Numerical Construction of K-Optimal Designs for Linear, Nonlinear, and Generalized Linear Models

by

Xiaoqing Zhang
B.Sc., University of Victoria, 2020

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Mathematics and Statistics

Numerical Construction of K-Optimal Designs for Linear, Nonlinear, and Generalized Linear Models

by

Xiaoqing Zhang
B.Sc., University of Victoria, 2020

Supervisory Committee

Dr. Julie Zhou, Supervisor
(Department of Mathematics and Statistics)

Dr. Min Tsao, Department Member
(Department of Mathematics and Statistics)

Supervisory Committee

Dr. Julie Zhou, Supervisor
(Department of Mathematics and Statistics)

Dr. Min Tsao, Department Member
(Department of Mathematics and Statistics)

ABSTRACT

This thesis investigates the numerical construction of K-optimal designs for a variety of statistical models. These include linear models such as polynomial, trigonometric, and second-order response models, nonlinear models such as Michaelis–Menten, compartmental, and Peleg models, and generalized linear models with a particular focus on logistic regression. K-optimality aims to minimize the condition number of the Fisher information matrix to improve the numerical stability in parameter estimation. A general algorithm is proposed and applied to all models to construct K-optimal designs, evaluated under different design spaces and parameter values. For nonlinear models, the K-optimal designs are compared with A-optimal and D-optimal designs, while for the logistic regression model, comparisons are made with D-optimal designs. The results show that K-optimal designs have stable patterns between different models. Factors such as design space, model type, and parameter values influence the support points, their weights, and the condition number. In addition, K-optimal designs achieve smaller condition numbers, indicating better numerical stability, and take less computation time than both D-optimal and A-optimal designs. All key findings are presented in tables and figures, and the MATLAB code used for the computations is provided in the thesis.

Table of Contents

List of Tables

List of Figures

Acknowledgements

I would like to thank my supervisor, Dr. Julie Zhou, for all her help and support during my studies. She has cared about both my academic work and my personal life. During these years, I faced some family challenges, but she was always patient and supportive. Her guidance went beyond my studies and I could not have reached this stage without her help.

I would also like to thank Dr. Min Tsao for his time as my committee member and Dr. Po Yang for serving as my external examiner.

Finally, my deepest thanks go to my family for always supporting me while I pursued my studies. They have been with me through all the challenges and we have overcome them together. I could not have completed this thesis without their love and encouragement.

Chapter 1

Introduction

The design of experiments (DOE) was developed by Ronald A. Fisher in the 1920s and 1930s (Telford, 2007). It is a structured method for running tests and analyzing the results to examine how different factors influence an outcome. It has been used in a variety of industries (Bowden et al., 2019). For example, Hoshmand (2006) demonstrated the application of DOE in agriculture and natural sciences by using real-world examples to show how practical issues in experimental design and analysis can be effectively addressed. In addition, DOE is used in the field of quality control and quality assurance to reduce defects and variation, as well as to improve quality (Lamidi et al., 2024). Although DOE helps identify significant factors and structure experiments, it does not determine the best combination of factor levels or indicate where to place the most efficient experimental points. To address this limitation, optimal design theory was developed to select experimental conditions based on various criteria to achieve the most precise parameter estimates and stable solutions.

In this thesis, we explore the theory and application of optimal design, with a particular focus on K-optimality proposed in Ye and Zhou (2013). Chapter 1 introduces the background and motivation for this study. In Chapter 2, we examine the K-optimal design criterion and present a general algorithm for computing K-optimal designs. Chapters 3 and 4 apply this algorithm to various models, including linear, nonlinear, and generalized linear regression models. Furthermore, we compare the performance of the K-optimal designs with the D- and A-optimal designs to investigate how they differ in numerical behavior across various models. The conclusions and limitations are summarized in Chapter 5.

