}\left\{[\boldsymbol{M}(u,\boldsymbol{p_r})]\left[\boldsymbol{I} - h\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\left(\boldsymbol{M}(\boldsymbol{p_r}) - (\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\right)\right]\right\}^{-1}\boldsymbol{H}'\right] \times det\left[\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]^{-1}\Bigg\}$$

$$= \lim_{h\to 0} \frac{1}{h}\Bigg\{ \ln det\left[\boldsymbol{H}\left\{\left[\boldsymbol{I} - h\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\left(\boldsymbol{M}(\boldsymbol{p_r}) - (\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\right)\right]^{-1}[\boldsymbol{M}(u,\boldsymbol{p_r})]^{-1}\right\}\boldsymbol{H}'\right] \times det\left[\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]^{-1}\Bigg\}$$

Assuming h is sufficiently small we use the binomial series expansion $(\boldsymbol{I} + h\boldsymbol{X})^{-1} = \sum_{i=0}^{\infty}(-t\boldsymbol{X})^i$ to obtain,

$$\phi(u,\boldsymbol{p_r}) = \lim_{h\to 0}\frac{1}{h}\left\{\ln det\left[\boldsymbol{I} + h\boldsymbol{B} + \mathcal{O}(h^2)\right]\right\},$$

$$\boldsymbol{B} = \boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\left[\boldsymbol{M}(\boldsymbol{p_r}) - (\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\right]\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\left[\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]^{-1}.$$

Using $\ln det\left[\boldsymbol{I} + h\boldsymbol{B} + \mathcal{O}(h^2)\right] = h\text{trace}(\boldsymbol{B}) + \mathcal{O}(h^2)$ (Withers & Nadarajah, 2010),

$$\phi(u,\boldsymbol{p_r}) = \text{trace}\Bigg\{\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\left[\boldsymbol{M}(\boldsymbol{p_r}) - (\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\right]\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}' \times \left[\boldsymbol{H}\boldsymbol{M}(u,\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]^{-1}\Bigg\}$$

$$\phi(u,\boldsymbol{p_r})|_{u=0} = \text{trace}\Bigg\{\boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\left[\boldsymbol{M}(\boldsymbol{p_r}) - (\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\right]\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}' \times \left[\boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]^{-1}\Bigg\}$$

$$= \text{trace}\Big\{ \boldsymbol{I}_{(t-1)} - \boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}(\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}' \times \left[\boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right]^{-1} \Big\}$$

$$= (t-1) - \text{trace}\Big\{ \boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}(\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*)\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}' \left(\boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right)^{-1} \Big\}$$

$$= (t-1) - \text{trace}\Big\{ \left[\boldsymbol{M}^{-1}\boldsymbol{H}' \left(\boldsymbol{H}\boldsymbol{M}^{-1}\boldsymbol{H}'\right)^{-1} \boldsymbol{H}\boldsymbol{M}^{-1}\right] (\boldsymbol{X_1}^*)^T(\boldsymbol{X_1}^*) \Big\} \quad (3.9)$$

The proof follows by equating the above expression in equation (3.9) to zero.

3.3.3 Illustration

To illustrate the results of the above general equivalence theorems, we consider a design space $\{AB, BA\}$ has $k = 2, p = 2$. Since we are considering a local optimality approach, for illustration purposes we assume that the parameter values are $\boldsymbol{\theta} = (\lambda, \beta_2, \tau_B, \rho_B)' = (0.5, -1.0, 4.0, -2.0)'$. Note that we need to assume parameter values before calculating the optimal proportions. Considering the AR(1) correlation structure with $\alpha = 0.1$, i.e.,

$$\boldsymbol{C_\alpha} = \left(\alpha^{|i-i'|}\right) = \begin{pmatrix} 1 & \alpha \\ \alpha & 1 \end{pmatrix},$$

for the assumed parameter values the optimal proportions are $p_1 = p_2 = 0.5$.

The graph of the objective function, $\Phi(p_1) = -\ln det(\boldsymbol{M}(p_1))$ and its directional derivative trace $\left(\boldsymbol{X_1}^*\boldsymbol{M}(p_1)^{-1}\boldsymbol{X_1}^{*T}\right) - m$ w.r.t $p_1 \in [0, 1]$ are shown in Figure 3.1.

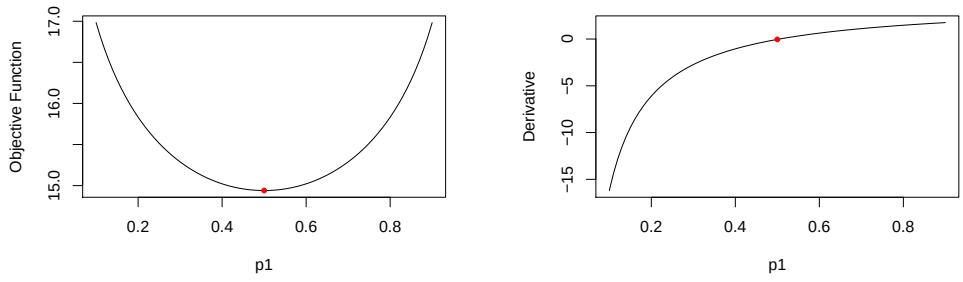


Figure 3.1: Objective function and its directional derivative for designs with two treatment sequences.

Graphs in Figure 3.1 verify that the minimum of the objective function is located at $p_1 = 0.5$ and directional derivative is zero at $p_1 = 0.5$. Using Theorem 1, we conclude that for assumed values of parameters, design

$$\xi = \left\{ \begin{array}{cc} AB & BA \\ 0.5 & 0.5 \end{array} \right\}$$

is the D-optimal design when the objective function is $Var(\hat{\boldsymbol{\theta}})$.

Considering $Var(\hat{\boldsymbol{\tau}})$ as the objective function, the graph of the objective function, $\Phi(p_1) = \ln det[\boldsymbol{H}\boldsymbol{M}(p_1)^{-1}\boldsymbol{H}^{'}]$ and it's directional derivative w.r.t $p_1 \in [0, 1]$ are shown in Figure 3.2.

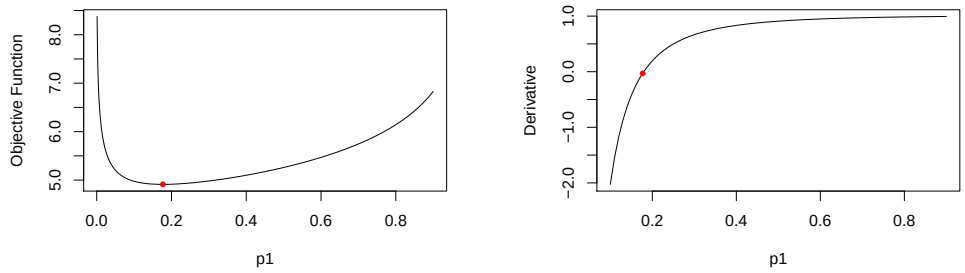


Figure 3.2: Objective function and its directional derivative for designs with two treatment sequences.

Graphs in Figure 3.2 verify that the minimum of the objective function is located at $p_1 = 0.177$ and directional derivative is zero at $p_1 = 0.177$. Using Theorem 2, we conclude that for assumed values of parameters, design

$$\xi = \left\{ \begin{array}{cc} AB & BA \\ 0.177 & 0.823 \end{array} \right\}$$

is the D-optimal design when the objective function is $Var(\hat{\boldsymbol{\tau}})$.

3.4 Real Example

Consider the first example discussed in Chapter 2, where data is obtained from the work environment experiment conducted at Booking.com (Pitchforth et al., 2020). In recent years, many corporate offices and organizations have adopted open office spaces over traditional cubical office spaces. Since there were no previous studies to examine the effects of office designs in workspaces, Booking.com conducted an experiment to assess different office spacing efficiency.

3.4.1 Work Environment Experiment

For illustration purposes, consider the response $commit\ count$ to illustrate the optimal crossover design for the Poisson response. The commit count is the number of commits submitted to the main git repository by each subject. In the fitted model, we examine three primary predictors: $area$, $wave$, and $carryover$. Here, $area$ represents the direct treatment effect, $wave$ denotes the period effect, and $carryover$ represents the effect of the treatment from the previous period. To illustrate the local optimality approach, we assume specific parameter values $\boldsymbol{\theta} = (2.0, 0.3, 0.8, -0.1, -2.0, 0.40, -2.0, -1.0, 0.3, -1.0)'$, which lead to non-uniform allocations using the log link function and AR(1) correlation structure with $\alpha = 0.1$.

According to Theorem 3.3.1, the D-optimal design, i.e., the optimal proportions, can be obtained by solving the following system of equations instead of performing constrained optimization:

$$\text{trace}\left(\boldsymbol{X_i}^{*}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{X_i}^{*T}\right) = 10,$$

for $i = 1, 2, 3, 4$. The resulting D-optimal design is the same as the one obtained through constrained optimization, indicating that the design is given by:

$$\xi = \left\{ \begin{array}{cccc} BADC & CDAB & DBCA & ACBD \\ 0.2375 & 0.2894 & 0.2246 & 0.2485 \end{array} \right\}$$

is the D-optimal design when the objective function is $Var(\hat{\boldsymbol{\theta}})$.

Similarly, according to Theorem 3.3.2, for the objective function $Var(\hat{\boldsymbol{\tau}})$, the D-optimal design can be obtained by solving the following system of equations:

$$\text{trace}\left\{\left[\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}'\left(\boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right)^{-1}\boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\right](\boldsymbol{X_i}^{*})^{T}(\boldsymbol{X_i}^{*})\right\} = 3,$$

for $i = 1, 2, 3, 4$. Again, the resulting D-optimal design is the same as the one obtained through constrained optimization, indicating that the design is given by:

$$\xi = \left\{ \begin{array}{cccc} BADC & CDAB & DBCA & ACBD \\ 0.2900 & 0.2963 & 0.1734 & 0.2403 \end{array} \right\}$$

is the D-optimal design.

Remark: In Chapter 1, we study the effect of misspecification of working correlation structures on optimal design. We calculate optimal designs under two

choices of unknown parameters for a misspecified working correlation structure. Then we calculate relative D-efficiency under two parameter choices. The relative D-efficiency under two parameter choices suggests that the effect of variance misspecification on the local optimal designs is minimal. We also study the performance of proposed locally optimal designs via a sensitivity study in terms of the relative loss of efficiency for choosing assumed parameter values instead of true parameter values. The relative loss of efficiency increases as we move away from true parameter values. However, Figure 1.9 in Chapter 1 suggest that this loss of efficiency does not go beyond 2%. We also calculate the optimal designs with all 24 sequences, by considering AR (1) correlation structure and different values of α. We observe that in the case of non-uniform allocations, the optimal design has more zeros than non-zero proportions; and these allocations do not vary a lot as α changes, particularly for the sequences where we have zero allocations.

3.4.2 Dairy Dietary Experiment

In the introduction section of this chapter, we provided a brief overview of a dairy dietary experiment that aimed to investigate the impact of various dietary starch levels on milk production in cows. The experiment followed a four-period four-treatment trial design, as first proposed by (Kenward & Jones, 1992). To administer the order in which diets were received by cows, a Latin square design with four treatment sequences was employed.

In this specific example, the researchers obtained binary responses from a four-period crossover trial. They allocated four treatment sequences to a group of eighty different subjects across the four periods. At the end of each period, the efficacy measurement of each subject was recorded as success or failure, depending on whether a diet was effective or not, resulting in a joint outcome at the end of the trial. The dataset contained the four pre-determined treatment sequences $\Omega = ABCD, BDAC, CADB, DCBA$ along with the joint outcomes of the four different periods for each subject following a specific treatment sequence. The Latin square design used in the above experiment is an example of $k = 4, p = 4$. To illustrate the local optimality approach, we assume specific parameter values $\boldsymbol{\theta} = (-2, 0.25, 0, 0.75, 1, 5, -1.5, -3.5, 2.75, 0.75)'$, which lead to non-uniform allocations with the logit link function and AR(1) correlation structure with $\alpha = 0.1$.

According to Theorem 3.3.1, the D-optimal design, i.e., optimal proportions, can be obtained by solving the following system of equations instead of performing constrained optimization:

$$\text{trace}\left(\boldsymbol{X_i}^{*}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{X_i}^{*T}\right) = 10,$$

for $i = 1, 2, 3, 4$. The resulting D-optimal design is the same as the one obtained through constrained optimization, indicating that the design given by:

$$\xi = \left\{ \begin{array}{cccc} ABCD & BDAC & CADB & DCBA \\ 0.3540 & 0.2108 & 0.2726 & 0.1626 \end{array} \right\}$$

is the D-optimal design when objective function is $Var(\hat{\boldsymbol{\theta}})$.

Similarly, according to Theorem 3.3.2, for the objective function $Var(\hat{\boldsymbol{\tau}})$, the D-optimal design can be obtained by solving the following system of equations:

$$\text{trace}\left\{\left[\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}'\left(\boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\boldsymbol{H}'\right)^{-1}\boldsymbol{H}\boldsymbol{M}(\boldsymbol{p_r})^{-1}\right](\boldsymbol{X_i}^{*})^{T}(\boldsymbol{X_i}^{*})\right\} = 3,$$

for $i = 1, 2, 3, 4$. Again, the obtained D-optimal design is the same as the one obtained through constrained optimization, indicating that the design given by

$$\xi = \left\{ \begin{array}{cccc} ABCD & BDAC & CADB & DCBA \\ 0.1725 & 0.2482 & 0.2225 & 0.3586 \end{array} \right\}$$

is the D-optimal design.

3.5 Discussion

In many experiments in real life, uniform designs are typically used. Uniform designs are optimal in the case of a linear model i.e., when the response is normally distributed. However, in situations where responses are non-normal, the obtained optimal proportions are not necessarily uniform. In this thesis, we derive an expression for the general equivalence theorem to check for the optimality of identified locally D-optimal crossover designs for GLMs. The equivalence theorem provides us with a system of equations that can calculate optimal proportions without performing constrained optimization of the objective function. We derive two different versions of the general equivalence theorem, one with the objective function $Var(\hat{\boldsymbol{\theta}})$ and the other with the objective function $Var(\hat{\boldsymbol{\tau}})$. We illustrate the application of these equivalence theorems on two real-life examples and obtain the same set of optimal proportions by solving the system of equations as obtained by performing constrained optimization.

Chapter 4

Overview: Computer Experiments

4.1 Surrogates

A surrogate acts as a stand-in for something real. In statistics, predictions made by a fitted model can serve as a substitute for the actual process that generates the data. If the model is flexible, properly regularized, trained on sufficient data, and fitted reliably, this surrogate can be highly useful. Since collecting real data can be expensive or even impossible in some cases, due to cost, feasibility, or ethical issues, a surrogate offers a more affordable and practical way to study patterns and explore hypothetical scenarios. Surrogate models differ from traditional statistical models in their main purpose. While typical models are often used for interpretation, identifying causal relationships, or estimating parameters, surrogates are more focused on accurately and practically replicating system behavior. Still, this comparison simplifies a more complex distinction.

The concept of surrogate modeling originated in fields like physics, applied mathematics, and engineering, where mathematical models using numerical solvers have long been standard practice. As these models grew in complexity and became more resource-intensive to run, practitioners began turning to meta-models based on a limited number of simulations. These efforts often involved collaboration with statisticians or used approaches similar to those found in statistics. The data generated from expensive computer simulations were used to fit flexible models that could stand in for the simulations themselves. The reasons for using these models varied, from cutting costs or reducing computational burden to dealing with limitations like expired software licenses or unavailable computing resources. These fitted meta-models came to be known as surrogates or emulators, terms that are frequently used interchangeably. Es-

sentially, a surrogate is meant to replicate the behavior of the original numerical solver. The entire process of designing, running, and fitting such models is referred to as a computer experiment.

A computer experiment is similar to a traditional statistical experiment, with the key difference being that the data come from computer simulations instead of real-world measurements, field studies, or surveys. Surrogate modeling involves applying statistical methods to the results of these computer experiments. Since simulations are often less costly than collecting physical data, they can serve as a practical alternative or a preliminary step. Even though simulations can sometimes be just as expensive, they're often preferred because the computational setup is more controlled and better understood. For instance, many numerical solvers are deterministic, while real-world data collection typically involves noise and measurement errors. Historically, the presence or absence of noise created a divide between the design and modeling of surrogates and broader statistical approaches.

The gap between surrogate modeling and traditional statistical methods is steadily closing, not just because experimentation is evolving (which it is), but largely due to progress in machine learning. A key example is the Gaussian Process (GP) regression model, originally adapted from the *kriging* method in geostatistics from the 1960s. GPs have become a standard surrogate model, particularly valuable in settings where prediction is the main goal. Machine learning researchers have shown that GPs are highly effective across a range of tasks, including regression, classification, active learning, reinforcement learning, optimization, and latent variable modeling. They've also created accessible software tools and libraries, making these methods easier for non-experts, especially in tech, to adopt. We will discuss GP surrogates in more detail in Chapter 6. For example, retail brands like Amazon and Nike use surrogate models to personalize their platform and increase user engagement, while Uber applies surrogates trained on traffic simulations to dynamically route shared rides, improving efficiency and reducing wait times.

Around the same time, computer simulation began to gain momentum as a serious tool for scientific investigation. Researchers in fields like mathematical biology and economics had pushed traditional, closed-form, equilibrium-based models as far as they could go. Like physicists and engineers before them, they turned to simulation to explore more complex, dynamic systems. However, their simulations were often different in nature. Instead of relying on deterministic solvers like finite element methods or Navier–Stokes equations, these researchers developed stochastic simulations and agent-based models to study things like predator-prey interactions, disease transmission, and resource man-

agement in areas like healthcare and economics. This shift was fueled by massive growth in computing power, better software tools, and improved STEM education at earlier stages. Together, these changes led to a revival in simulation-based science. While we're still learning how to best model these complex simulation experiments, one thing is clear: the line between a surrogate model and a traditional statistical model has nearly disappeared.

When working with real-world data, like from a past epidemic, further experimentation is often limited to simulations and mathematical modeling. You can't ethically or practically infect a community with something like Ebola just to observe the outcome. Instead, simulations that model how virtual agents spread disease are run, and surrogates are built from these costly, complex runs. These surrogates can then be calibrated using limited real data. Getting meaningful insights from this process relies heavily on good surrogate modeling and experimental design. Traditional statistical methods aren't much help here. Concepts like population or causality are less relevant, causal relationships are built directly into the simulation itself. What matters more is whether the surrogate provides reliable, flexible predictions. That involves more than just replicating the simulated dynamics; it means creating models that can be fit with minimal intervention, yet still offer robust diagnostics, sensitivity analysis, and tools for optimization and refinement, either automatically or with expert guidance. All of this must also be computationally efficient; a surrogate model that's more expensive than the original simulation defeats the purpose. Efficiency and practicality are key to effective meta-modeling.