1.1 Optimal design of experiment

Among the various types of experimental designs in the science of DOE, optimal design focuses on selecting the most informative test points for a given design criterion and a specified number of runs (Morgan-Wall and Khoury, 2021). It aims to collect data in the most efficient way to ensure reliable estimation of the model parameters (Li, 2022). In fact, the idea behind the optimal design theory came before the formal development of DOE. In 1918, Danish statistician Kirstine Smith published a pioneering paper exploring the core idea of optimal design. She provided advice on how to design an experiment by minimizing the maximum prediction variance in the study space in polynomial regression. This approach later became known as G-optimality, although her contribution was not widely recognized until about 30 years later (Walsh and Borkowski, 2022). In 1959, Jack Kiefer developed a general framework for constructing optimal designs by using the Fisher information matrix to formalize specific optimality criteria (Kiefer, 1959). Since then, optimal designs have been developed and have become widely used in various research fields (Berni et al., 2022). For example, Dette et al. (2012) applied optimal design theory to improve experiments in pharmacokinetic–pharmacodynamic models. They showed that their proposed designs are more robust and efficient than those commonly used in practice. In addition, Rodriguez et al. (2022) applied optimal design in an environmental science study to improve the understanding of atrazine degradation in the soil. Their results suggest that optimal design improves model discrimination and increases knowledge from laboratory and field experiments in soil systems. Moreover, Berni et al. (2022) applied optimal design in the rail freight sector to optimize hierarchical engineering systems. They found that optimal design helped achieve the best train configuration and showed that their approach can be successfully applied to similar technological problems. These examples demonstrate that optimal design is a powerful tool for improving experimental efficiency and addressing complex design challenges.

1.2 Various design criteria

Optimal design is used to select the most informative design points, and it is guided by a chosen criterion which is a scalar function of the Fisher information matrix. The Fisher information matrix measures how much information the observed data carry about the unknown parameters that need to be estimated in a model (Pavithra and Deepak, 2022). Therefore, design criteria focus on the Fisher information matrix and guide how to maximize information about the parameters of interest. One of the most commonly used criteria is the D-optimality criterion, which maximizes the determinant of the Fisher information matrix. D-optimal design was introduced by Chernoff (1953) and was further studied by many other researchers (Kiefer, 1959; Kiefer and Wolfowitz, 1959). It has been used in various fields. Padmanabhan et al. (2010) applied D-optimal designs in clinical trials to identify the most informative and effective dose levels of a new drug. Moreover, Rahman et al. (2022) applied D-optimality to optimize the osmotic dehydration of zucchini cubes by experimenting with various osmotic agents at different concentrations.

Other popular criteria include A-optimality, which minimizes the trace of the inverse of the Fisher information matrix, and E-optimality, which maximizes the smallest eigenvalue of the Fisher information matrix. Both were formalized by Kiefer (1959) and have since been further developed and applied in practice by many researchers (Jacroux and Notz, 1983; Dette and Sahm, 1998; Ye et al., 2016; Jones et al., 2020; Müller and Schorning, 2023). In addition to these classical criteria, more recent work has proposed alternative approaches such as the K-optimality criterion. It focuses on a different aspect of the information matrix to define optimality and will be introduced in the next section.

1.3 K-optimality criterion

A key concept underlying the K-optimal design criterion is the condition number of the Fisher information matrix, where it measures how sensitive the solution of the equations is to errors (Strang, 2016, p.520). High condition numbers result in high variances and unstable solutions. The term condition number was defined by Turing (1948). He suggested using condition numbers to measure the ill-conditioning of a matrix for linear systems. There are two measures, the N-condition number and the M-condition number, which are defined differently from what we focused on. Later, Todd (1949) investigated the work of a few mathematicians and suggested a condition number that uses the 2-norm or Euclidean norm to get the ratio of the largest to the smallest eigenvalue of a matrix, and it is called the P-condition number. This thesis is based on the P-condition number. Subsequently, Kaplan and Gentry (1987) proposed using condition number as a criterion for experimental design, which led to its being used in optimal design.

Ye and Zhou (2013) introduced the K-optimal design criterion, which minimizes the condition number of the Fisher information matrix. They have obtained theoretical and numerical results of K-optimal designs for polynomial regression models. Later, Rempel and Zhou (2014) investigated numerical methods for computing K-optimal designs for factorial experiments of linear models. They compared K-optimal designs with other optimal designs and found similar results. More recently, Yue et al. (2023) established several theoretical results for K-optimal designs for polynomial, trigonometric, and second-order response models and developed a geneal numerical method for finding the K-optimal design.

From these contributions, we can see that the K-optimal design is relatively new compared to classical criteria such as D-, A-, and E-optimality. It focuses on minimizing the condition number, which allows it to address problems of ill-conditioning that can weaken the stability and reliability of parameter estimates in experimental design. For example, Duarte et al. (2023) used the support points of K-optimal designs to construct the surface response, then used this surface to transform the original parameters into a new set with improved orthogonality and numerical stability. This motivated my interest in K-optimal design because it provides a useful alternative to traditional methods and offers a new way to improve experimental design. As it is a recent development, there are many interesting findings to be discovered by further exploring it.

As noted by Ye and Zhou (2013) and Yue et al. (2023), deriving theoretical results for K-optimal designs is challenging, even for relatively simple models. Extending K-optimality to more complex models makes theoretical derivations much more difficult to obtain analytically. Therefore, in this thesis, I focus on the numerical computation of the K-optimal design, applying the existing algorithm that developed by Yue et al. (2023) to explore it in more complex and varied settings.

1.4 Main contributions

This thesis makes several contributions to the study of K-optimal experimental design numerically.

1. We applied the numerical algorithm developed by Yue et al. (2023) to compute K-optimal designs on discrete design spaces for more complex models than previously studied.

2. We computed K-optimal designs for a second-order response model in a three-dimensional design space.

3. We computed K-optimal designs for several nonlinear models, including the Michaelis–Menten, compartmental, and Peleg models.

4. We computed K-optimal designs for a generalized linear model, specifically a logistic regression model.

5. We computed K-optimal designs for polynomial and trigonometric models under different design spaces and parameter settings than previously studied.

6. We compared the performance of K-optimal designs with D- and A-optimal designs for the nonlinear models, and with D-optimal designs for the logistic regression model.

7. I also worked on the joint research paper (Yue et al., 2023), which is part of my MSc research program. The theoretical results of the paper are reviewed in Chapter 2 of this thesis.

Chapter 2

K-optimal Design Criterion

Consider a general form of the regression model after n trials of experiments,

$$y_i = g(\boldsymbol{x_i}; \boldsymbol{\theta}) + \epsilon_i, i = 1, \cdots, n, \tag{2.1}$$

where y_i is the value of a dependent variable y corresponding to the ith design point $\boldsymbol{x_i} \in S \subset R^p$, $g(\boldsymbol{x_i}; \boldsymbol{\theta})$ is the response function depending on $\boldsymbol{x_i}$ and $\boldsymbol{\theta} = (\theta_0, \theta_1, \ldots, \theta_q)^T$ is the vector of unknown regression parameters; ϵ_i is the error term associated with observation y_i and is assumed to be independent and identically distributed with mean zero and variance σ^2, and n is the sample size.

In this chapter, we study an optimal design criterion to choose design points, $\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}$, so that we can obtain the most information for $\boldsymbol{\theta}$. We begin with the details of an approximate design in a discrete design space. We then discuss the Fisher information matrix for linear and nonlinear regression models. In addition, we formulate optimal design problems under K-optimality based on its criterion and convexity. At the end of the chapter, we develop an algorithm to find K-optimal designs for any regression models. The main results of this chapter are from Yue et al. (2023), and I am a co-author of the paper.

2.1 Discrete design space and approximate design

The design points chosen from a continuous design space have infinite possibilities. Obtaining precise measurements of $\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}$ is also a challenge. A discrete design space is a finite set, and it is recommended to find optimal designs in real-world situations. Some examples demonstrate the benefits of using a discrete design space in various field studies. In the medical field, a study might test specific doses of drugs. Often discrete doses are considered, such as 0, 50, 100,..., 950, and 1000 mg, rather than precisely measuring the doses as decimals. In the marketing field, different price points are tested to see the effect on the product's sales. If the price range is from \$10 to \$20, it is effective to test the whole dollar value \$10, \$11,..., \$20 first and then narrow down to test some typical values such as 25, 75, or 99 cents. Testing every possible price point in the range is unrealistic and overwhelming. In the field of botany, the height of the plant is measured over time. All possible heights are assumed to range from 10 to 50cm. It will be less challenging to implement if the design space is discrete, with measurements at 10, 11, . . ., 50cm instead of decimal values. Therefore, using a discrete design space in optimal design problems is more efficient and effective for many applications than using a continuous design space.

There are two types of designs studied in optimal design, exact and approximate (Nyarko and Doku-Amponsah, 2022). The exact design requires a specific number of subjects assigned to each $\boldsymbol{x_i}$, while the approximate design only requires knowing the proportion of the total count for each $\boldsymbol{x_i}$. Kiefer (1985) pioneered the concept of approximate designs and provided multiple reasons for using approximate designs instead of exact designs. For example, it would be complicated to assign a specific number of subjects to each $\boldsymbol{x_i}$ when the total sample counts change each time. Therefore, the approximate optimal design is preferred. It often works with discrete design space and is used in this thesis. Weights are introduced in approximate designs, representing the proportion of each design point used in the experiment.

When the weight of the points is greater than 0, those points are selected as the supporting points of the design.