4.2 Review: Space-Filling and Orthogonal Designs

Many industries are shifting toward using computer simulations to model products and processes instead of relying entirely on physical experimentation. While this approach offers efficiency and cost savings, it raises a legitimate concern: if the computer model does not accurately mirror the real-world system, the outcomes it produces could be significantly off-target. Despite this limitation, calibrating a computer model with a small number of physical experiments is often far more practical than building a statistical model purely from experimental data. When appropriately calibrated, a computer model can offer superior predictive performance, as it integrates the physical principles governing the system (Joseph & Melkote, 2009; Kennedy & O'Hagan, 2001). Such models

are especially useful for guiding early-stage prototyping and design, enabling quicker optimization of the physical system with fewer experimental trials.

To improve the realism and accuracy of computer models, the underlying mathematical representations are becoming increasingly complex. These models often involve systems of partial differential equations that require advanced numerical methods, such as finite element analysis, for their solution. As a result, running such simulations can be computationally expensive and time-intensive, making it challenging to explore and optimize the model efficiently. This is where experimental design and statistical modeling techniques can play a critical role, helping to streamline the process and reduce computational burden.

4.2.1 Space Filling Designs

The deterministic nature of computer experiments marks a significant departure from traditional experimental design practices used in physical experiments. For instance, replication, randomization, and blocking, cornerstones of physical experimentation, are generally unnecessary in computer experiments (see (C. F. J. Wu, 2015)). Additionally, the absence of random error allows for exploration across a much broader experimental region than would typically be feasible in physical settings. Another advantage is that all input factors can be easily varied, even those with many levels, which is often impractical in physical experiments. These distinctions, however, introduce unique challenges for experimental design and analysis in computer experiments (Sacks et al., 1989). Standard linear regression models, commonly applied in physical experiments, are less appealing here because residuals, used to assess model fit, are irrelevant in a deterministic setting. Furthermore, the expansive experimental region increases the likelihood of complex, nonlinear behavior, rendering simple polynomial models insufficient. As a result, interpolation-based methods like kriging are preferred, as they can effectively capture intricate nonlinear surfaces (Santner et al., 2003). To support such modeling, experimental designs must be carefully constructed to extract the most information possible about the underlying response surface.

In physical experiments, the design process typically begins by selecting a few discrete levels for each factor and arranging them using a factorial structure to accommodate a fixed number of experimental runs (C. J. Wu & Hamada, 2011). However, in computer experiments, changing the levels of input factors incurs no additional cost, which opens the door to a completely different design strategy. Rather than focusing on a limited set of levels, the goal shifts to intel-

ligently distributing design points throughout the entire experimental region. This gives rise to the concept of space-filling designs. Simply put, a space-filling design aims to spread points evenly across the design space, minimizing gaps and ensuring that no region is left unexplored.

A natural and intuitive strategy for designing computer experiments is to place the design points so that they *thoroughly cover* the experimental region. That is, for any location within the region, there should be a nearby design point. To formalize this idea, suppose there are p factors and let $\mathcal{X}$ represent the experimental region. In most cases, this region can be rescaled to the unit hypercube, $\mathcal{X} = [0, 1]^p$, which simplifies both the design and analysis of the experiment. Let $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ denote a set of design points, where each $\mathbf{x}_i$ lies within the unit hypercube $[0, 1]^p$. For any arbitrary point $\mathbf{x}$ in this space, its distance to the nearest design point is defined as $\min_i d(\mathbf{x}, \mathbf{x}_i)$, where the distance function is given by

$$d(\mathbf{u}, \mathbf{v}) = \left(\sum_{j=1}^{p} |u_j - v_j|^s \right)^{1/s}.$$

The parameter s determines the metric: $s = 1$ gives the Manhattan (rectangular) distance, and $s = 2$ gives the Euclidean distance. The point in the experimental space that is farthest from all design points represents the worst-case location in terms of coverage. The corresponding distance to the nearest design point is expressed as $\max_{\mathbf{x} \in [0,1]^p} \min_i d(\mathbf{x}, \mathbf{x}_i)$. To ensure a well-distributed design, we aim to minimize this worst-case distance. This leads to the following optimization problem:

$$\min_{\mathcal{D}} \max_{\mathbf{x} \in \mathcal{X}} \min_i d(\mathbf{x}, \mathbf{x}_i) \tag{4.1}$$

which defines the *minimax distance design* (Johnson et al., 1990) a widely used criterion for constructing space-filling designs.

Since computer experiments are deterministic and contain no measurement error, the information obtained from two nearby points is often redundant. This insight motivates an alternative strategy for achieving space-filling designs: placing design points as far apart from each other as possible. To formalize this idea, we compute the minimum distance among all pairs of design points $\min_{i \neq j} d(\mathbf{x}_i, \mathbf{x}_j)$. A space-filling design can then be obtained by maximizing this minimum distance:

$$\max_{\mathcal{D}} \min_{i \neq j} d(\mathbf{x}_i, \mathbf{x}_j). \tag{4.2}$$

This approach is known as the *maximin distance design* (Johnson et al., 1990). Compared to minimax distance designs, maximin designs are computationally simpler to construct, as they require only the pairwise distances among the design points rather than distances across the entire experimental region.

One limitation of minimax and maximin distance designs is that they tend to perform poorly when projected onto lower-dimensional subspaces. For instance, projecting a minimax design with seven points onto the x_1 axis might result in only three distinct values, while the same projection of a maximin design might yield just four. This is problematic when a factor like x_2 has little to no effect on the response, as the extra runs corresponding to repeated x_1 values offer no additional information. This inefficiency is especially concerning in light of the *effect sparsity* principle, which suggests that in most systems, only a few of the many input factors are truly influential. As a result, it is advantageous in computer experiments to use designs that avoid replicating values in lower-dimensional projections. One effective solution to this problem is the *Latin hypercube design* (McKay et al., 1979), which ensures that the projections along each factor span the experimental region more uniformly.

To construct a Latin hypercube design (LHD) with n runs, the range of each input factor is divided into n equally spaced intervals. A single value, often the midpoint, is sampled from each interval, ensuring that each factor level appears exactly once in the design. As a result, when projected onto any single factor, the design consists of n unique levels. This makes LHDs particularly suitable when only a few factors significantly influence the response. However, not all LHDs are equally effective. For example, simply placing points along the diagonal of the grid yields a valid LHD, but one that lacks desirable space-filling properties. To address this, (Morris & Mitchell, 1995a) proposed the *maximin Latin hypercube design* (MmLHD), which maximizes the minimum distance between any pair of design points. They introduced the following criterion to evaluate and search for high-quality MmLHDs:

$$\min_{\mathcal{D}} \left\{ \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{1}{d^k(\mathbf{x}_i, \mathbf{x}_j)} \right\}^{1/k}, \tag{4.3}$$

where $d(\mathbf{x}_i, \mathbf{x}_j)$ is the distance between design points and k is a large positive constant. Minimizing this objective encourages large pairwise distances and improves the overall space-filling nature of the design. It is easy to see that as $k \to \infty$ this criterion becomes the maximin distance criterion in 4.2.

(Morris & Mitchell, 1995a) recommended selecting the smallest value of k that still yields a maximin Latin hypercube design. This strategy helps reduce

the number of point pairs with the minimum distance, a count referred to as the index of the design. Designs with a lower index are typically more desirable due to better spacing among points. Beyond MmLHDs, several alternative space-filling LHDs have been proposed using different optimality criteria. These include orthogonal array–based LHDs (Owen, 1994; Tang, 1993), orthogonal LHDs (Ye, 1998), uniform LHDs (Jin et al., 2005), orthogonal-maximin LHDs (Joseph & Hung, 2008), and generalized LHDs (Dette & Pepelyshev, 2010), among others. Among these, the MmLHD remains one of the most widely used designs in practice due to its simplicity and ease of implementation in software.

In addition to Latin hypercube–based designs, other types of designs have been developed for computer experiments. These include uniform designs (Fang, 1980), maximum entropy designs (Shewry & Wynn, 1987), integrated mean squared error designs (Sacks et al., 1989), nested space-filling designs (Qian et al., 2009), sliced space-filling designs (Qian & Wu, 2009), multilayer designs (Ba & Joseph, 2011), minimum energy designs (Joseph, Dasgupta, et al., 2015), and bridge designs (Jones et al., 2015). Further information can be found in the comprehensive texts by (Santner et al., 2003) and (Fang et al., 2006). In the next section, we introduce a design known as the *maximum projection* (MaxPro) design, developed by (Joseph, Gul, & Ba, 2015). We use the MaxPro criterion to obtain a space-filling design with our algorithm.

4.2.2 Maximum Projection Design

(Joseph, Gul, & Ba, 2015) introduced the *maximum projection design* (MaxPro) to ensure desirable projection properties across all subspaces of the factors. The method is based on a weighted distance function:

$$d(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}) = \left(\sum_{l=1}^{p} \theta_l |x_{il} - x_{jl}|^s \right)^{1/s},$$

where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)$, with $\sum_{l=1}^{p} \theta_l = 1$ and $\theta_l \geq 0$. The weights θ_l represent the relative importance of the l^{th} factor. To focus on a sub-dimensional space, one sets $\theta_l = 1$ for the relevant factors and 0 otherwise.

To achieve good projections in all subspaces, a prior distribution $p(\boldsymbol{\theta})$ is assigned to the weight vector, and the reciprocal distance criterion is averaged

over this distribution:

$$\min_{D} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \int \frac{1}{d^k(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta})} \, p(\boldsymbol{\theta}) \, d\boldsymbol{\theta}. \tag{4.4}$$

While this formulation is generally difficult to evaluate and optimize, it simplifies considerably under a uniform prior on $\boldsymbol{\theta}$ and with $k = sp$, yielding the criterion:

$$\min_{D} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{1}{\prod_{l=1}^{p} |x_{il} - x_{jl}|^s}. \tag{4.5}$$

Typically, the exponent s is set to 2. This simplified criterion is computationally convenient and effective, as it enforces the Latin hypercube structure by penalizing repeated coordinates through the product term.

The resulting designs exhibit good space-filling properties along with favorable projections. A key difference between these designs and MmLHDs is that the factor levels in MaxPro designs are not equally spaced; instead, they tend to cluster toward the boundaries of the design space. Whether this is advantageous depends on the goal of the computer experiment. For instance, if the aim is prediction, which is the common case, then concentrating points near the boundaries is beneficial (Dette & Pepelyshev, 2010). On the other hand, if the purpose is numerical integration through sample averages, then uniform distribution of the design points is preferable. As such, MaxPro designs are well-suited for most computer experiments, especially those focused on prediction. When integration is the primary objective, the uniformity of MaxPro designs can be enhanced by imposing equally spaced level constraints, called MaxProLHD.

4.2.3 Orthogonal Design

Latin Hypercube Sampling (LHS) possesses several strengths:

(a) it is computationally efficient to construct;

(b) it scales well with a large number of design points and input dimensions; and

(c) it typically results in a sample mean with lower variance compared to simple random sampling.

Nevertheless, LHS does not attain the theoretically minimal variance for the sample mean. To enhance its performance, various researchers have sought

modifications aimed at further reducing variance. Notably, (Owen, 1994) and (Tang, 1993) independently explored the use of randomized orthogonal arrays to achieve this goal.

An orthogonal array (OA) of strength t, with n runs and s factors, is denoted by OA(n, s, q, r). It can be viewed as a fractional factorial design (FFD) in which every projection onto any $m \leq r$ factors yields a complete factorial design. Orthogonal arrays of strength two are widely used for experiment planning across numerous domains and are frequently represented in the form of orthogonal design tables.

(Tang, 1993) proposed to construct orthogonal array-based LHDs (OALHDs) from existing orthogonal arrays (OAs). The key idea of this construction is to deterministically replace OA entries with a random permutation of LHD elements. OALHDs inherit the properties of OAs and tend to have better space-filling properties compared to random LHDs. Note that the design sizes of OALHDs rely on the existence of corresponding OAs. Search algorithms should be used to generate optimal LHDs when no construction methods are available. (Morris & Mitchell, 1995a) proposed a simulated annealing (SA) algorithm, which randomly exchanges elements to seek improvements over iterations to identify the global best LHDs. Following the work of (Morris & Mitchell, 1995a) and (Leary et al., 2003; Tang, 1993) proposed to construct orthogonal array-based LHDs (OALHDs) using the SA algorithm. They proposed to exchange elements that share the same original OA entry randomly. (Joseph & Hung, 2008) proposed a multi-objective criterion and developed a modified SA algorithm to generate optimal LHDs having good space-filling properties as well as orthogonality. This algorithm can lead to many good designs, but it is often computationally heavy, since it calculates all average pairwise correlations and row-wise distances at each iteration.

Orthogonal LHDs (OLHDs) are another type of optimal LHDs which aim to minimize the correlations between factors (Georgiou, 2009; Steinberg & Lin, 2006; Sun & Tang, 2017). Two correlation-based criteria are often used to measure designs' orthogonality: the average absolute correlation criterion and the maximum absolute correlation criterion (Georgiou, 2009), which are defined as

$$\rho^2 = \frac{2\sum_{i=1}^{k-1}\sum_{j=i+1}^{k}\rho_{ij}^2}{k(k-1)} \text{ and } \max\rho^2 = \max_{i,j}\rho_{ij}^2, \tag{4.6}$$

where ρ_{ij}^2 is the correlation between the ith and jth columns in the design matrix. Orthogonal designs may not exist for all sizes. In practice, designs with small ρ^2 or $\max(\rho^2)$ are preferred.

In literature, construction methods of OLHDs are widely explored. Specifically, (Ye, 1998) proposed a method to construct OLHDs with run sizes $n = 2^m + 1$ and factor sizes $k = 2m - 2$, where m is any integer no less than 2. (Cioppa & Lucas, 2007) extended the work of (Ye, 1998) to accommodate more factors. (Steinberg & Lin, 2006) developed a method based on factorial designs with group rotations for $n = 2^{2^m}$ and $k = 2^m t$, where m is any positive integer and t is the number of rotation groups. (Sun et al., 2010) improved their earlier work (Sun et al., 2009) to construct OLHDs with even more flexible run sizes: $n = r2^{c+1}$ or $n = r2^{c+1} + 1$ and $k = 2^c$, where c and r are any two positive integers. (J. Yang & Liu, 2012) proposed to use generalized orthogonal designs to construct OLHDs and nearly orthogonal LHDs (NOLHDs) with $n = 2^{r+1}$ or $n = 2^{r+1} + 1$ and $k = 2^r$, where r is any positive integer. (Georgiou & Efthimiou, 2014) proposed to take advantage of OAs and their full fold-overs for OLHDs with $n = 2ak$ runs and k factors, where k is the size of the orthogonal matrix and a is any positive integer. (Butler, 2001) implemented the Williams transformation (E. Williams, 1949) to construct OLHDs with odd prime run-size n and factor-size $k \leq n - 1$. (Lin et al., 2009) proposed to couple OLHDs or NOLHDs with OAs to accommodate large numbers of factors with fewer runs: n^2 runs and $2fp$ factors, where n and p are design sizes of the OLHDs or NOLHDs and $2f$ is the number of columns in the coupled OA.

Chapter 5

Lasso Surrogate for Complex Computer Experiments

5.1 Overview

As discussed in Chapter 4, computer experiments involve using complex simulations based on mathematical models derived from engineering or physical principles to replicate real-world phenomena. Because these simulations produce deterministic results, repeating the same input conditions (replicates) is unnecessary and should be avoided. Furthermore, following the effect sparsity principle, which states that only a small number of input variables typically have a significant impact, it is important to design experiments that minimize replicates, even when considering subsets of the input factors.

Latin hypercube designs (LHDs) (McKay et al., 1979) are widely used in computer experiments because they provide stratified sampling along each input dimension. However, LHDs generated randomly can be suboptimal. One issue is that their columns may exhibit high correlations, making it difficult to separate the effects of different factors. Another concern is that the design points (rows) might not be well distributed across the input space, limiting the design's ability to thoroughly explore the experimental region. To address these issues, as discussed in Chapter 4 various criteria have been proposed in the literature to enhance both the space-filling and correlation properties of LHDs.

Most space-filling designs emphasize coverage in the full-dimensional input space but may exhibit poor projection behavior onto lower-dimensional subsets (Joseph, Gul, & Ba, 2015). In computer experiments, where often only a small

subset of factors significantly influences the output, this can be problematic. To address this, maximum projection (MaxPro) LHDs are commonly used, as they enhance space-filling properties across all possible subsets of factors. Similar to orthogonal space-filling LHDs, which are suitable when all factors are active, orthogonal-MaxPro LHDs are more appropriate when only a few factors drive the system's behavior. Recently, (Wang et al., 2024) proposed an algorithm for an efficient construction of one-shot orthogonal-MaxPro LHDs.

A widely adopted strategy in statistical modeling is the use of regularization penalties during model fitting (Hoerl & Kennard, 1970). By minimizing a combination of empirical error and a penalty term, regularization aims to produce models that both fit the data well and avoid excessive complexity, thereby reducing variance. A notable example is the Lasso method (Tibshirani, 1996), which introduces an l_1-penalty to encourage sparsity in the coefficients. This often results in models with sparse solutions, enhancing interpretability, a key advantage in scientific and social science applications. In contrast, traditional model selection methods typically rely on computationally intensive combinatorial searches to identify sparse models. Orthogonal (or nearly orthogonal) LHDs, where the correlations between columns are zero (close to zero), are particularly effective for parameter estimation. Their structure allows for independent estimation of linear main effects, as the absence of correlation prevents confounding among input variables. In fact, orthogonal designs support consistent model selection when using methods like Lasso. Conversely, space-filling designs are valuable for thoroughly exploring the response surface, making it easier to identify significant factors.

The remainder of this chapter is organized as follows: in Section 5.2, we first introduce a new criterion, the weighted MaxPro criterion, designed to construct space-filling designs concerning significant factors. Building on this, in Section 5.3 we propose a multi-objective criterion, the orthogonal-weighted MaxPro criterion, which integrates both correlation and weighted MaxPro metrics to enhance design quality for important variables. To ensure that the weighted MaxPro criterion is well-defined, with a clear interpretation of weights and a bounded range from 0 to 1, we derive theoretical bounds specific to LHDs. Next, in Section 5.4 we present a sequential design strategy using active learning, resulting in Sequential Orthogonal Weighted MaxPro designs. To efficiently select the next design point, we develop a modified simulated annealing algorithm that is significantly faster than existing global search methods. Numerical results demonstrated in Section 5.5, show that our proposed algorithm efficiently generates Sequential Orthogonal Weighted MaxPro, which outperforms existing one-shot orthogonal LHDs and MaxPro designs in terms of both orthogonal-

ity and space-filling properties, as well as in terms of identifying significant and weakly significant factors.

5.2 Weighted MaxPro Criterion

Let p be the dimension of the input space, then consider p binary random variables corresponding to each input parameter as follows:

$$\theta_i = \begin{cases} 1 & \text{with probability } w_i \\ 0 & \text{with probability } 1 - w_i \end{cases}$$

, where $i = 1, \ldots, p$ and w_i is the inclusion probability for input parameter i, and $w_i = g(c_i)$ where c_i is the coefficient estimate obtained from Lasso regression. The random variables $\theta_1, \ldots, \theta_p$ are assumed to be independent but not identically distributed. The joint distribution of the vector $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_p)$ is given by

$$P(\boldsymbol{\theta}) = \prod_{i=1}^{p} w_i^{\theta_i} (1 - w_i)^{1-\theta_i}. \tag{5.1}$$

For instance, if $\boldsymbol{\theta}^* = (1, 0, 1, 0, \ldots, 0)$, then the corresponding joint probability is $P(\boldsymbol{\theta}^*) = w_1(1 - w_2)w_3(1 - w_4) \cdots (1 - w_p)$.

For illustration, consider the $p = 3$ case. Then, the vector $\boldsymbol{\theta}$ can take values in the set $\{(1, 0, 0), (0, 1, 0), (0, 0, 1), (1, 1, 0), (1, 0, 1), (0, 1, 1), (1, 1, 1)\}$, representing all non-zero combinations of three binary indicators. When a design is projected onto a subspace, the distances between the points are calculated with respect to the factors that define the subspace. Therefore, define a weighted Euclidean distance between the points x_i and x_j with weights $\boldsymbol{\theta}$, as

$$d^k(x_i, x_j; \boldsymbol{\theta}) = \left\{ \sum_{l=1}^{p} \theta_l (x_{il} - x_{jl})^2 \right\}^{k/2}.$$

, where $\theta_l = 1$ for all factors defining the subspace and $\theta_l = 0$ for the remaining factors. It makes sense to use weights between 0 and 1, which can be viewed as measures of importance for the factors, but we use 0 to denote insignificant factors and 1 to denote significant factors. By setting $k = 2p$, the expectation of the weighted reciprocal distance function $\phi_k(D; \boldsymbol{\theta})$, as introduced in

Equation 4.4, takes the following form:

$$E\left\{\phi_k(D;\boldsymbol{\theta})\right\} = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left\{\frac{w_1(1-w_2)(1-w_3)}{((x_{i1}-x_{j1})^2)^p} + \frac{(1-w_1)w_2(1-w_3)}{((x_{i2}-x_{j2})^2)^p} + \frac{(1-w_1)(1-w_2)w_3}{((x_{i3}-x_{j3})^2)^p} + \frac{w_1w_2(1-w_3)}{((x_{i1}-x_{j1})^2+(x_{i2}-x_{j2})^2)^p} + \cdots + \frac{w_1w_2w_3}{\left(\sum_{l=1}^{3}(x_{il}-x_{jl})^2\right)^p}\right\}.$$

Each term corresponds to a particular realization of $\boldsymbol{\theta}$, weighted by the product of the corresponding inclusion probabilities and their complements. This formulation provides an interpretable, probabilistically weighted criterion for space-filling based on the influence of input parameters.

For a general input dimension p, the expected value of the weighted reciprocal distance function $\phi_k(D;\boldsymbol{\theta})$ can be expressed as

Theorem 3 *If $k = 2p$, then under the prior in 5.1*

$$E\left\{\phi_k(D;\boldsymbol{\theta})\right\} = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left\{\sum_{m=1}^{p}\frac{(1-w_1)\cdots w_m\cdots(1-w_p)}{((x_{im}-x_{jm})^2)^p} + \sum_{1\le m<l\le p}\frac{(1-w_1)\cdots w_mw_l\cdots(1-w_p)}{((x_{im}-x_{jm})^2+(x_{il}-x_{jl})^2)^p} + \cdots + \frac{\prod_{m=1}^{p}w_m}{(\sum_{m=1}^{p}(x_{im}-x_{jm})^2)^p}\right\}.$$

This formulation enumerates all non-zero combinations of active input variables $\boldsymbol{\theta}$, each weighted according to its joint probability. Based on this, we propose the following new criterion for Weighted MaxPro Designs:

$$\min_{D}\psi_p(D) = \left\{\frac{1}{\binom{n}{2}}E\left\{\phi_k(D;\boldsymbol{\theta})\right\}\right\}^{1/p}. \tag{5.2}$$

This criterion balances space-filling properties across all possible projections, giving greater weight to projections aligned with more influential input variables as indicated by their inclusion probabilities.

The proposed criterion $\psi_p(D)$ satisfies several important properties that align with the principles of LHDs. First, the criterion is non-negative, i.e., $\psi_p(D) \geq 0$ for any design D. Second, if for any coordinate l, there exists a

pair of points $i \neq j$ such that $x_{il} = x_{jl}$, then the criterion evaluates to infinity, i.e., $\psi_p(D) = \infty$. This implies that to avoid singularities, the design must have distinct values in each factor for all design points. Consequently, any design that minimizes $\psi_p(D)$ must have n distinct levels for each factor. Therefore, the Latin hypercube property is inherently enforced by the criterion, without the need for additional constraints.

5.3 Orthogonal Weighted MaxPro Criterion

The objective is to identify a design D that simultaneously minimizes both the squared correlation measure ρ^2, defined in equation 4.6, and the weighted space-filling criterion $\psi_p(D)$, defined in equation 5.2. The straightforward idea is to adopt a weighted average $\tau_1\rho^2 + \tau_2\psi_p$, where τ_1 and τ_2 are some weights. However, it is important to note that $\rho^2 \in [0, 1]$, while the value of $\psi_p(D)$ is not bounded above and can exceed 1. To enable a meaningful trade-off between these two objectives, $\psi_p(D)$ must be rescaled to the unit interval. This is achieved by identifying a lower bound $\psi_{p,L}$ and an upper bound $\psi_{p,U}$ for ψ_p, and then normalizing it accordingly.

Theorem 4 *For any LHD (n, p), we have $0 < \psi_p \leq \psi_{p,U}$ with,*

$$\psi_{p,U} = \left\{ \binom{n}{2}^{-1} \sum_{i=1}^{n-1} \frac{n-i}{i^{2p}} \right\}^{1/p}.$$

Theorem 4 is the extension of Theorem 1 in (Wang et al., 2024) for the Weighted Max-Pro criterion. The upper bound $\psi_{p,U}$ is achieved by the worst-case scenario LHD (n, p):

$$D_{\text{worst}} = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & & \vdots \\ n-1 & n-1 & \cdots & n-1 \end{pmatrix}$$

and inclusion probability vector $\mathbf{w} = (1, 0, \ldots, 0)$, since only one factor is significant. This design is worse under both space-filling criteria and orthogonal criteria, which have all points on a straight line and a correlation of 1 between any two columns. For simulation purposes, we consider a loose lower bound $\psi_{p,L} = 0$, since calculating the strict lower bound would require certain distance constraints on the design points, which sequential design might not satisfy.

The resulting composite criterion is defined as

$$\Psi_p = \tau_1 \rho^2 + \tau_2 \left\{ \frac{\psi_p - \psi_{p,L}}{\psi_{p,U} - \psi_{p,L}} \right\}, \tag{5.3}$$

where $\tau_1$and$\tau_2 \in (0, 1)$ are some user-specified weights controlling the trade-off between orthogonality and space-filling. A design that minimizes the combined criterion Ψ_p is referred to as an *Orthogonal-Weighted MaxPro Design*.

5.4 Sequential Variable Selection Algorithm

(Morris & Mitchell, 1995b) developed a simulated annealing algorithm, referred to as MMA, to optimize the ϕ_k criterion. The algorithm starts from a randomly selected LHD and iteratively explores the design space by generating perturbed designs. Each perturbation X_{try} is created by randomly choosing a column of the current design X and interchanging two randomly selected entries within that column. If the perturbed design improves the value of ϕ_k, it is accepted as the new current design. The modification of the above algorithm was proposed by (Joseph & Hung, 2008).

In the original algorithm, a column and two elements within that column are chosen randomly to generate a perturbation. But to make better improvements, especially for our multi-objective purpose, it's better to choose them more carefully. As the search progresses, some columns may already become almost uncorrelated. Then, perturbing such columns won't help much. Instead, it's better to pick a column that is still highly correlated, because perturbing it may help reduce correlation and improve the objective function. Similarly, if a point is already far away from the others, there's no need to perturb the elements in that row. Instead, we should focus on points that are too close to others. Perturbing such points could increase their distance from the rest and thus improve the objective function. We propose a slightly modified version of the above algorithm to fit an active learning, i.e., sequential framework.

The proposed algorithm begins with a randomly selected MaxPro LHD consisting of n_{initial} runs. In each step of the active learning process, we randomly select a new point $\mathbf{x}_{\text{new}}$ to add to the design, resulting in a new design matrix X. To guide the search, we first compute a correlation measure for each column $l = 1, \ldots, p$, defined as

$$\rho_l^2 = \frac{1}{p-1} \sum_{j \neq l} \rho_{lj}^2.$$

We then identify the column l^* that is most correlated with the others and perform a first perturbation by swapping the value x_{new,l^*} with a randomly selected value. This gives a perturbed design X_{try}. If $\rho^2(X_{\text{try}}) \leq \rho^2(X)$, we accept the perturbation and update X. Next, to address the space-filling property, we compute the distance-based criterion between the new point and each existing point $i = 1, \ldots, n$ using the weighted MaxPro formulation:

$$\phi_{\text{new}}(X; \boldsymbol{\theta}) = \left\{ \sum_{m=1}^{p} \frac{(1-w_1)\cdots w_m \cdots (1-w_p)}{((x_{\text{new},m} - x_{i,m})^2)^p} + \sum_{1 \leq m < l \leq p} \frac{(1-w_1)\cdots w_m w_l \cdots (1-w_p)}{((x_{\text{new},m} - x_{i,m})^2 + (x_{\text{new},l} - x_{i,l})^2)^p} + \cdots + \frac{\prod_{m=1}^{p} w_m}{(\sum_{m=1}^{p}(x_{\text{new},m} - x_{i,m})^2)^p} \right\}.$$

We then identify the coordinate m^* in the row closest to $\mathbf{x}_{\text{new}}$ (in that coordinate) and perturb x_{new,m^*} by swapping it with a random value. If this reduces $\phi_{\text{new}}(X; \boldsymbol{\theta})$, we accept the perturbation. This modified $\mathbf{x}_{\text{new}}$ is then used as the starting point in a simulated annealing algorithm to optimize the orthogonal-weighted MaxPro criterion defined in Equation 5.3. In essence, the search for a new design point is initialized in a region that is far from existing points and will likely reduce the correlation between columns, resulting in a nearly orthogonal space-filling design.

Algorithm: Pseudo-code for Orthogonal-Weighted MaxPro Design and Variable Selection through Active Learning

---

- Define the number of factors p, initial design size n_{initial}, final design size n_{final}, number of active learning steps $sim = n_{\text{final}} - n_{\text{initial}}$, annealing temperature τ, and number of repetitions $\mathcal{M}$.
- Enumerate all $2^p - 1$ values of $\boldsymbol{\theta}$ (excluding the null model).
- Define the **Objective Function** Ψ_p and the **Simulation Function**.
- Compute lower and upper bounds $\psi_{p,L}$ and $\psi_{p,U}$ for normalization.

for $j = 1$ to $\mathcal{M}$

- **Data:** Start with initial design X of size n_{initial} and response vector Y.

  for $i = 1$ to sim

- Randomly generate a new point $\mathbf{x}_{\text{new}}$ and response y_{new}, then append to X and Y.
- Fit an initial Lasso model to identify significant variables.
- Compute weights $\mathbf{w}_i = g(\mathbf{c}_i)$, and evaluate $p(\boldsymbol{\theta})$ for each $\boldsymbol{\theta}$.
- **Modified Simulated Annealing:**
  * Identify column b that is highly correlated with others. Perturb it:
    $$X[\text{last row}, b] \leftarrow \text{runif}(1)$$
  * Identify row a that is closest to $\mathbf{x}_{\text{new}}$ in some dimension a_1. Perturb it:
    $$X[\text{last row}, a_1] \leftarrow \text{runif}(1)$$
  * Perform constrained optimization of Ψ_p using simulated annealing, with initial starting point = $X[\text{last row}, :]$.
- Fit the final Lasso model using the updated design.
- Store estimated coefficients and weight vector $\mathbf{w}_i$.

end for

- Store final coefficients, weights, and design matrix for repetition j.

end for

It is worth noting that the proposed exchange procedure can also be combined with other stochastic optimization algorithms, such as the columnwise-pairwise algorithm (Li & Wu, 1997; Ye et al., 2000), the threshold accepting heuristic (Winker & Fang, 1998), or the stochastic evolutionary algorithm (Jin et al., 2005).

5.5 Simulation Study

In this section, we benchmark our proposed method against several established techniques. To ensure a fair comparison, all parameters of the simulated annealing algorithm are set to the recommended values provided by (Morris & Mitchell, 1995a). To illustrate the proposed method, consider a synthetic test function incorporating a decreasing coefficient structure and four fake factors. The function is defined as

$$f(\boldsymbol{x}) = 0.2x_1 + \frac{0.2}{2}x_2 + \frac{0.2}{4}x_3 + \frac{0.2}{8}x_4 + \frac{0.2}{16}x_5 + \frac{0.2}{32}x_6 + \text{higher-order interaction terms.}$$

This linear function is intentionally constructed to evaluate the sensitivity of the algorithms. The coefficients decay geometrically, implying that variables associated with smaller coefficients are less influential, though not entirely negligible. For this simulation, we consider $p = 10$ input factors, with the first six having decreasing importance. The design begins with $n_{\text{initial}} = 10$ runs and is sequentially augmented to a final size of $n_{\text{final}} = 40$. Each scenario is replicated $\mathcal{M} = 100$ times to assess variability. To evaluate the ability of the design to identify important factors, we use two different threshold values to determine the significance of a factor based on its estimated Lasso coefficient. Any factor with a coefficient estimate below the threshold is deemed insignificant in the corresponding analysis.

Table 5.1 presents the discovery rates of each input factor under different design strategies and significance thresholds based on Lasso coefficient estimates. The goal is to assess the ability of each method to correctly identify both strongly active and weakly active features. A value of 1 indicates the feature was consistently identified as active across all replications, while lower values reflect decreasing frequency of detection.

Table 5.1: Discovery rate of active and weakly active features under two threshold levels.

Threshold	**Design**	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
0.005	Seq-W-Ortho-MaxPro	1	1	1	1	0.70	0.50	0.10	0.10	0.10	0.10
	One Shot MaxPro	1	1	1	1	1	0.10	0.10	0.10	0.10	0.10
	One Shot OLHD	1	1	1	1	1	0	0	0	0	0
0.01	Seq-W-Ortho-MaxPro	1	1	1	1	0.60	0.20	0	0	0	0
	One Shot MaxPro	1	1	1	1	0.50	0	0	0	0	0
	One Shot OLHD	1	1	1	1	0.04	0	0	0	0	0

The results indicate that all design methods consistently identify the most significant variables (x_1 to x_4). However, the proposed sequential Weighted-Orthogonal-MaxPro design shows a clear advantage in detecting weakly active features such as x_5 and x_6, particularly under tighter threshold levels. In contrast, one-shot designs like OLHD tend to miss these subtle effects entirely, while MaxPro captures them to a lesser extent. This highlights the strength of the sequential approach in adaptively exploring variable importance.

Figure 5.1 illustrate the properties of a design generated by the proposed sequential Weighted-Orthogonal MaxPro method. The correlation matrix confirms that the design is nearly orthogonal, a property that supports consistent

variable selection when using Lasso-based methods. Additionally, the scatter plot demonstrates that the design points are well-distributed throughout the input space, avoiding concentration near the boundaries. This space-filling behavior ensures effective exploration of the response surface, particularly within the interior of the domain, which is essential for building accurate surrogate models.

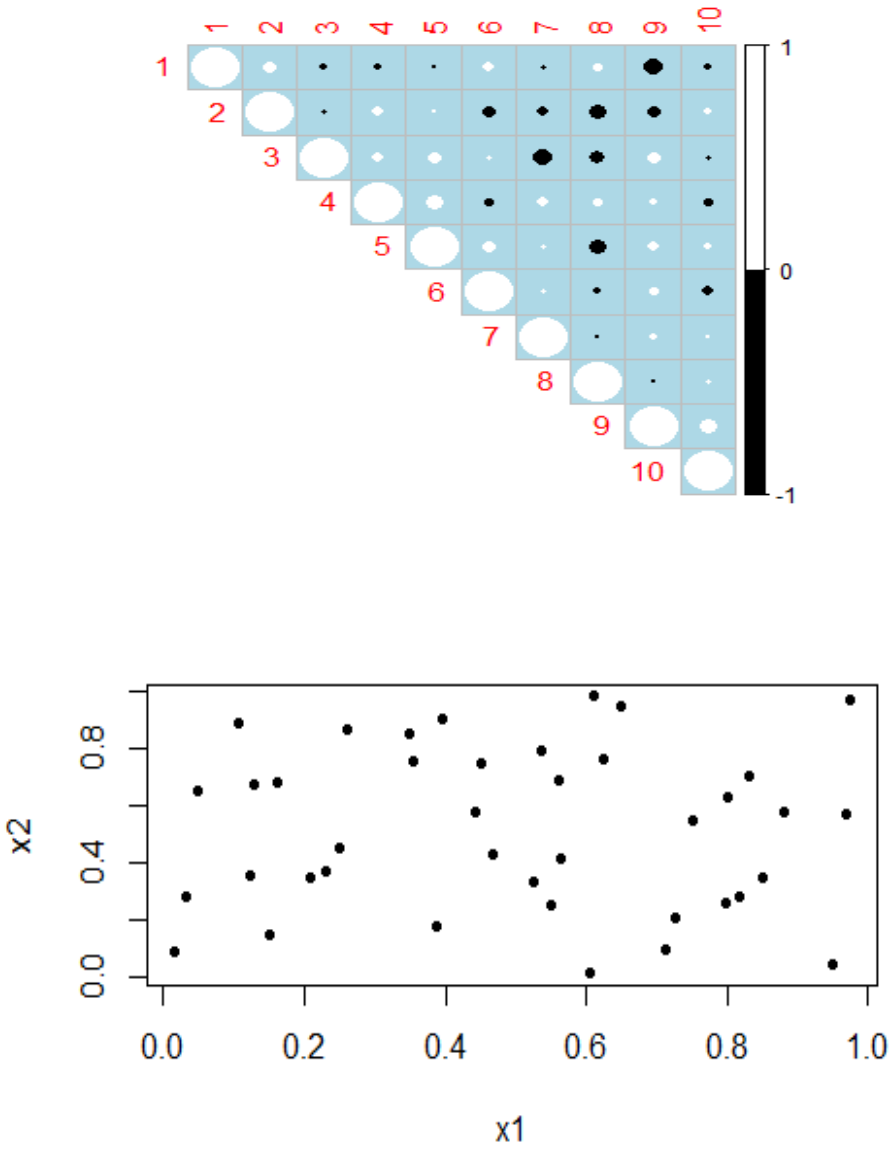


Figure 5.1: Illustration of the proposed design's nearly orthogonal (above) and space-filling (below) properties.