The distinctive design points $\boldsymbol{u_1}, \ldots, \boldsymbol{u_N}$ in R^p are taken to form a discrete design space S_N with N points. An approximate design ξ on S_N is given as

$$\xi(\boldsymbol{w}) = \begin{pmatrix} \boldsymbol{u_1} & \boldsymbol{u_2} & \cdots & \boldsymbol{u_N} \\ w_1 & w_2 & \cdots & w_N \end{pmatrix},$$

where $\boldsymbol{w} = (w_1, w_2, \ldots, w_N)^\top$ is the weight vector, and each weight satisfies $w_i \geq 0$ and $\sum_{i=1}^N w_i = 1$. Let r_i represent the number of experiments that are repeated at each selected design point $\boldsymbol{u_i}, i = 1, \ldots, N$. To implement the design, an integer number of points is taken. Therefore, the number of points at $\boldsymbol{u_i}$ is rounded from nw_i, such as $r_i = [nw_i]$ for $i = 1, \ldots, N$, where [ ] is rounded to an integer. Also note that $\sum_{i=1}^N r_i = n$.

2.2 Fisher information matrix

Let $\hat{\boldsymbol{\theta}}$ be the least squares estimator (LSE) of $\boldsymbol{\theta}$ in model (2.1), which is defined as

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \left(y_i - g\left(\boldsymbol{x_i}; \boldsymbol{\theta}\right)\right)^2. \tag{2.2}$$

This minimization leads to a set of normal equations that can be solved analytically if $g\left(\boldsymbol{x_i}; \boldsymbol{\theta}\right)$ is a linear function of $\boldsymbol{\theta}$; otherwise, it is solved numerically for nonlinear models. The asymptotic covariance matrix of $\hat{\boldsymbol{\theta}}$ is approximated by

$$\mathrm{Cov}(\hat{\boldsymbol{\theta}}) = \frac{\sigma^2}{n} \boldsymbol{A^{-1}}, \tag{2.3}$$

where $\boldsymbol{A}$ is the Fisher information matrix, defined as

$$\boldsymbol{A} = \frac{1}{n}\sum_{i=1}^{n} \frac{\partial g\left(\boldsymbol{\theta}^*, \boldsymbol{x}_i\right)}{\partial \boldsymbol{\theta}} \frac{\partial g\left(\boldsymbol{\theta}^*, \boldsymbol{x}_i\right)}{\partial \boldsymbol{\theta}^T}, \tag{2.4}$$

with $\boldsymbol{\theta}^*$ being the true parameter value of $\boldsymbol{\theta}$. The Cramér-Rao lower bound is a fundamental result in estimation theory. It provides the lower limit on the variance of any unbiased parameter estimator and is obtained by the inverse of the Fisher information matrix (Chao et al., 2016). We can minimize the inverse of the Fisher information matrix using various optimality criteria to achieve the lowest variance of unbiased estimators, which results in the most precise estimates. It is clear that in equation (2.3), $\mathrm{Cov}(\hat{\boldsymbol{\theta}})$ is proportional to $\boldsymbol{A}^{-1}$. Thus, minimizing $\phi(\mathrm{Cov}(\hat{\boldsymbol{\theta}}))$ is equivalent to minimizing $\phi(\boldsymbol{A}^{\boldsymbol{-1}})$, where ϕ is a scalar function. Various scalar functions can be used in optimal design problems. For instance, the determinant of $\mathrm{Cov}(\hat{\boldsymbol{\theta}})$ is minimized for the D-optimal design, and the trace of $\mathrm{Cov}(\hat{\boldsymbol{\theta}})$ is minimized for A-optimal design.

The following example shows the Fisher information matrix in linear regression model under an approximate design on a discrete design space.

Example 2.1 Consider a linear regression model with $g\left(\boldsymbol{x_i};\boldsymbol{\theta}\right) = \boldsymbol{f}^\top\left(\boldsymbol{x}_i\right)\boldsymbol{\theta}$. Model (2.1) becomes

$$y_i = \boldsymbol{f}^\top\left(\boldsymbol{x}_i\right)\boldsymbol{\theta} + \epsilon_i, \quad i = 1, \ldots, n, \tag{2.5}$$

where $\boldsymbol{f}^\top(\boldsymbol{x}_i) = (f_0(\boldsymbol{x_i}), \cdots, f_q(\boldsymbol{x_i}))$, and $f_j(\boldsymbol{x}_i)$, $j = 0, 1, \cdots, q$, is a given function of vector $\boldsymbol{x}_i$ for each j. The LSE of $\boldsymbol{\theta}$ is

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \left(y_i - \boldsymbol{f}^\top\left(\boldsymbol{x}_i\right)\boldsymbol{\theta}\right)^2.$$

The covariance matrix of $\hat{\boldsymbol{\theta}}$ is given by

$$\operatorname{Cov}(\hat{\boldsymbol{\theta}})=\frac{\sigma^2}{n}\left(n^{-1}\boldsymbol{X}^\top\boldsymbol{X}\right)^{-1}=\sigma^2\left(\boldsymbol{X}^\top\boldsymbol{X}\right)^{-1},$$

where $\boldsymbol{X}=\left(\boldsymbol{f}\left(\boldsymbol{x}_1\right),\ldots,\boldsymbol{f}\left(\boldsymbol{x}_n\right)\right)^\top$. Therefore, the Fisher information matrix is

$$\boldsymbol{A}=\frac{1}{n}\boldsymbol{X}^\top\boldsymbol{X}.$$

For an approximate design on a discrete design space $S_N=\{\boldsymbol{u_1},\ldots,\boldsymbol{u_N}\}$, the Fisher information matrix is

$$\boldsymbol{A}(\boldsymbol{w})=\sum_{i=1}^{N}w_i\boldsymbol{f}\left(\boldsymbol{u}_i\right)\boldsymbol{f}^\top\left(\boldsymbol{u}_i\right), \tag{2.6}$$

where the wights w_i are non-negative.

Note that equation (2.6) applies to nonlinear regression model as well, where $\boldsymbol{f}\left(\boldsymbol{u}_i\right)$ is obtained from $\frac{\partial g(\boldsymbol{\theta}^*,\boldsymbol{u}_i)}{\partial\boldsymbol{\theta}}$.

Since each term of the summation in $\boldsymbol{A(w)}$ in (2.6) involves an outer product of the gradient with itself, it is clear that $\boldsymbol{A(w)}$ is symmetric. Substituting $\boldsymbol{A(w)}$ into quadratic form, $\boldsymbol{z}^\top\boldsymbol{A(w)z}$, where $\boldsymbol{z}$ can be any vector, $\boldsymbol{z}\in\mathbb{R}^q$ and $\boldsymbol{z}^\top\boldsymbol{A(w)z}$ can be expressed as,

$$\begin{aligned}
\boldsymbol{z}^\top\boldsymbol{A(w)z}&=\boldsymbol{z}^\top\sum_{i=1}^{N}w_i\frac{\partial f\left(\boldsymbol{\theta}^*,\boldsymbol{u_i}\right)}{\partial\boldsymbol{\theta}}\frac{\partial f\left(\boldsymbol{\theta}^*,\boldsymbol{u_i}\right)}{\partial\boldsymbol{\theta}^\top}\boldsymbol{z}\\
&=\sum_{i=1}^{N}w_i\boldsymbol{z}^\top\frac{\partial f\left(\boldsymbol{\theta}^*,\boldsymbol{u_i}\right)}{\partial\boldsymbol{\theta}}\frac{\partial f\left(\boldsymbol{\theta}^*,\boldsymbol{u_i}\right)}{\partial\boldsymbol{\theta}^\top}\boldsymbol{z}\\
&=\sum_{i=1}^{N}w_i\boldsymbol{z}^\top\frac{\partial f\left(\boldsymbol{\theta}^*,\boldsymbol{u_i}\right)}{\partial\boldsymbol{\theta}}\boldsymbol{z}^\top\frac{\partial f\left(\boldsymbol{\theta}^*,\boldsymbol{u_i}\right)}{\partial\boldsymbol{\theta}}\\
&=\sum_{i=1}^{N}w_i\left(\boldsymbol{z}^\top\frac{\partial f\left(\boldsymbol{\theta}^*,\boldsymbol{u_i}\right)}{\partial\boldsymbol{\theta}}\right)^2.
\end{aligned}$$

Note that the result of $\frac{\partial f(\boldsymbol{\theta}^*,\boldsymbol{u_i})}{\partial\boldsymbol{\theta}^\top}\boldsymbol{z}$ is a scalar. Since each $w_i\geq 0$ and $\left(\boldsymbol{z}^\top\frac{\partial f(\boldsymbol{\theta}^*,\boldsymbol{u_i})}{\partial\boldsymbol{\theta}}\right)^2\geq 0$, $\boldsymbol{z}^\top\boldsymbol{A(w)z}$ is nonnegative. Therefore, $\boldsymbol{A(w)}$ is a positive semi-definite matrix.