5.6 Discussion

In this chapter, we proposed a novel sequential design framework that constructs orthogonal space-filling designs tailored to the significant factors. Central to our approach is a new variable selection method that integrates active learning with Lasso regression and employs a weighted distance function to guide the sequential selection of variables. To identify informative design points, we formulated a multi-objective optimization strategy for generating high-quality Sequential Weighted MaxPro Designs and Sequential Orthogonal-Weighted MaxPro Designs. Although Lasso is used as the surrogate model in our implementation, the framework is flexible and can readily accommodate other surrogate models. Simulation results show that our sequential design performs comparably to traditional one-shot designs in identifying significant factors, while offering superior performance in detecting weakly active variables. The resulting designs are nearly orthogonal, as reflected in their correlation structure, which contributes to the model selection consistency of Lasso. Additionally, they are nearly space-filling and avoid boundary points, thus promoting thorough exploration of the response surface within the interior of the input space. These findings highlight the effectiveness of the proposed sequential design methodology in addressing the dual goals of factor screening and efficient surrogate modeling.

Chapter 6

Deep Gaussian Process Surrogate for Complex Computer Experiments

6.1 Overview

A computer experiment is a system of complex computer codes simulating a physical process, where inputs are varied to observe different outputs. Compared to a traditional laboratory experiment, this automation can reduce the cost, time, and/or management expenses (see, for example (Gramacy, 2020)). Computer experiments are often deterministic (specified inputs will always produce the same output), making the results more stable and less prone to random errors than physical experiments. Researchers can adjust the code to systematically explore a wide range of inputs and generate outputs based on the objective of the study. These computer experiments are instrumental in cases where a physical experiment would be impossible to conduct, such as modeling a black hole formation (Kidder et al., 2000).

As discussed in Chapter 4, computer experiments are often computationally intensive, despite the recent development of modern computing technology. To reduce the computational expenses, emulators (surrogate models) are used to rapidly generate many outputs. Emulators also allow uncertainty quantification, a quantitative characterization to determine how likely certain outcomes are if some aspects of the system are not exactly known. The Gaussian Process (GP) model is widely used as an emulator (Gramacy, 2020; Sacks et al., 1989). The GP assumes all observations follow a multivariate normal distribution, characterized by a mean vector and a variance-covariance matrix. The GP model would interpolate the observations, which is desirable for computer ex-

periments having deterministic outputs. It also allows for accurate uncertainty quantification for model outputs. By specifying different types of covariance functions, researchers may further include prior knowledge about the shape of the response surface.

The GP model has been applied to many computer experiments in chemistry, computational biology, robotics and others (Kruckow et al., 2018). As an illustration, it has accurately simulated the collision dynamics of complex molecules (Cui & Krems, 2015), the spread of COVID-19 (Velásquez & Lara, 2020), online heart rate prediction (Zhang et al., 2019) and autonomous learning in robots (Deisenroth et al., 2015). Data scientists at Microsoft introduced a framework that enables application of GP models to data sets containing millions of data points (Hensman et al., 2013). (Ek et al., 2008) used a Bayesian framework for tracking human body pose, as pictured in Figure 6.1. Instead of using computationally expensive Bayesian techniques, an efficient GP model can be used to take in a description of a human silhouette as input and identify human pose as an output (Zhu & Fujimura, 2010). Another useful application of the GP in Astronomy is modeling the collision of two black holes (D. Williams et al., 2019). Researchers cannot have black holes at their disposal to observe and experiment with, so computer experiments offer a viable alternative to incorporate the scientific knowledge and simulate their formation and collisions. Figure 6.2 illustrates that computer models and GP emulators are created based on the known properties of black holes and the surrounding system of space. They are compared to the naturally observed black hole movements to assess the accuracy of the computer model (D. Williams et al., 2019). Another interesting application of the GP is in car crash simulation to study the damage to the car (Bayarri et al., 2009). Here, models are validated by comparing simulation results with controlled physical crashes.

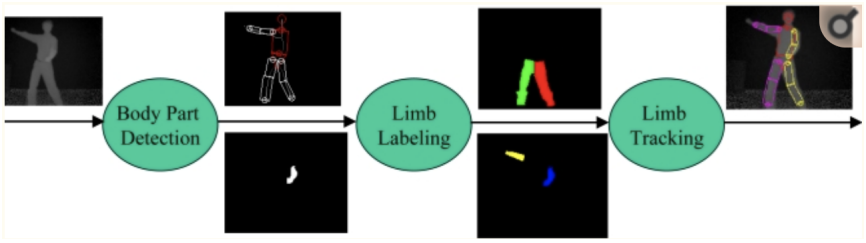


Figure 6.1: An example of a Bayesian framework for human pose tracking
Source: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3292173/ (Zhu & Fujimura, 2010)

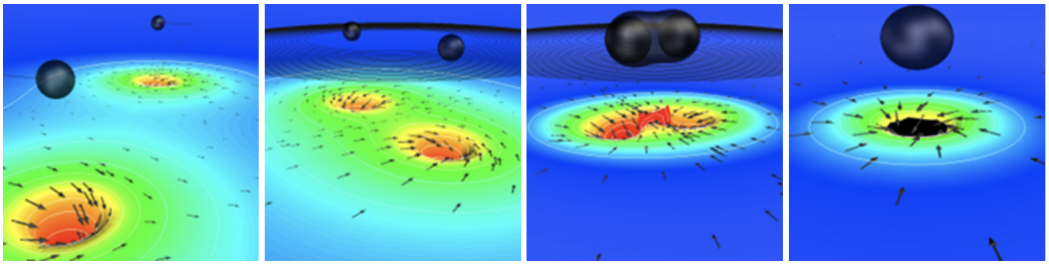


Figure 6.2: Computer simulation of two black holes colliding.
Source: https://www.black-holes.org/code/SpEC.html

The remainder of this chapter is organized as follows: in Section 6.2, we systematically review the GP models. Specifically, we discuss the ordinary and universal GP along with their model estimations and uncertainty quantification. In Section 6.3, details of the fully-Bayesian Deep Gaussian Process (DGP) are provided. In Section 6.4, the reference distribution variable selection (RDVS) methodology is introduced in detail, and a variable selection criterion is proposed. We finish the chapter with some concluding remarks in Section 6.5.

6.2 Shallow Gaussian Process

In this section, we aim to understand the GP as a flexible nonparametric regression for surrogate modeling in computer experiments. The GP is widely used in many statistical and probabilistic modeling enterprises. The GP is a very generic term, and all it means is that any finite collection of realizations is modeled as having a multivariate normal (MVN) distribution. That means a finite collection of n observations can be completely characterized by their mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

Let $y(\mathbf{x_i})$ be the output which is assumed to be a deterministic real-valued function of the p-dimensional variable $\mathbf{x_i} = (x_{i1}, \ldots, x_{ip})^T \in D \subset R^p$, for $i = 1, 2, \ldots, n$. Let $(Y_x)_{x \in D}$ be a square-integrable random field and y is a realization of $(Y_x)_{x \in D}$. Let $\mathbf{X}_n = \{\mathbf{x_1}, \ldots, \mathbf{x_n}\}$ be the points where their responses have been observed, which is denoted by $\mathbf{Y}_n = (y(\mathbf{x_1}), \ldots, y(\mathbf{x_n}))^T$. The aim of GP is to optimally predict Y_x by a linear combination of the observations $\mathbf{Y}_n$, for any $\mathbf{x} \in D$.

6.2.1 Model Formulation

Ordinary GP, a.k.a. ordinary Kriging, has the form

$$y(\mathbf{x_i}) = \mu + Z(\mathbf{x_i}), \tag{6.1}$$

where $\boldsymbol{\mu}$ is the mean vector and $Z(\mathbf{x_i})$ is a GP such that $Z(\mathbf{x_i}) \sim GP(0, \sigma^2\boldsymbol{\Sigma})$. In the above model, $Z(\mathbf{x_i})$ is GP with zero mean, and the covariance function $\phi(\cdot) = \sigma^2\boldsymbol{\Sigma}(\cdot|\boldsymbol{\theta})$, where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_p)^T$ is the vector of unknown correlation parameters with all $\theta_k > 0$ $(k = 1, \ldots, p)$ and $\boldsymbol{\Sigma}$ is a stationary correlation function that determines the correlation between inputs with parameters $\boldsymbol{\theta}$. The mean of the GP controls the trend, whereas the correlation function controls the smoothness of its sample paths. Power-exponential, Gaussian and Matérn correlation functions are the most widely used ones in literature.

In the power-exponential correlation structure, the $(i, j)^{\text{th}}$ element in the correlation matrix is defined as follows:

$$\boldsymbol{\Sigma}\left(\mathbf{x}_i, \mathbf{x}_j \mid \boldsymbol{\theta}\right) = \prod_{k=1}^{p} \exp\left\{-\theta_k \left|x_{ik} - x_{jk}\right|^{l_k}\right\} \quad \text{for all } i, j, \tag{6.2}$$

with two inputs $\mathbf{x_i} = (x_{i1}, \ldots, x_{ip})^T$ and $\mathbf{x_j} = (x_{j1}, \ldots, x_{jp})^T$ and smoothness parameters $l_1, \ldots, l_p$ which lie between 0 and 2, with 0 giving the most rough results and 2 giving the most smooth. If we take $l_k = 2$ for all $k = 1, \ldots, p$ then it results in the popular Gaussian correlation function:

$$\boldsymbol{\Sigma}\left(\mathbf{x}_i, \mathbf{x}_j \mid \boldsymbol{\theta}\right) = \prod_{k=1}^{p} \exp\left\{-\theta_k \left|x_{ik} - x_{jk}\right|^{2}\right\} \quad \text{for all } i, j. \tag{6.3}$$

The correlation functions of Matérn family is given by:

$$\boldsymbol{\Sigma}(\mathbf{h} \mid \boldsymbol{\theta}) = \prod_{k=1}^{p} \frac{1}{\Gamma(v)2^{v-1}} \left(\frac{2\sqrt{v}\,|h_k|}{\theta_k}\right)^{v} K_v\left(\frac{2\sqrt{v}\,|h_k|}{\theta_k}\right), \tag{6.4}$$

where $v > 0$ is a smoothness parameter, $\Gamma(\cdot)$ is the Gamma function, and $K_v(\cdot)$ is the modified Bessel function of order v. Two commonly used orders are $v = 3/2$ and $v = 5/2$.

Different correlation functions mentioned above impose different characteristics for function draws, allowing for different properties when modeling computer models. For example, when using the power exponential function, all

sample paths are infinitely differentiable when $h_k = 2$. For the Matérn correlation function, when we have $p = 1$, all sample paths are $\lceil v \rceil - 1$ differentiable. Hence, v is viewed as a smoothness parameter.

In the literature, two important assumptions are often imposed on the ordinary GP model to effectively analyze computer experiments. One assumption is that the GP is separable (Doob & Doob, 1953), which means finite-dimensional distributions can determine sample path properties of function draws, which are usually infinite-dimensional. The second important assumption is that the model is stationary. Consider $\{\mathbf{x_1}, \ldots, \mathbf{x_n}\} \in D$ and any $h \in R$, then a GP model is said to be stationary if the random vectors $(Y(\mathbf{x_1}), \ldots, Y(\mathbf{x_n}))$ and $(Y(\mathbf{x_1} + h), \ldots, Y(\mathbf{x_n} + h))$ follow the same distribution. This means that both of these random vectors should have the same mean and covariance.

The second assumption is restrictive, and we may need more flexibility while modeling computer experiments. One popular approach is to extend the above ordinary GP model to incorporate a global trend function for the mean part. This extended model is known as *Universal Kriging*, which has the form:

$$y(\mathbf{x}) = \mu(\mathbf{x}) + Z(\mathbf{x}), \tag{6.5}$$

with $\mu(\mathbf{x}) = \mathbf{f}(\mathbf{x})^T \boldsymbol{\beta} = \sum_{s=1}^{m} \beta_s f_s(\mathbf{x})$, where $\mathbf{f}$ is a m-dimensional known function and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_m)^T$ is a vector of unknown parameters. The idea is to rely on functions in $\mathbf{f}(\mathbf{x})$ to detrend the process and then model any residual variation as a zero-mean stationary GP. Taking the constant mean $\mathbf{f}(\mathbf{x}) \equiv 1$ results in the ordinary GP model discussed above. The stationary correlation functions discussed above in Equations (6.2) and (6.4) can also be applied here, that is,

$$Cov\left(Z(\mathbf{x} + \mathbf{h}), Z(\mathbf{x})\right) = \sigma^2 \Sigma(\mathbf{h}),$$

where correlation function $\Sigma(\mathbf{h})$ is positive semi-definite function with $\Sigma(\mathbf{0}) = 1$ and $\Sigma(\mathbf{h}) = \Sigma(-\mathbf{h})$.

6.2.2 Estimation and Uncertainty Quantification

In this section, we present equations used for predicting and quantifying uncertainty on $y(\mathbf{x})$ given observed responses $\mathbf{Y}_n = (y(\mathbf{x_1}), \ldots, y(\mathbf{x_n}))^T$. The question we are trying to answer is: given examples of function in pairs $\mathbf{D}_n = (\mathbf{x_1}, y(\mathbf{x_1})), \ldots, (\mathbf{x_n}, y(\mathbf{x_n})) = (\mathbf{X}_n, \mathbf{Y}_n)$, what random function realizations could explain or could have generated those observed values? In other words, we want to calculate the conditional distribution $(Y(\mathbf{x_1}), \ldots, Y(\mathbf{x_n}))\,|\mathbf{D}_n$.

Before we calculate the *predictive distribution*, we need to address the key question of how the parameters $\boldsymbol{\beta}, \sigma^2$ and $\boldsymbol{\theta}$ are estimated from data $\mathbf{D}_n$. The most popular approach for parameter estimation is *maximum likelihood estimation* (MLE), and the log-likelihood function under the above assumed GP model can be written as:

$$\log L\left(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\theta}\right) = -\frac{1}{2}\left[n \log \sigma^2 + \log \det \boldsymbol{\Sigma}_\theta + \frac{1}{\sigma^2}\mathbf{R}^T \boldsymbol{\Sigma}_\theta^{-1} \mathbf{R}\right], \quad (6.6)$$

where $\boldsymbol{\Sigma}_\theta = [\Sigma(\mathbf{x_i}, \mathbf{x_j})]_{i=1}^{n}{}_{j=1}^{n}$, $\mathbf{R} = (\mathbf{y} - \mathbf{F}\boldsymbol{\beta})$ and $\mathbf{F} = [f_s(\mathbf{x_i})]_{i=1}^{n}{}_{s=1}^{m}$. Hence, the MLEs for $(\boldsymbol{\beta}, \sigma^2, \boldsymbol{\theta})$ are the parameter estimates that maximize the above log-likelihood function. ML estimates of $(\boldsymbol{\beta}, \sigma^2)$ for fixed value of $\boldsymbol{\theta}$ can be easily obtained as follows:

$$\hat{\boldsymbol{\beta}}_{\boldsymbol{\theta}} = \left(\mathbf{F}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \mathbf{F}\right)^{-1} \mathbf{F}^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1} \mathbf{y} \quad (6.7)$$

and

$$\hat{\sigma}_\theta^2 = \frac{1}{n}\left(\mathbf{y} - \mathbf{F}\hat{\boldsymbol{\beta}}_{\boldsymbol{\theta}}\right)^T \boldsymbol{\Sigma}_{\boldsymbol{\theta}}^{-1}\left(\mathbf{y} - \mathbf{F}\hat{\boldsymbol{\beta}}_{\boldsymbol{\theta}}\right) \quad (6.8)$$

Substituting these ML estimates back into Equation (6.6), we get the profile likelihood function as follows:

$$\log L\left(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2, \boldsymbol{\theta}\right) = -\frac{1}{2}\left[n \log \hat{\sigma}^2 + \log \det \boldsymbol{\Sigma}_\theta + n\right], \quad (6.9)$$

where the MLE of $\boldsymbol{\theta}$ is one that maximizes above function in Equation (6.9). This optimization problem does not enjoy a closed form solution, and numerical methods, e.g. quasi-Newton algorithms (Nocedal & Wright, 2006) are used for solving the problem.

Once we have estimates of parameters, we can calculate the conditional distribution as mentioned above. Let $\left(\hat{\boldsymbol{\beta}}, \hat{\sigma}^2, \hat{\boldsymbol{\theta}}\right)$ denote the ML estimates of unknown parameters for a given GP model. Then for a new input $\mathbf{x}^* \in R^p$, the mean and variance of random variable $Y(\mathbf{x}^*|\mathbf{y})$ is as follows:

$$\hat{y}\left(\mathbf{x}^*\right) = E\left[Y\left(\mathbf{x}^*\right) \mid \mathbf{y}\right] = \mathbf{f}^T\left(\mathbf{x}^*\right)\hat{\boldsymbol{\beta}} + \mathbf{r}_{\hat{\boldsymbol{\theta}}}^T\left(\mathbf{x}^*\right)\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}^{-1}\left(\mathbf{y} - \mathbf{F}\hat{\boldsymbol{\beta}}\right), \quad (6.10)$$

$$s\left(\mathbf{x}^*\right)^2 = \mathrm{Var}\left[Y\left(\mathbf{x}^*\right) \mid \mathbf{y}\right] = \hat{\sigma}^2\left(1 - \mathbf{r}_{\hat{\boldsymbol{\theta}}}^T\left(\mathbf{x}^*\right)\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}^{-1}\mathbf{r}_{\hat{\boldsymbol{\theta}}}\left(\mathbf{x}^*\right)\right), \quad (6.11)$$

where the covariance vector $\mathbf{r}_{\hat{\boldsymbol{\theta}}}\left(\mathbf{x}^*\right) = \left[\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}\left(\mathbf{x}^*, \mathbf{x}_1\right), \ldots, \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}\left(\mathbf{x}^*, \mathbf{x}_n\right)\right]^{\mathrm{T}}$.