2.3 K-optimal design criterion

Unlike traditional criteria such as D- and A-optimality, which minimize functions of the inverse of the information matrix, K-optimality minimizes the condition number of the information matrix. The condition number of a matrix measures how stable the solution is when there are small errors or noise in the input. The higher the condition number, the larger the variances, which lead to unstable solutions. When a matrix $\boldsymbol{A}$ is nonsingular and matrix norm $\| \cdot \|$ is used, the condition number $\kappa(\boldsymbol{A})$ is defined as (Higham, 1995)

$$\kappa(\boldsymbol{A}) = \| \boldsymbol{A} \| \| \boldsymbol{A}^{-1} \| .$$

Additionally, when Euclidean norm is used, for a symmetric and positive definite $\boldsymbol{A}$, $\kappa(\boldsymbol{A})$ is given as (Layton and Sussman, 2020, p.81)

$$\kappa(\boldsymbol{A}) = \frac{\lambda_{\max}(\boldsymbol{A})}{\lambda_{\min}(\boldsymbol{A})}, \tag{2.7}$$

where $\lambda_{\max}$ and $\lambda_{\min}$ denote the largest and the smallest eigenvalues, respectively.

Consider model (2.5), where $\boldsymbol{A}(\boldsymbol{w})$ is a symmetric, positive semi-definite matrix. Assume $\boldsymbol{A}(\boldsymbol{w})$ is nonsingular, then $\boldsymbol{A}(\boldsymbol{w})$ is a symmetric, positive definite matrix, and $\kappa(\boldsymbol{A})$ from (2.7) is given by

$$\kappa(\boldsymbol{A}(\boldsymbol{w})) = \frac{\lambda_{\max}(\boldsymbol{A}(\boldsymbol{w}))}{\lambda_{\min}(\boldsymbol{A}(\boldsymbol{w}))}. \tag{2.8}$$

A K-optimal design can be defined as the solution $\boldsymbol{w}^*$ to the following optimization problem,

$$\min_{\boldsymbol{w}} \kappa(\boldsymbol{A}(\boldsymbol{w})) = \min_{\boldsymbol{w}} \frac{\lambda_{\max}(\boldsymbol{A}(\boldsymbol{w}))}{\lambda_{\min}(\boldsymbol{A}(\boldsymbol{w}))}, \tag{2.9}$$

where $\boldsymbol{A}(\boldsymbol{w})$ is nonsingular with each weight $w_i \geq 0$ and $\sum_{i=1}^{N} w_i = 1$.

2.4 Convexity of K-optimal design problems

Optimal design problems in statistics are often convex optimization problems. Convexity ensures that problems are easier to solve and that the global optimum is found effectively. For instance, D-optimality and A-optimality design problems are convex optimization problems. Let us discuss the convexity of the K-optimal design problem.

In equation (2.8), $\lambda_{\max}\left(\boldsymbol{A}\left(\boldsymbol{w}\right)\right)$ is a convex function of $\boldsymbol{w}$ and $\lambda_{\min}\left(\boldsymbol{A}\left(\boldsymbol{w}\right)\right)$ is a concave function of $\boldsymbol{w}$ (Boyd and Vandenberghe, 2004, p.118). If $\boldsymbol{A}\left(\boldsymbol{w}\right)$ is nonsingular, then $\boldsymbol{A}\left(\boldsymbol{w}\right)$ is a symmetric positive definite matrix as mentioned previously. Therefore, all eigenvalues of $\boldsymbol{A}\left(\boldsymbol{w}\right)$, $\lambda\left(\boldsymbol{A}\left(\boldsymbol{w}\right)\right) > 0$. Since the ratio between a nonnegative convex function and a positive concave function is quasiconvex (Agrawal and Boyd, 2020), $\kappa\left(\boldsymbol{A}\left(\boldsymbol{w}\right)\right)$ is a quasiconvex function of $\boldsymbol{w}$. It would be difficult to solve (2.9) as $\kappa\left(\boldsymbol{A}\left(\boldsymbol{w}\right)\right)$ may not be a convex function. Reformulating the quasiconvex function to a convex structure is necessary. By using the Charnes-Cooper Transformation, the following optimization problem

$$\begin{cases} \min_{\boldsymbol{w}} \frac{\lambda_{\max}(\boldsymbol{A}(\boldsymbol{w}))}{\lambda_{\min}(\boldsymbol{A}(\boldsymbol{w}))} \\ \text{subject to: } \boldsymbol{w} \geq 0, \quad \sum_{i=1}^{N} w_i = 1 \end{cases} \tag{2.10}$$

is able to convert to a convex optimization problem (Charnes and Cooper, 1962, p.184). Note that $\boldsymbol{w} \geq 0$ means that each $w_i \geq 0$ for $i = 1, \ldots, N$. Here is the transformation process.