When some observed data points are very close to each other, the covariance matrix $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}$ may become nearly singular, making it difficult to obtain a stable inverse matrix $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}^{-1}$. This is a common issue for GP models when the run and/or factor sizes are large. One way to deal with this problem is to add a positive scalar g, called the *nugget* parameter, to the diagonal elements in $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}$, i.e., replacing $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ with $\boldsymbol{\Sigma}_{\boldsymbol{\theta}} + g\mathbf{I}$, where $\mathbf{I}$ is an identity matrix. Adding g is analogous to adding the ridge parameter in ridge regression, which helps in moving the smallest eigenvalue of $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$ away from zero, thus stabilizing the calculation of its inverse.

For large data size, the estimation of GP models can be very time consuming, mainly due to the matrix inverse calculation of order $O(n^3)$. To deal with this problem, (Gramacy & Apley, 2015) proposed a localize GP (LaGP) approach. Based on a local subset of the data, they provide a family of local sequential design schemes that define a GP predictor's support points. The idea is to ensure that for a given choice of covariance structure, the data points far from the target location $\mathbf{x}^*$ will have little effect on the prediction. Hence, it is not unwise to calculate the inverse of the full covariance matrix, as the elements corresponding to "far away" points will contribute very little to predicting $y(\mathbf{x}^*)$. Interested readers may refer to (Gramacy & Apley, 2015) for further details.

Given $\mathbf{D}_n$, and under settings of hyperparameter (either MLE or via posterior sampling), the posterior predictive distribution for an $n^{'} \times d$ matrix of multiple testing locations $\mathcal{X}$ has a closed form and follows a multivariate normal distribution:

$$Y(\mathcal{X}) \mid \mathbf{D}_n \sim \mathcal{N}\left(\boldsymbol{\mu}_Y(\mathcal{X}), \boldsymbol{\Sigma}_Y(\mathcal{X})\right).$$

The predictive mean and covariance are given by

$$\boldsymbol{\mu}_Y(\mathcal{X}) = \boldsymbol{\Sigma}\left(\mathcal{X}, \mathbf{X}_n\right)\boldsymbol{\Sigma}_n^{-1}\mathbf{Y}_n,$$

$$\boldsymbol{\Sigma}_Y(\mathcal{X}) = \boldsymbol{\Sigma}(\mathcal{X}) - \boldsymbol{\Sigma}\left(\mathcal{X}, \mathbf{X}_n\right)\boldsymbol{\Sigma}_n^{-1}\boldsymbol{\Sigma}\left(\mathbf{X}_n, \mathcal{X}\right),$$

where $\boldsymbol{\Sigma}\left(\mathcal{X}, \mathbf{X}_n\right)$ is an $n \times n^{'}$ matrix derived by extending the kernel across training and testing locations. These expressions allow the GP to interpolate training observations and quantify uncertainty in its predictions at testing input locations.

The GP model described is stationary because it depends solely on the relative distances between training and testing inputs (as in Equation 6.3). This implies that the same input-output relationship is assumed to hold throughout the entire input space, which can be restrictive in certain computer simulations. A notable example arises in aeronautics or computational fluid dynamics, where

lift forces on an aircraft vary significantly between low and high speeds, especially near the speed of sound, where an abrupt transition occurs (Pamadi et al., 2004). This assumption of stationarity poses a limitation and is further compounded by computational challenges. Even when the stationarity constraint is removed, detecting distinct behavioural regimes typically demands a large volume of training data.

6.3 Deep Gaussian Process

A deep Gaussian process (DGP) is a hierarchical extension of the standard GP model, in which each layer produces a multivariate normal distribution conditioned on the previous layer (Damianou & Lawrence, 2013). (Dunlop et al., 2018) outlined four distinct frameworks for constructing DGPs, each balancing computational feasibility with interpretability in different ways. Among these, treating DGPs as functional compositions offers a particularly intuitive and easily implemented perspective.

In this formulation, the input data $\mathbf{X}_n$ is passed through one or more hidden GP layers before producing the final response $\mathbf{Y}_n$. These hidden layers introduce latent variables that are not observed directly but serve to transform the input space in a nonlinear manner. This warping can help approximate stationarity in regions of the input space even when the global behavior is nonstationary. From this point forward, we refer to standard GP regression (as introduced in Section 6.2) as a "single-layer" GP.

In a two-layer DGP, we define a latent variable matrix $\mathbf{W}$, where each column corresponds to a latent feature or "node." Each $\mathbf{W}_k$ is modeled as a GP over $\mathbf{X}_n$, while $\mathbf{Y}_n$ is modeled as a GP over the latent space $\mathbf{W}$. The model can be specified hierarchically as:

$$\mathbf{Y}_n \mid \mathbf{W} \sim \mathcal{N}_n\left(0, \mathbf{\Sigma}(\mathbf{W})\right), \tag{6.12}$$

$$\mathbf{W}_k \sim \mathcal{N}_n\left(0, \mathbf{\Sigma}_k(\mathbf{X}_n)\right), \quad k = 1, \ldots, p. \tag{6.13}$$

Here, $\mathbf{W} = [\mathbf{W}_1, \ldots, \mathbf{W}_p]$ is an $n \times p$ matrix in which each row corresponds to one input point and each column to a latent dimension. Each latent node has its own kernel function $\mathbf{\Sigma}_k(\mathbf{X}_n)$, which may differ across dimensions in terms of kernel type or hyperparameters.

Although deeper networks may provide increased model flexibility, these gains tend to diminish beyond a certain depth. Damianou and Lawrence (Damianou & Lawrence, 2013) demonstrated that a five-layer DGP can be beneficial

in some classification tasks, but further layering may not yield significant additional improvements. Two- and three-layer DGPs have been sufficient for real-valued outputs common to computer surrogate modeling (Radaideh & Kozlowski, 2020).

(Dunlop et al., 2018) advocate for architectures with only two or three layers, a recommendation that aligns well with our proposed method. In this work, I constrain the number of latent nodes to be no greater than the input dimension and limit the network depth to three layers or fewer. Furthermore, fixing the number of latent nodes equal to the input dimension allows us to perform variable selection for DGPs more effectively. Nonetheless, the methods and implementation I propose are not inherently confined to these architectural choices.

Given covariance functions $\boldsymbol{\Sigma}(\cdot)$ and $\boldsymbol{\Sigma}_k(\cdot)$, the marginal likelihood of the observed outputs $\mathbf{Y}_n$ given inputs $\mathbf{X}_n$ is formulated as an integral over the latent variables $\mathbf{W}$:

$$L(\mathbf{Y}_n \mid \mathbf{X}_n) = \int L(\mathbf{Y}_n \mid \mathbf{W}) \prod_{k=1}^{p} L(\mathbf{W}_k \mid \mathbf{X}_n)\, d\mathbf{W}, \tag{6.14}$$

where $\log L(\mathbf{Y}_n \mid \mathbf{W})$ and $\log L(\mathbf{W}_k \mid \mathbf{X}_n)$ follow analogous forms to Equation 6.6. A three-layer DGP introduces two latent layers, typically denoted $\mathbf{Z}$ and $\mathbf{W}$, and corresponds to three stacked GP mappings. In this setting, the marginal likelihood becomes a double integral over both $\mathbf{Z}$ and $\mathbf{W}$.

As inputs $\mathbf{X}_n$, often assumed uniformly distributed, are passed through successive layers, their representations are nonlinearly warped. This warping breaks the original stationarity of the process, changing the distribution of outputs in meaningful ways. Such transformations have notable consequences for active learning (AL). The induced nonstationarity effectively reshapes distances in the latent space, guiding the acquisition function toward regions of higher informativeness rather than promoting uniform coverage. This adaptation improves the efficiency of sample selection. However, fully Bayesian treatment of DGPs with two or more layers is analytically intractable, since the marginal likelihood in Equation (6.14) cannot be evaluated in closed form due to the need to integrate out latent variables. Variational inference and other optimization-based methods (Damianou & Lawrence, 2013; Salimbeni & Deisenroth, 2017) are often computationally efficient, but they tend to produce posterior approximations that may oversimplify uncertainty quantification (UQ), potentially under-representing the true variability in the model.

6.3.1 Model Formulation

In DGPs, extreme flexibility often comes at the cost of identifiability and computational practicality. To mitigate these challenges, we use a modeling template that has demonstrated strong performance in surrogate modeling across several realistic applications and is also well-suited to support downstream tasks such as active learning (AL).

A two-layer DGP configuration, where the hierarchical structure is defined via distance-based covariance functions, is expressed as:

$$\mathbf{Y}_n \mid \mathbf{W} \sim \mathcal{N}_n\left(0, \sigma^2\left(K_{\theta_y}(\mathbf{W}) + gI_n\right)\right),$$

$$\mathbf{W}_k \overset{\text{iid}}{\sim} \mathcal{N}_n\left(0, K_{\theta_w[k]}(\mathbf{X}_n)\right), \tag{6.15}$$

where $k = 1, \ldots, p$ and the response vector $\mathbf{Y}_n$ is conditionally Gaussian given the latent layer $\mathbf{W}_n$, with covariance scaled by σ^2 and stabilized using a nugget term g. The hidden layers are assumed to be noiseless and unit-scaled, which empirically improves the stability and reliability of posterior inference.

As mentioned earlier, setting $p = \dim(\mathbf{W}) = \dim(\mathbf{X}_n)$ is generally effective in variable selection tasks, although smaller values for p may be advisable in high-dimensional input spaces. Each $\mathbf{W}_k$ corresponds to a latent feature evaluated across the n training points, analogous to a column of the input matrix $\mathbf{X}_n$ or the response vector $\mathbf{Y}_n$. Although a joint model over $\mathbf{W}$ incorporating cross-covariances is possible (e.g., see Schmidt and O'Hagan, 2003), we recommend the following simplifying assumptions suggested in (Sauer et al., 2022):

1. The latent dimensions $\mathbf{W}_j$ and $\mathbf{W}_k$ are conditionally independent given $\mathbf{X}$, for all $j \neq k$ (Salimbeni & Deisenroth, 2017).

2. Each $\mathbf{W}_k$ is modeled using an isotropic kernel (6.3) in inputs $\mathbf{X}_n$ with its own scalar lengthscale $\theta_w[k]$, for each $k = 1, \ldots, p$, regardless of the input dimension d.

3. The output $\mathbf{Y}_n$ is also modeled isotropically over the latent representation $\mathbf{W}$, using a scalar lengthscale θ_y.

Figure 6.3 presents a two-layer model architecture, where the latent layer $\mathbf{W}$ serves as input to the observed output $\mathbf{Y}$. Two modeling constraints, (i) conditional independence and (ii) isotropy, are key to limiting the model's complexity while preserving flexibility. Notably, even a low-dimensional latent space

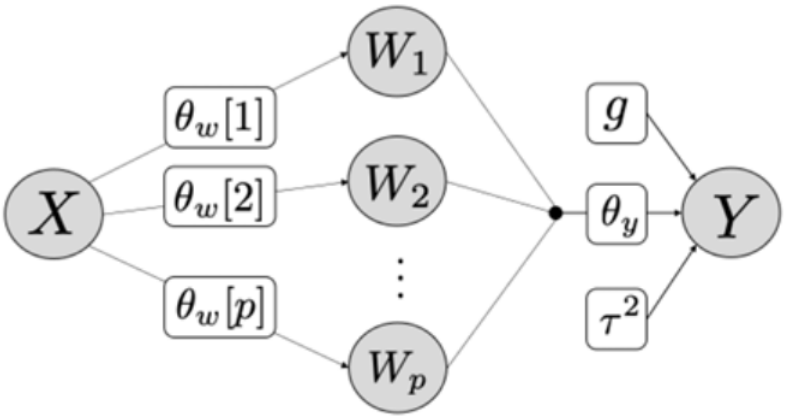


Figure 6.3: Two Layer Deep Gaussian Process
Source: (Sauer et al., 2022)

(e.g., a single-node $\mathbf{W}$ layer) can approximate a wide range of kernel behaviors. When $p \gg 1$, this flexibility becomes especially useful. For instance, when the dimension of latent space is p, the model can emulate anisotropic or separable covariance structures without assigning distinct lengthscales to each input dimension. If the data indicate weaker correlation along one coordinate axis, components of $\mathbf{W}$ can self-organize to reflect that, for example, $\mathbf{W}_j$ may exhibit greater variability than $\mathbf{W}_k$. The filled circle between $\mathbf{W}$ and $\mathbf{Y}$ in the diagram visually encodes constraint (iii): the entire $\mathbf{W}$ vector influences the output layer $\mathbf{Y}$ via a GP with a shared, isotropic lengthscale θ_y.

This structure can be extended to a three-layer model by introducing an additional latent layer $\mathbf{Z}$. The generative structure becomes:

$$\begin{aligned}
\mathbf{Y} \mid \mathbf{W} &\sim \mathcal{N}_n\left(0, \tau^2\left(K_{\theta_y}(\mathbf{W}) + g\mathbf{I}_n\right)\right), \\
\mathbf{W}_k \mid \mathbf{Z} &\overset{\text{ind}}{\sim} \mathcal{N}_n\left(0, K_{\theta_w[k]}(\mathbf{Z})\right), \quad k = 1, \ldots, p, \\
\mathbf{Z}_j &\overset{\text{ind}}{\sim} \mathcal{N}_n\left(0, K_{\theta_z[j]}(\mathbf{X})\right), \quad j = 1, \ldots, p.
\end{aligned} \tag{6.16}$$

As with the earlier model, both $\mathbf{W}$ and $\mathbf{Z}$ are governed by conditional independence and isotropic GP priors. To maintain a balance between expressiveness and tractability, it is often helpful to match the dimension of $\mathbf{Z}$ with that of $\mathbf{W}$, i.e., $\dim(\mathbf{Z}) = \dim(\mathbf{W}) = p$. Each $\mathbf{Z}_j$ is an n-dimensional latent variable vector. Figure 6.4 depicts this extended model, with the filled circle connecting the $\mathbf{Z}$ and $\mathbf{W}$ layers, with bidirectional edges extending to each node, represent-

ing a fully connected structure between the two layers. This "dense interaction structure" encodes the p^2 possible dependencies between the components of $\mathbf{Z}$ and $\mathbf{W}$. In this setup, each latent variable $\mathbf{Z}_j$ (for $j = 1, \ldots, p$) contributes to the generation of every $\mathbf{W}_k$ (for $k = 1, \ldots, p$). These relationships are governed by GP kernels based on inverse distances. Each of these kernels uses a scalar lengthscale parameter $\theta_w[k]$ to modulate the smoothness of the mapping from $\mathbf{Z}$ to the corresponding $\mathbf{W}_k$.

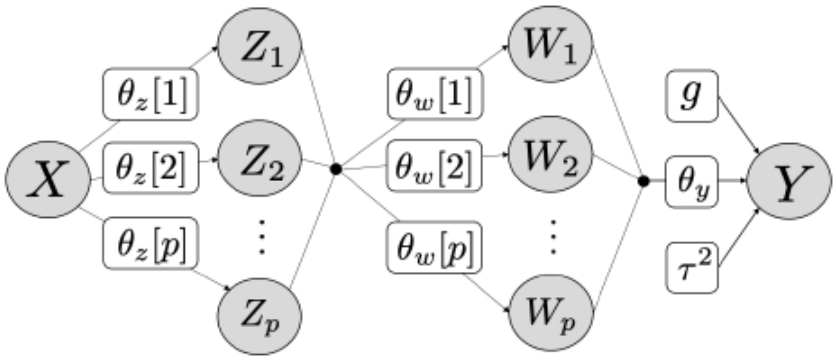


Figure 6.4: Three Layer Deep Gaussian Process
Source: (Sauer et al., 2022)

6.3.2 Hyperparemter Prior

One of the most widely used covariance functions in GP modeling is the *radial basis function* (RBF) kernel, also known as the squared exponential kernel. Its functional form is:

$$C(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp\left(-\frac{1}{2}\sum_{k=1}^{p}\frac{(x_{i,k} - x_{j,k})^2}{\theta_k^2}\right),$$

where σ^2 is the marginal variance, θ_k is the lengthscale parameter associated with the k-th input dimension, and the hyperparameters that control the shape of the GP.

Despite its popularity, the standard RBF kernel poses interpretational challenges when applied in the context of variable selection. Specifically, the meaning of the lengthscale parameters becomes less intuitive: larger lengthscales imply lower importance of the associated input dimensions, which complicates regularization since we typically prefer to penalize irrelevant inputs. To address

this issue, we adopt a reparameterized version of the RBF kernel, referred to here as the *inverse-RBF* kernel, defined as:

$$C(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp\left(-\frac{1}{2}\sum_{k=1}^{p} \theta_k^2 (x_{i,k} - x_{j,k})^2\right), \tag{6.17}$$

where θ_k denotes the inverse lengthscale parameter for the k-th input dimension. In this formulation, large values of θ_k imply high sensitivity of the covariance to variations in the k-th input, indicating that the input is important. Conversely, if θ_k is near zero, the k^{th} input has a negligible effect on the covariance, effectively removing it from the model. This makes the interpretation of θ_k analogous to that of coefficients in standard generalized linear models (GLMs), facilitating regularization and model selection.

To enable variable selection in DGP models, it is necessary to regularize the inverse length-scales so that uninformative input dimensions are shrunk toward zero. This requires a prior distribution that places substantial mass near zero. Since the inverse length-scales θ_w^2 and θ_z^2 are constrained to be nonnegative, we place an exponential prior on them: $\pi(\theta_w^2) \propto \exp(b_{[\theta_w]}\theta_w^2)$ and $\pi(\theta_z^2) \propto \exp(b_{[\theta_z]}\theta_z^2)$, which corresponds to a Gamma distribution $\theta_w^2 \sim \mathrm{G}\left(1, \frac{1}{b_{[\theta_w]}}\right)$ and $\theta_z^2 \sim \mathrm{G}\left(1, \frac{1}{b_{[\theta_z]}}\right)$ respectively. For other hyperparameters, such as θ_y and the nugget g, we adopt independent Gamma priors of the form $\{\theta_y, g\} \overset{\text{iid}}{\sim} \mathrm{G}\left(3/2, \frac{1}{b_{[\cdot]}}\right)$, where the rate parameter $b_{[\cdot]}$ is chosen based on the specific parameter. A hierarchical structure is encouraged by setting $b_{[\theta_y]} \leq b_{[\theta_w]} \leq b_{[\theta_z]}$, reflecting the prior belief that deeper layers should exhibit smoother (less wiggly) behaviour. In cases where the outer layer models a deterministic computer simulation, the nugget g is fixed to a small constant $\epsilon > 0$.

6.3.3 Posterior Distribution for DGPs

The posterior distribution in a two-layer DGP model is derived by first expressing the log-likelihood of the observed data $\mathbf{Y}_n$ conditioned on the latent variables $\mathbf{W}$, the length-scale θ_y, and the nugget g:

$$\log \mathcal{L}\left(\mathbf{Y}_n \mid \mathbf{W}, \theta_y, g\right) \propto -\frac{n}{2}\log(n\hat{\sigma}^2) - \frac{1}{2}\log\left|K_{\theta_y}(\mathbf{W}) + gI_n\right|,$$

where the variance estimator is defined as $\hat{\sigma}^2 = \mathbf{Y}_n^\top \left(K_{\theta_y}(\mathbf{W}) + gI_n\right)^{-1} \mathbf{Y}_n / n$. The full log-likelihood for the DGP is the sum of this outer-layer term and the

log-likelihood of the latent variables $\mathbf{W}$ under the GP prior given $\mathbf{X}_n$ and θ_w:

$$\log \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \mathbf{X}_n, \theta_y, \theta_w, g) = \log \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \theta_y, g) + \log \mathcal{L}(\mathbf{W} \mid \mathbf{X}_n, \theta_w).$$

Combining this likelihood with the prior distributions yields the joint posterior:

$$\pi(\mathbf{W}, \theta_w, \theta_y, g \mid \mathbf{D}_n) \propto \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \mathbf{X}_n, \theta_w, \theta_y, g) \times \pi(\theta_y, g) \times \pi(\theta_w).$$

Taking logarithms and simplifying, the log-posterior becomes

$$\begin{aligned} \log \pi(\mathbf{W}, \theta_w, \theta_y, g \mid \mathbf{D}_n) \propto \log \mathcal{L}(\mathbf{Y}_n \mid \mathbf{W}, \mathbf{X}_n, \theta_w, \theta_y, g) \\ + \log \pi(\theta_y, g) - b_{[\theta_w]} \sum_{k=1}^{p} \theta_{w,k}^2. \end{aligned}$$

The final term, $b_{[\theta_w]} \sum_{k=1}^{p} \theta_{w,k}^2$, acts as a regularization penalty, shrinking large values of θ_w^2 and thus encouraging sparsity in the latent representation. Increasing the rate parameter $b_{[\theta_w]}$ intensifies this shrinkage effect, offering a mechanism to control model complexity.

6.4 Variable Selection Algorithm

GP models provide a highly adaptable approach for modeling response surfaces, offering significantly greater flexibility than conventional linear or polynomial regression techniques. This adaptability enables GPs to naturally accommodate complex features such as non-linear trends and interactions among input variables. However, this same flexibility introduces difficulties in determining which input factors have a substantial effect on the response and which can be regarded as negligible. With p input variables, the number of possible combinations of active and inactive factors is 2^p, resulting in a vast model space. (Chipman et al., 2001) presents an insightful treatment of how model priors can be assigned in such high-dimensional contexts. In a fully Bayesian setting, variable screening often entails specifying prior beliefs across all 2^p potential models, an undertaking that is not only computationally intensive but also conceptually demanding. Furthermore, the decisions regarding which variables to retain are frequently influenced by the prior structure, making them inherently subjective and potentially unstable under different prior choices.

In this section, we extend the reference distribution variable selection (RDVS) method introduced by (Linkletter et al., 2006) to DGPs. The task of determining which individual estimates of $\theta_w[k]$ are sufficiently small, meaning they

deviate substantially from one, to indicate the presence of a significant factor, is conceptually similar to the frequentist methodology for variable selection. The main challenge lies in defining an appropriate reference distribution and a selection criterion that allows for the systematic evaluation of each factor's relevance in the experimental setting.

To outline the approach, consider two-layer DGP model outlined in equation 6.15. Because the relevant factors are unknown at the outset, identifying them is the primary objective of this work. It can be challenging to directly interpret the relative magnitudes of the $\theta_w[k]$ values. In RDVS, an artificial variable that is known to be inert, that is, it does not influence the response, is added to the design matrix. This known null variable provides a baseline, offering insight into how an inactive factor typically behaves. By comparing the behavior of actual factors against this benchmark, one can more reliably assess their importance. Specifically, using the distribution of the posterior median for the inert variable as a reference distribution for evaluating which of the real factors are active.

Consider a design matrix $\mathbf{X}$ of dimension $n \times p$, where each row corresponds to the settings of p continuous variables in an individual experimental run. To introduce a reference (null) variable into the analysis, construct an expanded design matrix of size $n \times (p+1)$ by adding a new column, denoted $\mathbf{X}_{\text{inert}} = (x_{1,0}, x_{2,0}, \ldots, x_{n,0})^T$. This added column is designed to resemble the structure of the original covariates, with values spanning the interval $[0, 1]$, consistent with the scaling applied to the original matrix $\mathbf{X}$. Ideally, this synthetic variable $\mathbf{X}_{\text{inert}}$ should exhibit no linear association with any of the existing variables in $\mathbf{X}$, that is, it should be orthogonal to all columns in $\mathbf{X}$. However, due to practical constraints, achieving perfect orthogonality is typically not feasible, so we aim to add an inert column that minimizes correlation with the existing design columns.

It is important to note that the augmented variable is deliberately constructed to have no effect on the response. The analysis is conducted as though there are $p+1$ input variables, but the added variable is known a priori to be inert. As a result, the posterior distribution of $\theta_w[0]$ reflects the behavior of a lengthscale parameter associated with an inactive factor. Because the objective of variable selection is to identify which covariates exert a meaningful influence on the response, distinguishable from random variation, the posterior distributions of the actual experimental variables can be compared to those of the inert variable. In essence, the posterior of the inert variable serves as a baseline or reference distribution, analogous to the role of a null distribution in frequentist hypoth-

esis testing, for evaluating the significance of the $\theta_w[k]$ values linked to the true experimental factors.

A central advantage of the RDVS approach is that it eliminates the need to define a threshold for determining when a particular $\theta_w[k]$ is "sufficiently small", that is, the experimenter does not need to subjectively decide how far from one a value must be to indicate significance. This is particularly beneficial because what constitutes a "small" value of $\theta_w[k]$ can vary depending on the specific data set. Instead, the only decision required is whether the posterior distributions of the experimental variables are distinguishable from that of the known inert variable. The ideal scenario would involve selecting $\mathbf{X}_{\text{inert}}$ so that it is orthogonal to all columns in $\mathbf{X}$, thereby ensuring that the posterior distribution of $\theta_w[0]$ is unaffected by the specific configurations of the actual experimental factors. However, achieving perfect orthogonality is generally impractical. To overcome this, $\mathbf{X}_{\text{inert}}$ is randomly drawn from the same design space as $\mathbf{X}$, and the DGP model is fit accordingly, and this procedure is repeated multiple times. The posterior distributions obtained for each realization of the inert variable are then pooled to construct a composite reference distribution representing a typical null variable. This effectively averages across different instances of $\mathbf{X}_{\text{inert}}$, smoothing out the influence of any single realization.

There are several potential methods for comparing the posterior distributions of the experimental factors to those of the inert (null) variable. One practical option involves utilizing the full set of MCMC samples, comparing the values of the experimental and null variables at each iteration of the sampling process. But the MCMC approach is known to be computationally expensive. Instead, consider the following variational inference (VI) approach, which has gained popularity in recent years due to its computational efficiency (see, Blei et al., 2017; Hensman et al., 2013; Hoffman et al., 2013). In the VI framework, the goal is to approximate the true posterior distribution $p(\theta \mid y)$ of a latent variable θ, given observed data y, by selecting a tractable family of distributions $q(\theta)$. The optimal variational distribution $q^*(\theta)$ is obtained by minimizing the Kullback-Leibler (KL) divergence between $q(\theta)$ and the true posterior. This converts the inference problem into an optimization problem, where the objective is to solve:

$$q^*(\theta) = \arg\min_q \ \mathrm{KL}(q(\theta) \| p(\theta \mid y)).$$

This variational objective enables efficient posterior approximation and parameter estimation, provided the joint distribution $p(\theta, y)$ is specified. For GP

models in particular, (Hensman et al., 2013, 2015) offer detailed discussions on implementing VI effectively.

Let $\widehat{\boldsymbol{\theta_w[j]}}^2 = (\widehat{\theta_w[j]}_1^2, \ldots, \widehat{\theta_w[j]}_m^2, \ldots, \widehat{\theta_w[j]}_M^2)^T$, where $\widehat{\theta_w[j]}_m^2$ denotes the maximum a posteriori (MAP) estimate of the inverse length-scale for the input variable $\mathbf{x_j}$ obtained from the m^{th} iteration of the VI algorithm, for $j = 0, 1, \ldots, p$. For each input j, we assess its importance by comparing the distribution of $\widehat{\boldsymbol{\theta_w[j]}}^2$ to that of the inert input, $\widehat{\boldsymbol{\theta_w[0]}}^2$. More specifically, we use the q^{th} percentile of the null distribution, denoted by α_q, as a decision threshold. If the median of $\widehat{\boldsymbol{\theta_w[j]}}^2$ is less than α_q, the corresponding feature $\mathbf{x_j}$ is deemed inactive and may be excluded from the model.

Variable Selection with RDVS for DGPs

1. Standardize the design matrix $\mathbf{X}$.
2. **For** $m = 1, \ldots, M$:
    - Sample a random nuisance column $\mathbf{x}_0$ from the design space.
    - Form the augmented matrix $\mathbf{X}^* = (\mathbf{x}_0, \mathbf{X})$.
    - Fit the DGP model using variational inference (VI).
    - Record the MAP estimates of inverse length-scales for all columns in $\mathbf{X}^*$.
3. Collect MAP estimates as $\mathbf{L} = (\widehat{\boldsymbol{\theta_w[0]}}^2, \widehat{\boldsymbol{\theta_w[1]}}^2, \ldots, \widehat{\boldsymbol{\theta_w[p]}}^2)$, where each $\widehat{\boldsymbol{\theta_w[j]}}^2 \in R^M$.
4. Compute α_q, the qth percentile of the values in $\widehat{\boldsymbol{\theta_w[0]}}^2$.
5. **For** $k = 1, \ldots, p$:
    - **If** median$(\widehat{\boldsymbol{\theta_w[k]}}^2) \geq \alpha_q$, classify feature $\mathbf{x_k}$ as **active**.
    - **Else**, classify feature $\mathbf{x_k}$ as **inactive**.
6. **Output:** Indices of features classified as active.

The choice of the threshold percentile q reflects the researcher's tolerance for false positives, that is, the likelihood of incorrectly classifying an inactive feature as active. Researchers may also choose to apply different threshold values for different features if desired. Selecting a higher value of q imposes a stricter criterion, thereby reducing the probability of falsely identifying inactive variables

as active. However, this also raises the risk of missing genuinely weak but active features. Empirically, the threshold can be intuitively interpreted as setting $(100 - q)\%$ as the upper bound on the acceptable false positive rate.

6.5 Discussion

In this chapter, we studied DGPs, which extend standard GP models by introducing a hierarchical structure with multiple hidden layers. These layers contain latent variables that are not directly observed but act as nonlinear transformations of the input space, allowing the model to better accommodate non-stationarity. We derived the posterior distribution for the DGP and introduced a modified formulation that utilizes an inverse-RBF kernel along with an exponential prior on the inverse length-scale parameters, inspired by LASSO, to encourage sparsity and enhance model interpretability. In addition, we adapted the reference distribution variable selection technique to the DGP context, yielding a novel framework for identifying influential variables within hierarchical GP models. This framework provides a flexible modeling approach while performing feature importance by identifying significant input variables in high-dimensional, non-stationary environments.

Chapter 7

Extension and Conclusion

This thesis presents significant methodological advancements in the optimal design of experiments, with a particular focus on crossover designs under GLMs and the development of surrogate models for complex computer experiments.

In the first part of the thesis, we studied optimal crossover designs for experiments with non-normal responses modeled using GLMs. While uniform designs are optimal for linear models, our results demonstrate that they are generally sub-optimal under GLMs. Through extensive numerical illustrations and simulations, we identified locally D-optimal designs that outperform uniform designs in terms of efficiency, and we showed that these optimal designs remain robust across different correlation structures.

Building on this foundation, we applied these methods to a real-world dataset from a work environment experiment, illustrating the practical value of optimal designs for Poisson, beta, and gamma responses. The results indicated that the optimal allocation of sequences depends heavily on the assumed parameter values and the choice of link functions, with non-uniform designs often yielding substantial improvements in efficiency. Importantly, these findings reinforce the limitations of conventional uniform designs in applied settings.

Next, we derived a general equivalence theorem tailored to crossover designs under GLMs. This development provides a theoretically grounded and computationally efficient method to verify the optimality of a proposed design. Two versions of the theorem were established, one targeting the variance of all parameters and the other focusing solely on treatment effects. Applications of the theorem to real-life examples demonstrated that the optimal designs derived from solving the system of equations aligned closely with those obtained

through constrained optimization. An interesting extension would be to use the Bayesian approach to avoid guessing the values of unknown parameters.

In the second part of the thesis, the focus shifted to surrogate modeling for complex computer experiments. We highlighted the importance of using space-filling and orthogonal designs in constructing effective surrogates and introduced the weighted MaxPro and orthogonal weighted MaxPro criteria for this purpose. A sequential variable selection algorithm was proposed to enhance model sparsity and interpretability. Simulations showed that the proposed methods yield better discovery rates and more stable estimates than existing one-shot designs. A natural extension of this framework would be to Gaussian Process models that accommodate both qualitative and quantitative factors, such as the EzGP (Xiao et al., 2021) and MaGP (Xiao et al., 2022) models. These models are capable of capturing complex input structures, including cases where the inputs may themselves be probability distributions and the response is non-normal. Extending the sequential design framework to such settings would be particularly valuable, enabling efficient learning in high-dimensional, heterogeneous design spaces. Such an extension holds significant promise for advancing the field of computer experiments.

Finally, we discuss surrogate modeling with Deep Gaussian Process (DGP) surrogates, which are capable of capturing nonstationary and highly complex input-output relationships more effectively than traditional Gaussian Processes. We propose a novel framework that integrates DGP models with a reference distribution variable selection algorithm, facilitating efficient dimension reduction and enhancing surrogate accuracy. These advances illustrate how modern statistical learning techniques can be combined with classical design of experiments principles to support and improve scientific discovery in complex modeling scenarios.

A promising direction for future work is the extension of the sequential variable selection framework developed in Chapter 5 to the Deep Gaussian Process (DGP) models introduced in Chapter 6. Although this thesis lays the groundwork for incorporating variable selection into DGP surrogates, a fully adaptive sequential design strategy, capable of updating both sampling locations and feature importance across DGP layers, remains an open area for development. Future research could focus on designing active learning algorithms that are tailored to the hierarchical and nonstationary nature of DGPs, accounting for both local and global predictive uncertainty.

Another important extension would be the development of sequential design strategies for calibration using Deep Gaussian Processes (DGPs). DGPs

have demonstrated strong performance in calibration settings (Marmin & Filippone, 2022), offering the flexibility to model nonstationary and hierarchical relationships that are common in complex simulators. At the same time, sequential design approaches for calibration using stationary Gaussian Processes are only beginning to emerge (Koermer et al., 2023). Integrating these two developments, nonstationary DGP modeling with adaptive calibration strategies, could yield a powerful framework for efficient and accurate model calibration, particularly in high-dimensional or computationally intensive settings.

In summary, this thesis makes key contributions to both the theory and practice of experimental design. It expands the frontier of crossover design under non-normal settings, introduces an equivalence theorem for designs under GLMs, and proposes cutting-edge surrogate modeling strategies.

APPENDICES

A.1 Appendix A: Optimal Crossover Designs

Effect of Misspecification of Working Correlation Structures

Table 1: Optimal Design under Variance Misspecification

True Correlation Structure	Working Correlation Structure	Optimal proportions for θ_1				Optimal proportions for θ_2				Relative D-efficiency	
		$ABCD$	$BDAC$	$CADB$	$DCBA$	$ABCD$	$BDAC$	$CADB$	$DCBA$	under θ_1	under θ_2
$Corr(1)$	$Corr(2)$	0.1723	0.2483	0.2222	0.3572	0.2463	0.2493	0.2504	0.2540	0.9999	0.9999
	$Corr(3)$	0.1726	0.2483	0.2223	0.3568	0.2463	0.2493	0.2504	0.2540	0.9999	0.9999
	$Corr(4)$	0.1723	0.2513	0.2202	0.3562	0.2500	0.2500	0.2500	0.2500	0.9997	0.9988
	$Corr(5)$	0.2447	0.1713	0.2495	0.2223	0.3569	0.2475	0.2557	0.2521	0.9994	0.9995
	$Corr(6)$	0.2500	0.1724	0.2508	0.2197	0.3571	0.2500	0.2500	0.2500	0.9999	0.9984
$Corr(2)$	$Corr(1)$	0.1745	0.2489	0.2183	0.3583	0.2462	0.2493	0.2500	0.2545	0.9999	0.9999
	$Corr(3)$	0.1744	0.2489	0.2182	0.3585	0.2462	0.2493	0.2500	0.2545	0.9999	0.9999
	$Corr(4)$	0.1745	0.2514	0.2177	0.3564	0.2500	0.2500	0.2500	0.2500	0.9998	0.9987
	$Corr(5)$	0.1740	0.2503	0.2180	0.3577	0.2450	0.2480	0.2530	0.2540	0.9997	0.9997
	$Corr(6)$	0.1744	0.2512	0.2174	0.3570	0.2463	0.2497	0.2505	0.2535	0.9999	0.9985
$Corr(3)$	$Corr(1)$	0.1714	0.2480	0.2236	0.3570	0.2461	0.2492	0.2507	0.2540	0.9999	0.9999
	$Corr(2)$	0.1711	0.2480	0.2235	0.3574	0.2462	0.2492	0.2506	0.2540	0.9999	0.9999
	$Corr(4)$	0.1713	0.2516	0.2209	0.3562	0.2500	0.2500	0.2500	0.2500	0.9996	0.9987
	$Corr(5)$	0.1700	0.2463	0.2235	0.3572	0.2441	0.2476	0.2561	0.2522	0.9992	0.9995
	$Corr(6)$	0.1713	0.2510	0.2204	0.3573	0.2500	0.2500	0.2500	0.2500	0.9999	0.9984
$Corr(4)$	$Corr(1)$	0.1783	0.2585	0.2140	0.3492	0.2500	0.2637	0.2347	0.2516	0.9994	0.9987
	$Corr(2)$	0.1784	0.2580	0.2156	0.3480	0.2486	0.2640	0.2344	0.2530	0.9996	0.9987
	$Corr(3)$	0.1782	0.2592	0.2131	0.3495	0.2498	0.2643	0.2342	0.2517	0.9992	0.9986
	$Corr(5)$	0.1778	0.2579	0.2167	0.3476	0.2470	0.2650	0.2343	0.2537	0.9992	0.9993
	$Corr(6)$	0.1790	0.2555	0.2165	0.3490	0.2485	0.2631	0.2337	0.2547	0.9999	0.9999
$Corr(5)$	$Corr(1)$	0.1774	0.2477	0.2092	0.3657	0.2466	0.2501	0.2486	0.2547	0.9994	0.9999
	$Corr(2)$	0.1776	0.2476	0.2099	0.3649	0.2470	0.2506	0.2470	0.2554	0.9997	0.9999
	$Corr(3)$	0.1770	0.2477	0.2087	0.3666	0.2462	0.2503	0.2485	0.2550	0.9992	0.9999
	$Corr(4)$	0.1776	0.2492	0.2108	0.3624	0.2472	0.2538	0.2450	0.2540	0.9996	0.9994
	$Corr(6)$	0.1774	0.2496	0.2110	0.3620	0.2465	0.2535	0.2456	0.2544	0.9998	0.9991
$Corr(6)$	$Corr(1)$	0.1748	0.2553	0.2142	0.3557	0.2482	0.2652	0.2332	0.2534	0.9997	0.9985
	$Corr(2)$	0.1748	0.2551	0.2160	0.3541	0.2470	0.2657	0.2329	0.2544	0.9999	0.9985
	$Corr(3)$	0.1748	0.2558	0.2133	0.3561	0.2482	0.2660	0.2325	0.2533	0.9996	0.9984
	$Corr(4)$	0.1754	0.2530	0.2172	0.3544	0.2476	0.2652	0.2324	0.2548	0.9999	0.9999
	$Corr(5)$	0.1741	0.2556	0.2180	0.3523	0.2452	0.2669	0.2339	0.2540	0.9994	0.9991

Optimal Design for Latin Square Example with 24 Sequences.