Introduce a new vector variable $\boldsymbol{v} = \frac{\boldsymbol{w}}{\gamma}$, where $\boldsymbol{v} = \left(v_1, \ldots, v_N\right)^\top$, and $\gamma = \frac{1}{\sum_{i=1}^{N} v_i} > 0$. Rewrite the objective function in terms of $\boldsymbol{v}$ and γ as

$$\min_{\boldsymbol{w}} \frac{\lambda_{\max}\left(\boldsymbol{A}\left(\boldsymbol{w}\right)\right)}{\lambda_{\min}\left(\boldsymbol{A}\left(\boldsymbol{w}\right)\right)} = \min_{\boldsymbol{v},\gamma} \frac{\lambda_{\max}\left(\boldsymbol{A}\left(\gamma\boldsymbol{v}\right)\right)}{\lambda_{\min}\left(\boldsymbol{A}\left(\gamma\boldsymbol{v}\right)\right)}$$

$$
\begin{aligned}
&= \min_{\boldsymbol{v}} \frac{\lambda_{\max}\left(\boldsymbol{A}\left(\frac{\boldsymbol{v}}{\sum_{i=1}^{N} v_i}\right)\right)}{\lambda_{\min}\left(\boldsymbol{A}\left(\frac{\boldsymbol{v}}{\sum_{i=1}^{N} v_i}\right)\right)} \\
&= \min_{\boldsymbol{v}} \frac{\frac{\lambda_{\max}(\boldsymbol{A}(\boldsymbol{v}))}{\sum_{i=1}^{N} v_i}}{\frac{\lambda_{\min}(\boldsymbol{A}(\boldsymbol{v}))}{\sum_{i=1}^{N} v_i}} \\
&= \min_{\boldsymbol{v}} \frac{\lambda_{\max}\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right)}{\lambda_{\min}\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right)}.
\end{aligned}
$$

Let us discuss the constraints of the new design problem. $\lambda\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right)$ is nonnegative as $\boldsymbol{v} = \frac{\boldsymbol{w}}{\gamma}$ with $\gamma > 0$. If $\lambda_{\min}\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right)$ approaches zero, the condition number could become arbitrarily large, making the solution unstable. In the transformed problem, if we multiply $\boldsymbol{v}$ by any positive constant, all eigenvalues of $\boldsymbol{A}\left(\boldsymbol{v}\right)$ change by the same factor, so the condition number stays the same. Therefore, we can fix this free scale by setting $\lambda_{\min}\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right) = 1$, which removes the denominator and simplifies the optimization. Moreover, since $\lambda_{\min}\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right)$ is a concave function of $\boldsymbol{v}$, $\lambda_{\min}\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right) \geq 1$ gives a convex set of $\boldsymbol{v}$ (Boyds and Vandenberghe, 2004, p.70). Hence, based on Lu and Pong (2011), a convex optimization problem for K-optimal design is given by

$$
\begin{cases}
\min_{\boldsymbol{v}} \lambda_{\max}\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right) \\
\text{subject to: } \boldsymbol{v} \geq 0, \quad \lambda_{\min}\left(\boldsymbol{A}\left(\boldsymbol{v}\right)\right) \geq 1.
\end{cases}
\tag{2.11}
$$

Also, from Lu and Pong (2011) and Yue et al. (2023), we can conclude the following connection between (2.10) and (2.11):

C1. Suppose $\boldsymbol{w}^*$ and $\boldsymbol{v}^*$ are solutions to (2.10) and (2.11), respectively. Then $\frac{\lambda_{\max}(\boldsymbol{A}(\boldsymbol{w}^*))}{\lambda_{\min}(\boldsymbol{A}(\boldsymbol{w}^*))} = \lambda_{\max}\left(\boldsymbol{A}\left(\boldsymbol{v}^*\right)\right)$.

C2. If $\boldsymbol{w}^*$ is a solution to (2.10), then $\boldsymbol{v}^* = \frac{\boldsymbol{w}^*}{\lambda_{\min}(\boldsymbol{A}(\boldsymbol{w}^*))}$ is a solution to (2.11).

C3. If $\boldsymbol{v}^*$ is a solution to (2.11), then $\boldsymbol{w}^* = \frac{\boldsymbol{v}^*}{\sum_{i=1}^{N} v_i^*}$ is a solution to (2.10).

These results confirm that fixing $\lambda_{\min}(\boldsymbol{A}(\boldsymbol{v})) = 1$ leads to an equivalent formulation of the K-optimal design problem. Therefore, if we can find the solution $\boldsymbol{v}^*$ to (2.11), then we scale it to $\frac{\boldsymbol{v}^*}{\sum_{i=1}^N v_i^*}$ to obtain the solution $\boldsymbol{w}^*$ to (2.10). The K-optimal design is $\xi\left(\boldsymbol{w}^* = \frac{\boldsymbol{v}^*}{\sum_{i=1}^N v_i^*}\right)$.