The following tables represent optimal designs for the Latin square example with 24 sequences under θ_1 and θ_2.

Table 2: Optimal design considering 24 sequences under θ_1

Treatment Sequence	Optimal Designs for θ_1					
	$\rho = 0.1$	$\rho = 0.2$	$\rho = 0.5$	$\rho = 0.6$	$\rho = 0.7$	$\rho = 0.9$
$ABCD$	0.0094	0.0071	0.0109	0.0119	0.0125	0.0122
$ABDC$						
$ACBD$	0.0716	0.1037	0.1148	0.1156	0.1153	0.1115
$ADBC$	0.1096	0.0820	0.0753	0.0795	0.0859	0.1003
$ACDB$						
$ADCB$						
$BACD$	0.0513	0.0537	0.0459	0.0417	0.0362	0.0250
$BADC$						
$CABD$	0.1254	0.1162	0.1042	0.1007	0.0972	0.0878
$DABC$	0.0200	0.0447	0.0469	0.0421	0.0356	0.0194
$CADB$						
$DACB$	0.0122					
$BCAD$						
$BDAC$	0.1735	0.1993	0.2055	0.2045	0.2031	0.2019
$CBAD$						
$DBAC$	0.1667	0.1404	0.1374	0.1461	0.1588	0.1924
$CDAB$	0.1265	0.1426	0.1483	0.1473	0.1448	0.1358
$DCAB$	0.1114	0.1082	0.1108	0.1107	0.1106	0.1120
$BCDA$						
$BDCA$	0.0224	0.0003				
$CBDA$						
$DBCA$						
$CDBA$						
$DCBA$						

Table 3: Optimal design considering 24 sequences under θ_2.

Treatment Sequence	Optimal Designs for θ_2					
	$\rho = 0.1$	$\rho = 0.2$	$\rho = 0.5$	$\rho = 0.6$	$\rho = 0.7$	$\rho = 0.9$
$ABCD$	0.1105	0.1107	0.0875	0.0870	0.0876	0.0846
$ABDC$						0.0112
$ACBD$	0.0488	0.0525	0.0615	0.0624	0.0625	0.0522
$ADBC$	0.0347	0.0329	0.0516	0.0561	0.0618	0.0807
$ACDB$	0.0402	0.0348	0.0126	0.0128	0.0135	0.0247
$ADCB$	0.0370	0.0417	0.0645	0.0625	0.0587	0.0383
$BACD$						0.0052
$BADC$	0.1125	0.1109	0.0903	0.0855	0.0801	0.0545
$CABD$	0.0467	0.0419	0.0127	0.0087	0.0054	0.0125
$DABC$		0.0041	0.0213	0.0192	0.0152	
$CADB$	0.0611	0.0619	0.0729	0.0733	0.0737	0.0674
$DACB$			0.0136	0.0198	0.0272	0.0537
$BCAD$	0.0363	0.0371	0.0472	0.0441	0.0392	0.0141
$BDAC$	0.0034		0.0003	0.0008	0.0027	0.0224
$CBAD$		0.0004	0.0360	0.0427	0.0503	0.0744
$DBAC$	0.1034	0.1056	0.0854	0.0859	0.0858	0.0728
$CDAB$						0.0055
$DCAB$	0.1157	0.1163	0.0946	0.0915	0.0888	0.0780
$BCDA$			0.0241	0.0294	0.0361	0.0617
$BDCA$	0.0882	0.0901	0.0719	0.0728	0.0733	0.0678
$CBDA$	0.0239	0.0297	0.0369	0.0356	0.0326	0.0166
$DBCA$	0.0276	0.0201	0.0238	0.0192	0.0153	0.0109
$CDBA$	0.1100	0.1093	0.0913	0.0907	0.0902	0.0802
$DCBA$						0.0106

Explicit Expression of the Objective Function

We give an example of a Latin square design to illustrate how we obtain the objective function. We use $Corr(1)$ correlation structure with $\rho = 0.2$.

First, we look at the design matrix for each of the subjects. The design matrix is obtained by using the expression of X_j mentioned in Section 1.2.1.

$$X_1 = X_{ABCD} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix},$$

$$X_2 = X_{CADB} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix},$$

$$X_3 = X_{BDAC} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$X_4 = X_{DCBA} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}.$$

Now, using the above design matrix for each subject and estimates of parameter values, we consider $\hat{\theta} = [0.5, 0.06, -0.53, -0.6, -0.35, 0.025, -0.23, 0.73, 0.23]$. Then the values of $\eta_j = X_j\hat{\theta}$ for each subject can be obtained as follows:

$$\eta_1 = X_1\hat{\theta} = \begin{pmatrix} 0.534 \\ 0.278 \\ 0.761 \\ -0.070 \end{pmatrix}, \qquad \eta_2 = X_2\hat{\theta} = \begin{pmatrix} 0.185 \\ 1.131 \\ 0.307 \\ -0.050 \end{pmatrix},$$

$$\eta_3 = X_3\hat{\theta} = \begin{pmatrix} 0.557 \\ 0.857 \\ -0.220 \\ -0.122 \end{pmatrix}, \qquad \eta_4 = X_4\hat{\theta} = \begin{pmatrix} 0.307 \\ 0.950 \\ -0.112 \\ 0.658 \end{pmatrix}.$$

Hence, using model (1.2) mentioned in Section 1.2.1, we can get corresponding $\mu_j = \frac{\exp\{\eta_j\}}{1+\exp\{\eta_j\}}$, they are as follows:

$$\mu_1 = \frac{\exp\{\eta_1\}}{1+\exp\{\eta_1\}} = \begin{pmatrix} 0.6304156 \\ 0.5690558 \\ 0.6815708 \\ 0.4825071 \end{pmatrix}, \qquad \mu_2 = \frac{\exp\{\eta_2\}}{1+\exp\{\eta_2\}} = \begin{pmatrix} 0.5461185 \\ 0.7560234 \\ 0.5761528 \\ 0.4875026 \end{pmatrix},$$

$$\mu_3 = \frac{\exp\{\eta_3\}}{1+\exp\{\eta_3\}} = \begin{pmatrix} 0.6357581 \\ 0.7020335 \\ 0.4452208 \\ 0.4695378 \end{pmatrix}, \qquad \mu_4 = \frac{\exp\{\eta_4\}}{1+\exp\{\eta_4\}} = \begin{pmatrix} 0.5761528 \\ 0.7211152 \\ 0.4720292 \\ 0.6588110 \end{pmatrix}.$$

We are using compound symmetric correlation structure $Corr(1)$ with $\rho = 0.2$. Hence we have $C(\alpha) = Corr(1)$ as true correlation matrix.

Correlation matrix $C(\alpha)$ and matrix H can be written down as follows:

$$C(\alpha) = \begin{pmatrix} 1 & 0.2 & 0.2 & 0.2 \\ 0.2 & 1 & 0.2 & 0.2 \\ 0.2 & 0.2 & 1 & 0.2 \\ 0.2 & 0.2 & 0.2 & 1 \end{pmatrix}, \quad H = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Using the expression for $Cov[Y_j]$ mentioned below we compute covariance matrix for each subject. We denote this covariance matrix by W_j for each subject j:

$$Cov[Y_j] = W_j = D_j^{1/2} Corr_1 D_j^{1/2}$$

where D_j in above equation is $diag\Big(\mu_{1j}(1-\mu_{1j}), \ldots, \mu_{pj}(1-\mu_{pj})\Big)$ and p is number of periods.

Hence corresponding D_j for Latin square example are as follows:

$$D_1 = \begin{pmatrix} 0.23 & 0 & 0 & 0 \\ 0 & 0.24 & 0 & 0 \\ 0 & 0 & 0.22 & 0 \\ 0 & 0 & 0 & 0.25 \end{pmatrix}, \quad D_2 = \begin{pmatrix} 0.25 & 0 & 0 & 0 \\ 0 & 0.18 & 0 & 0 \\ 0 & 0 & 0.24 & 0 \\ 0 & 0 & 0 & 0.25 \end{pmatrix},$$

$$D_3 = \begin{pmatrix} 0.23 & 0 & 0 & 0 \\ 0 & 0.21 & 0 & 0 \\ 0 & 0 & 0.25 & 0 \\ 0 & 0 & 0 & 0.25 \end{pmatrix}, \quad D_4 = \begin{pmatrix} 0.24 & 0 & 0 & 0 \\ 0 & 0.20 & 0 & 0 \\ 0 & 0 & 0.25 & 0 \\ 0 & 0 & 0 & 0.22 \end{pmatrix}.$$

Calculating matrix $D_j^{1/2}$ and using the above formula for W_j , we have inverse of W_j matrices as follows:

$$W_1^{-1} = \begin{pmatrix} 4.69 & -0.65 & -0.69 & -0.65 \\ -0.65 & 4.46 & -0.68 & -0.63 \\ -0.69 & -0.68 & 5.04 & -0.67 \\ -0.65 & -0.63 & -0.67 & 4.38 \end{pmatrix},$$

$$W_2^{-1} = \begin{pmatrix} 4.41 & -0.73 & -0.64 & -0.63 \\ -0.73 & 5.93 & -0.74 & -0.73 \\ -0.64 & -0.74 & 4.48 & -0.63 \\ -0.63 & -0.73 & -0.63 & 4.38 \end{pmatrix},$$

$$W_3^{-1} = \begin{pmatrix} 4.72 & -0.71 & -0.65 & -0.65 \\ -0.71 & 5.23 & -0.69 & -0.68 \\ -0.65 & -0.69 & 4.43 & -0.63 \\ -0.65 & -0.68 & -0.63 & 4.39 \end{pmatrix},$$

$$W_4^{-1} = \begin{pmatrix} 4.48 & -0.70 & -0.63 & -0.67 \\ -0.70 & 5.44 & -0.70 & -0.73 \\ -0.63 & -0.70 & 4.39 & -0.66 \\ -0.67 & -0.73 & -0.66 & 4.87 \end{pmatrix}.$$

Note that $D_\omega = D_j$ and $W_\omega = W_j$.

The variance of parameter estimate $\text{Var}(\hat{\theta}) = \left[\sum_{\omega \epsilon \Omega} np_\omega \frac{\partial \mu'_\omega}{\partial \theta} W_\omega^{-1} \frac{\partial \mu_\omega}{\partial \theta}\right]^{-1}$ has another component which is $\frac{\partial \mu_\omega}{\partial \theta}$ and the ith row of $\frac{\partial \mu_\omega}{\partial \theta}$ is $x_i' d_i$, where x_i is the ith row of design matrix X_ω and d_i corresponds to ith diagonal entry of matrix D_j.

Hence, $\frac{\partial \mu_\omega}{\partial \theta}$ matrix for each subject are as follows:

$$\frac{\partial \mu_1}{\partial \theta} = \begin{pmatrix} 0.23 & 0.23 & 0 & 0 & 0 & 0.23 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.24 & 0 & 0.24 & 0 & 0 & 0 & 0.24 & 0 & 0 & 0.24 & 0 & 0 & 0 \\ 0.22 & 0 & 0 & 0.22 & 0 & 0 & 0 & 0.22 & 0 & 0 & 0.22 & 0 & 0 \\ 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0.25 & 0 \end{pmatrix},$$

$$\frac{\partial \mu_2}{\partial \theta} = \begin{pmatrix} 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.18 & 0 & 0.18 & 0 & 0 & 0 & 0 & 0 & 0.18 & 0 & 0.18 & 0 & 0 \\ 0.24 & 0 & 0 & 0.24 & 0 & 0.24 & 0 & 0 & 0 & 0 & 0 & 0 & 0.24 \\ 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0.25 & 0 & 0.25 & 0 & 0 & 0 \end{pmatrix},$$

$$\frac{\partial \mu_3}{\partial \theta} = \begin{pmatrix} 0.23 & 0.23 & 0 & 0 & 0 & 0 & 0 & 0.23 & 0 & 0 & 0 & 0 & 0 \\ 0.21 & 0 & 0.21 & 0 & 0 & 0.21 & 0 & 0 & 0 & 0 & 0 & 0.21 & 0 \\ 0.25 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 \\ 0.25 & 0 & 0 & 0 & 0.25 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0.25 \end{pmatrix},$$

$$\frac{\partial \mu_4}{\partial \theta} = \begin{pmatrix} 0.24 & 0.24 & 0 & 0 & 0 & 0 & 0 & 0 & 0.24 & 0 & 0 & 0 & 0 \\ 0.20 & 0 & 0.20 & 0 & 0 & 0 & 0 & 0.20 & 0 & 0 & 0 & 0 & 0.20 \\ 0.25 & 0 & 0 & 0.25 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0.25 & 0 \\ 0.22 & 0 & 0 & 0 & 0.22 & 0.22 & 0 & 0 & 0 & 0 & 0.22 & 0 & 0 \end{pmatrix}.$$

Using above calculated inverse of each W_ω matrix, and the corresponding calculated $\frac{\partial \mu_\omega}{\partial \theta}$ matrices we can calculate required 13×13 matrices $\frac{\partial \mu'_\omega}{\partial \theta} W_\omega^{-1} \frac{\partial \mu_\omega}{\partial \theta}$ for each ω.

Further, inverse of $\left[\sum_{\omega \epsilon \Omega} n p_\omega \frac{\partial \mu'_\omega}{\partial \theta} W_\omega^{-1} \frac{\partial \mu_\omega}{\partial \theta}\right]$ i.e $\mathrm{Var}(\hat{\theta})$ is found numerically and calculate objective function $\mathrm{Var}(\hat{\tau}) = H \mathrm{Var}(\hat{\theta}) H'$, where we try to minimize $\mathrm{Var}(\hat{\tau})$ w.r.t p_ω. These values of p_ω, which minimizes the objective function, are the optimal proportions we are looking for.

A.2 Appendix B: A General Equivalence Theorem for Crossover Designs

Alternative Proof of Lemma 1:

We use the first-order condition of a convex function stated in equation (3.2) of (Boyd & Vandenberghe, 2004), *A differentiable function f defined on a convex domain is convex if and only if $f(\boldsymbol{x}) \geq f(\boldsymbol{y}) + \nabla f(\boldsymbol{y})^T(\boldsymbol{x} - \boldsymbol{y})$ hold for all $\boldsymbol{x}, \boldsymbol{y}$ in the domain.*

Let $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)'$, $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)' \in \Re^n_{>0}$.

Then, we want to show

$$f(\boldsymbol{x}) \geq f(\boldsymbol{y}) + \nabla f(\boldsymbol{y})^T(\boldsymbol{x} - \boldsymbol{y}),$$

$$\text{i.e., to show,} \quad \frac{1}{\prod_{i=1}^n x_i} - \frac{1}{\prod_{i=1}^n y_i} \geq \begin{bmatrix} \frac{-1}{y_1^2 y_2 \ldots y_n} \cdots \frac{-1}{y_1 \ldots y_{n-1} y_n^2} \end{bmatrix} \begin{bmatrix} x_1 - y_1 \\ x_2 - y_2 \\ \ldots \\ x_n - y_n \end{bmatrix},$$

$$\text{i.e., to show,} \quad \frac{1}{\prod_{i=1}^n x_i} - \frac{1}{\prod_{i=1}^n y_i} \geq \frac{y_1 - x_1}{y_1^2 y_2 \ldots y_n} + \cdots + \frac{y_n - x_n}{y_1 \ldots y_{n-1} y_n^2},$$

$$\text{i.e., to show,} \quad \frac{1}{\prod_{i=1}^n x_i} - \frac{1}{\prod_{i=1}^n y_i} \geq \frac{n}{\prod_{i=1}^n y_i} - \sum_{i=1}^n \frac{x_i}{y_1 \ldots y_i^2 \ldots y_n},$$

$$\text{i.e., to show,} \quad \frac{1}{\prod_{i=1}^n x_i} + \sum_{i=1}^n \frac{x_i}{y_1 \ldots y_i^2 \ldots y_n} \geq \frac{n+1}{\prod_{i=1}^n y_i}.$$

Since $\boldsymbol{x}, \boldsymbol{y} > 0$, the LHS is the mean of $(n+1)$ positive terms. By applying AM $\geq$ GM inequality, the result follows.

Bibliography

Atkinson, A., Donev, A., & Tobias, R. (2007). *Optimum experimental designs, with sas*.

Ba, S., & Joseph, V. R. (2011). Multi-layer designs for computer experiments. *Journal of the American Statistical Association*, *106*, 1139–1149.

Bayarri, M. J., Berger, J. O., Kennedy, M. C., Kottas, A., Paulo, R., Sacks, J., Cafeo, J. A., Lin, C.-H., & Tu, J. (2009). Predicting vehicle crashworthiness: Validation of computer models for functional and hierarchical data. *Journal of the American Statistical Association*, *104*(487), 929–943.

Blei, D. M., Kucukelbir, A., & McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, *112*(518), 859–877. https://doi.org/10.1080/01621459.2017.1285773

Bornemann, F. (2010). On the numerical evaluation of fredholm determinants. *Mathematics of Computation*, *79*(270), 871–915.

Bose, M., & Dey, A. (2009). *Optimal crossover designs*. World Scientific.

Box, G. (1980). Sampling and Bayes' inference in scientific modelling and robustness (with discussion). *Journal of the Royal Statistical Society, Series A*, *143*.

Box, G., Hunter, J., & Hunter, W. (2005). *Statistics for experimenters: Design, innovation, and discovery*. Wiley.

Boyd, S. P., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Butler, N. A. (2001). Optimal and orthogonal Latin hypercube designs for computer experiments. *Biometrika*, *88*(3), 847–857.

Carriere, K. C., & Huang, R. (2000). Crossover designs for two-treatment clinical trials. *J. Statist. Plann. Inference*, *87*, 125–134.

Cheng, C. S., & Wu, C. F. (1980). Balanced repeated measurements designs. *Ann. Statist.*, *8*, 1272–1283.

Chernoff, H. (1953). Locally optimal designs for estimating parameters. *Ann. Math. Statist.*, *24*, 586–602.

Chipman, H. A., George, E. I., & McCulloch, R. E. (2001). The practical implementation of bayesian model selection. In *Model selection* (pp. 65–

134, Vol. 38). Institute of Mathematical Statistics. https://doi.org/10.1214/lnms/1215540960

Cioppa, T. M., & Lucas, T. W. (2007). Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics*, *49*(1), 45–55.

Cox, D., & Reid, N. (2000). *The theory of the design of experiments*. Taylor & Francis. https://books.google.com/books?id=bPijngEACAAJ

Cui, J., & Krems, R. V. (2015). Gaussian process model for collision dynamics of complex molecules. *Physical Review Letters*, *115*(7). https://doi.org/10.1103/physrevlett.115.073202

Damianou, A., & Lawrence, N. D. (2013). Deep gaussian processes. *Artificial Intelligence and Statistics*, 207–215.

Deisenroth, M., Fox, D., & Rasmussen, C. (2015). Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*, 408–423.

Dette, H., & Pepelyshev, A. (2010). Generalized latin hypercube design for computer experiments. *Technometrics*, *52*, 421–429.

Dey, A., Gupta, V. K., & Singh, M. (1983). Optimal change-over designs. *Sankhya B45*, 233–239.

Doob, J., & Doob, J. (1953). *Stochastic processes*. Wiley.

Dunlop, M. M., Girolami, M. A., Stuart, A. M., & Teckentrup, A. L. (2018). How deep are deep gaussian processes? *Journal of Machine Learning Research*, *19*(54), 1–46.

Ek, C. H., Torr, P. H. S., & Lawrence, N. D. (2008). Gaussian process latent variable models for human pose estimation. In A. Popescu-Belis, S. Renals, & H. Bourlard (Eds.), *Machine learning for multimodal interaction* (pp. 132–143). Springer Berlin Heidelberg.

Fang, K.-T. (1980). The uniform design: Application of number-theoretic methods in experimental design. *Acta Mathematica Applicate Sinica*, *3*, 363–372.

Fang, K.-T., Li, R., & Sudijanto, A. (2006). *Design and modeling for computer experiments*. Chapman Hall.

Fedorov, V. V. (1971). The design of experiments in the multiresponse case. *Theory Probab. Appl.*, *16*, 323–332.

Fedorov, V. V. (1972). *Theory of optimal experiment*. Academic Press.

Fedorov, V. V., & Leonov, S. L. (2014). *Optimal design for nonlinear response models*. Chapman & Hall/CRC.

Fedorov, V. V., & Malyutov, M. B. (1972). Optimal designs in regression problems. *Math. Operat. Statist.*, *3*, 281–308.

Finney, D. (1945). The fractional replication of factorial arrangements. *Annals of Eugenics*, *12*.

Fisher, R. (1925). *Statistical methods for research workers*. Edinburgh Oliver & Boyd.

Georgiou, S. D. (2009). Orthogonal Latin hypercube designs from generalized orthogonal designs. *Journal of Statistical Planning and Inference*, *139*(4), 1530–1540.

Georgiou, S. D., & Efthimiou, I. (2014). Some classes of orthogonal Latin hypercube designs. *Statistica Sinica*, *24*(1), 101–120.

Gramacy, R. B. (2020). *Surrogates: Gaussian process modeling, design and optimization for the applied sciences*. Chapman Hall/CRC.

Gramacy, R. B., & Apley, D. W. (2015). Local gaussian process approximation for large computer experiments. *Journal of Computational and Graphical Statistics*, *24*(2), 561–578. https://doi.org/10.1080/10618600.2014.914442

Hedayat, A., & Afsarinejad, K. (1975). Repeated measurements designs, i. In J. N. Srivastava (Ed.), *A survey of statistical designs and linear models* (pp. 229–242). North-Holland.

Hensman, J., Fusi, N., & Lawrence, N. D. (2013). Gaussian processes for big data. *arXiv preprint arXiv:1309.6835*. https://arxiv.org/abs/1309.6835

Hensman, J., Matthews, A., & Ghahramani, Z. (2015). Scalable variational gaussian process classification. *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS)*, *38*, 351–360. https://proceedings.mlr.press/v38/hensman15.html

Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, *12*(1), 55–67.

Hoffman, M. D., Blei, D. M., Wang, C., & Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, *14*, 1303–1347. http://jmlr.org/papers/v14/hoffman13a.html

Jankar, J., Mandal, A., & Yang, J. (2020). Optimal crossover designs for generalized linear models. *Journal of Statistical Theory and Practice*, *14*, 23.

Jin, R., Chen, W., & Sudjianto, A. (2005). An efficient algorithm for constructing optimal design of computer experiments. *Journal of Statistical Planning and Inference*, *134*, 268–287.

Johnson, M. E., Moore, L. M., & Ylvisaker, D. (1990). Minimax and maximin distance designs. *Journal of Statistical Planning and Inference*, *26*, 131–148.

Jones, B., Silvestrini, R. T., Montgomery, D. C., & Steinberg, D. M. (2015). Bridge designs for modeling systems with low noise. *Technometrics*, *57*, 155–163.

Joseph, V. R., Dasgupta, T., Tuo, R., & Wu, C. F. J. (2015). Sequential exploration of complex surfaces using minimum energy designs. *Technometrics*, *57*, 64–74.

Joseph, V. R., Gul, E., & Ba, S. (2015). Maximum projection designs for computer experiments. *Biometrika*, *102*, 371–380.

Joseph, V. R., & Hung, Y. (2008). Orthogonal-maximin latin hypercube designs. *Statistica Sinica*, *18*, 171–186.

Joseph, V. R., & Melkote, S. N. (2009). Statistical adjustments to engineering models. *Journal of Quality Technology*, *41*, 362–375.

Kennedy, M. C., & O'Hagan, A. (2001). Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B*, *63*, 425–464.

Kenward, M. G., & Jones, B. (1992). Alternative approaches to the analysis of binary and categorical repeated measurements. *Journal of Biopharmaceutical Statistics*, *2*, 137–170.

Kenward, M. G., & Jones, B. (2014). *Design and analysis of cross-over trials* (3rd). Chapman; Hall.

Kershner, R. P., & Federer, W. T. (1981). Two-treatment crossover designs for estimating a variety of effects. *J. Amer. Statist. Assoc.*, *76*, 612–619.

Khuri, A. I., Mukherjee, B., Sinha, B. K., & Ghosh, M. (2006). Design issues for generalized linear models: A review. *Statistical Science*, *21*, 376–399.

Kidder, L. E., Scheel, M. A., Teukolsky, S. A., Carlson, E. D., & Cook, G. B. (2000). Black hole evolution by spectral methods. *Phys. Rev. D*, *62*, 084032. https://doi.org/10.1103/PhysRevD.62.084032

Kiefer, J., & Wolfowitz, J. (1960). The equivalence of two extremum problems. *Canadian Journal of Mathematics*, *12*, 363–366.

Koermer, S., Loda, J., Noble, A., & Gramacy, R. B. (2023). Active learning for simulator calibration. *arXiv preprint arXiv:2301.10228*. https://arxiv.org/abs/2301.10228

Kruckow, M., Tauris, T., Langer, N., Kramer, M., & Izzard, R. (2018). Progenitors of gravitational wave mergers: Binary evolution with the stellar grid-based code combine. *Monthly Notices of the Royal Astronomical Society*, *481*, 1908–1949.

Kunert, J. (1983). Optimal design and refinement of the linear model with applications to repeated measurements designs. *Ann. Statist.*, *11*, 247–257.

Kunert, J. (1984). Optimality of balanced uniform repeated measurements designs. *Ann. Statist.*, *12*, 1006–1017.

Kushner, H. B. (1997). Optimal repeated measurements designs: The linear optimality equations. *Ann. Statist.*, *25*, 2328–2344.

Laska, E., & Meisner, M. (1985). A variational approach to optimal two-treatment crossover designs: Application to carryover effect models. *J. Amer. Statist. Assoc.*, *80*, 704–710.

Layard, M. W., & Arvesen, J. N. (1978). Analysis of poisson data in crossover experimental designs. *Biometrics*, *34*, 421–428.

Leary, S., Bhaskar, A., & Keane, A. (2003). Optimal orthogonal-array-based Latin hypercubes. *Journal of Applied Statistics*, *30*(5), 585–598.

Li, W., & Wu, C. F. J. (1997). Columnwise-pairwise construction of supersaturated designs. *Statistica Sinica*, *7*, 639–652.

Liang, K. Y., & Zeger, S. L. (1986). Longitudinal data analysis using generalized linear models. *Biometrika*, *73*, 13–22.

Liang, K. Y., Zeger, S. L., & Albert, P. S. (1988). Models for longitudinal data: A generalized estimating equation approach. *Biometrics*, *44*, 1049–1060.

Lin, C. D., Mukerjee, R., & Tang, B. (2009). Construction of orthogonal and nearly orthogonal Latin hypercubes. *Biometrika*, *96*(1), 243–247.

Linkletter, C., Bingham, D., Hengartner, N., Higdon, D., & Ye, K. Q. (2006). Variable selection for gaussian process models in computer experiments. *Technometrics*, *48*(4), 478–490. https://doi.org/10.1198/004017006000000222

Marmin, S., & Filippone, M. (2022). Deep gaussian processes for calibration of computer models. *Bayesian Analysis*, 1–30. https://doi.org/10.1214/21-BA1293

Matthews, J. N. S. (1987). Recent developments in crossover designs. *Internat. Statist. Rev.*, *56*, 117–127.

McCullagh, P., & Nelder, J. A. (1989). *Generalized linear models* (2nd). Chapman; Hall.

McKay, M. D., Beckman, R. J., & Conover, W. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, *21*, 239–245.

Montgomery, D. (2009). *Introduction to statistical quality control*. Wiley.

Morris, M. D., & Mitchell, T. J. (1995a). Exploratory designs for computer experiments. *Journal of Statistical Planning and Inference*, *43*, 381–402.

Morris, M. D., & Mitchell, T. J. (1995b). Exploratory designs for computational experiments. *Journal of Statistical Planning and Inference*, *43*(3), 381–402.

Nocedal, J., & Wright, S. (2006). *Numerical optimization*. Springer New York.

Owen, A. (1994). Controlling correlations in latin hypercube samples. *Journal of the American Statistical Association*, *89*, 1517–1522.

Pamadi, B., Covell, P., Tartabini, P., & Murphy, K. (2004). Aerodynamic characteristics and glide-back performance of langley glide-back booster. *22nd Applied Aerodynamics Conference and Exhibit*, 5382.

Pitchforth, J., Nelson-White, E., van den Helder, M., & Oosting, W. (2020). The work environment pilot: An experiment to determine the optimal office design for a technology company. *PLOS ONE*, *15*(5). https://doi.org/10.1371/journal.pone.0232949

Prentice, R. L. (1988). Correlated binary regression with covariates specific to each binary observation. *Biometrics*, *44*(4), 1033–1048.

Pukelsheim, F. (1993). *Optimal design of experiments*. Wiley.

Qian, P. Z. G., Tang, B., & Wu, C. F. J. (2009). Nested space-filling designs for experiments with two levels of accuracy. *Statistica Sinica*, *19*, 287–300.

Qian, P. Z. G., & Wu, C. F. J. (2009). Sliced space-filling designs. *Biometrika*, *96*, 945–956.

Radaideh, M. I., & Kozlowski, T. (2020). Surrogate modeling of advanced computer simulations using deep gaussian processes. *Reliability Engineering & System Safety*, *195*, 106731.

Rao, C. R. (1973). *Linear statistical inference and its applications*. John Wiley & Sons, Ltd.

Sacks, J., Welch, W. J., Mitchell, T. J., & Wynn, H. P. (1989). Design and analysis of computer experiments. *Statistical Science*, *4*, 409–423.

Salimbeni, H., & Deisenroth, M. (2017). Doubly stochastic variational inference for deep gaussian processes. *arXiv preprint arXiv:1705.08933*.

Santner, T. J., Williams, B. J., & Notz, W. I. (2003). *The design and analysis of computer experiments*. Springer.

Sauer, A., Gramacy, R. B., & Higdon, D. (2022). Active learning for deep gaussian process surrogates. *Technometrics*, *65*(1), 4–18. https://doi.org/10.1080/00401706.2021.2008505

Schmidt, A. M., & O'Hagan, A. (2003). Bayesian inference for non-stationary spatial covariance structure via spatial deformations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *65*(3), 743–758.

Senn, S. (2003). *Cross-over trials in clinical research* (2nd). Wiley.

Shewry, M. C., & Wynn, H. P. (1987). Maximum entropy sampling. *Journal of Applied Statistics*, *14*, 165–170.

Silvey, S. D. (1980). *Optimal design*.

Singh, S. P., & Mukhopadhyay, S. (2016). Bayesian crossover design for generalized linear models. *Computational Statistics and Data Analysis*, *104*, 35–50.

Snee, R. D. (1990). Statistical thinking and its contribution to total quality. *The American Statistician*, *44*(2).

Steinberg, D. M., & and, W. G. H. (1984). Experimental design: Review and comment. *Technometrics*, *26*.

Steinberg, D. M., & Lin, D. K. J. (2006). A construction method for orthogonal Latin hypercube designs. *Biometrika*, *93*(2), 279–288.

Stufken, J., & Yang, M. (2012). Optimal designs for generalized linear models. In K. Hinkelmann (Ed.), *Design and analysis of experiments, volume 3: Special designs and applications*. Wiley.

Sun, F., Liu, M.-Q., & Lin, D. K. J. (2009). Construction of orthogonal Latin hypercube designs. *Biometrika*, *96*(4), 971–974.

Sun, F., Liu, M.-Q., & Lin, D. K. J. (2010). Construction of orthogonal Latin hypercube designs with flexible run sizes. *Journal of Statistical Planning and Inference*, *140*(11), 3236–3242.

Sun, F., & Tang, B. (2017). A general rotation method for orthogonal Latin hypercubes. *Biometrika*, *104*(2), 465–472.

Tang, B. (1993). Orthogonal array-based latin hypercubes. *Journal of the American Statistical Association*, *88*, 1392–1397.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(1), 267–288.

Velásquez, R., & Lara, J. V. M. (2020). Forecast and evaluation of covid-19 spreading in usa with reduced-space gaussian process regression. *Chaos Solitons & Fractals*, *136*, 109924.

Wang, Y., Liu, S., & Xiao, Q. (2024). Construction of orthogonal-maxpro latin hypercube designs. *Journal of Quality Technology*, *56*(4), 342–354.

Whittle, P., & Malyutov, M. B. (1973). Some general points in the theory of optimal experimental design. *J. Roy. Statist. Soc. Ser.*, *35*, 123–130.

Williams, D., Heng, I., Gair, J., Clark, J., & Khamesra, B. (2019). A precessing numerical relativity waveform surrogate model for binary black holes: A gaussian process regression approach. *arXiv: General Relativity and Quantum Cosmology*. https://arxiv.org/abs/1903.09204

Williams, E. (1949). Experimental designs balanced for the estimation of residual effects of treatments. *Australian Journal of Chemistry*, *2*(2), 149–168.

Winker, P., & Fang, K.-T. (1998). Applications of threshold accepting to the evaluation of the quality of experimental designs. *Mathematical and Computer Modelling*, *27*(1-2), 1–15.

Withers, C. S., & Nadarajah, S. (2010). Expansion for functions of determinants of power series. *Canadian Applied Mathematics Quarterly*, *18*(1), 107–114.

Wu, C. F. J. (2015). Post-fisherian experimentation: From physical to virtual. *Journal of the American Statistical Association*, *110*, 612–620.

Wu, C. J., & Hamada, M. S. (2011). *Experiments: Planning, analysis, and optimization*. John Wiley & Sons.

Xiao, Q., Mandal, A., & Deng, X. (2022). Modeling and active learning for experiments with quantitative-sequence factors. *Journal of the American Statistical Association*, *119*(545), 407–421.

Xiao, Q., Mandal, A., Lin, C. D., & Deng, X. (2021). Ezgp: Easy-to-interpret gaussian process models for computer experiments with both quantitative and qualitative factors. *SIAM/ASA Journal on Uncertainty Quantification*, *9*(2), 333–353.

Yang, J., Mandal, A., & Majumdar, D. (2016). Optimal design for 2^k factorial experiments with binary response. *Statistica Sinica*, *26*(1), 385–411.

Yang, J., & Liu, M.-Q. (2012). Construction of orthogonal and nearly orthogonal Latin hypercube designs from orthogonal designs. *Statistica Sinica*, 433–442.

Ye, K. Q. (1998). Orthogonal column latin hypercubes and their application in computer experiments. *Journal of the American Statistical Association*, *93*, 1430–1439.

Ye, K. Q., Li, W., & Sudjianto, A. (2000). Algorithmic construction of optimal supersaturated designs. *Journal of Statistical Planning and Inference*, *90*(1), 145–159.

Zhang, M., Dumitrascu, B., Williamson, S., & Engelhardt, B. (2019). Sequential gaussian processes for online learning of nonstationary functions. *ArXiv*, *abs/1905.10003*.

Zhu, Y., & Fujimura, K. (2010). A bayesian framework for human body pose tracking from depth image sequences. *Sensors (Basel, Switzerland)*, *10*, 5280–5293.