2.5 Finding K-optimal design

The CVX in MATLAB can solve the transformed K-optimal design problem in (2.11). CVX is a software package for disciplined convex programming (Grant and Boyd, 2020). It is a useful tool for solving convex optimization problems. The following steps compute the K-optimal design via CVX in MATLAB.

Step 1: Construct N distinct design points $\boldsymbol{u_1}, \ldots, \boldsymbol{u_N}$ in a given design space. Design points are often equally spaced, and the endpoints mark the boundary of the design space.

Step 2: For a given model, use (2.6) to calculate the information matrix, and $\boldsymbol{f}(\boldsymbol{u}_i)\,\boldsymbol{f}^\top(\boldsymbol{u}_i)$ is the information at each design point without involving the weights.

Step 3: In CVX, compute the information matrix $\boldsymbol{A}(\boldsymbol{v})$ where $\boldsymbol{v}$ satisfies: $\boldsymbol{v} \geq 0$ and $\lambda_{\min}(\boldsymbol{A}(\boldsymbol{v})) \geq 1$. Solve problem (2.11) to get the solution $\boldsymbol{v}^*$.

Step 4: Get the solution $\boldsymbol{w}^* = \frac{\boldsymbol{v}^*}{\sum_{i=1}^N v_i^*}$, and print the design points if their weights are greater than a given threshold, such as 10^{-4} or 10^{-5}.

It is easy to find K-optimal designs for any regression model using the above steps. In Chapter 3, we will construct K-optimal designs for various models using CVX in MATLAB.

Chapter 3

Computing K-optimal Designs for Linear Models

In this chapter, we compute K-optimal designs for polynomial, trigonometric, and second-order response models. We present and discuss the key features and results of K-optimal designs. Representative MATLAB code is provided in the Appendix. All computations in this thesis were performed on a laptop with an 8-core processor and 16GB of RAM.

3.1 K-optimal designs for polynomial models

Consider the dth degree polynomial regression models with one predictor variable x_i, model (2.1) becomes

$$y_i = \boldsymbol{f}^\top(x_i)\,\boldsymbol{\theta} + \epsilon_i, \quad i = 1, \ldots, n, \tag{3.1}$$

where $\boldsymbol{f}^\top(x_i) = \left(1, x_i, x_i^2, \cdots, x_i^d\right)$, and x_i are the design points located within the design space $S = [a, b]$. Also note that $d \geq 1$. We use an equally spaced grid of points to construct a discrete design space, denoted as $S_N = \{u_1, \ldots, u_N\}$, where $u_i = a + \frac{i-1}{N-1}(b-a)$, $i = 1, \ldots, N$. The information matrix $\boldsymbol{A}(\boldsymbol{w})$ is a square matrix of size $(d+1) \times (d+1)$, which is defined in (2.6), with $\boldsymbol{f}^\top(u_i) = \left(1, u_i, u_i^2, \cdots, u_i^d\right)$.

Following the steps outlined in Section 2.5 to find K-optimal designs in MATLAB for polynomial models, we set $N = 1001$ and apply a threshold of 10^{-5} to all designs to allow comparison. There are two cases to consider in the polynomial regression model. The first case is for the asymmetric design space $S = [a, b]$, where a and b can be arbitrary numbers. We focus on $S = [0, b]$, or $S = [-b, 0]$ with $b > 0$. Representative results are given in Table 3.1. When $d = 1$ and $b = 1, 2, \ldots, 100$, each K-optimal design results in a 2×2 information matrix with two support points at 0 and b, and most weights are concentrated at 0. For both $d = 2$ and $d = 3$, K-optimal designs include the two boundary points 0 and b. When $d = 2$, the designs consist of three support points, while when $d = 3$, they include four or more points. From the table, we can see that for different values of d, the weight at support point 0 increases as b increases. Figure 3.1 shows that the condition number generally decreases as b increases for $d = 1, 2$ and 3, with the maximum value of b set to 100. There are two reasons for the gaps in the figures, some values are missing because the K-optimal design could not be solved in MATLAB, and others were excluded to make the graphs easier to read if their condition numbers were greater than 100. As d increases, the starting point of the plot shifts higher and more values are omitted. This shows that complex models make the K-optimal design problem difficult to solve numerically. Table 3.2 presents the minimum condition numbers achieved for K-optimal designs of different polynomial degrees, along with the corresponding running times. The running time increases with model complexity, from $1 \sim 2$ seconds for $d = 1$ to $3 \sim 4$ seconds for $d = 3$. Similarly, the minimum condition number increases from 1 to approximately 5.83.

The second case is for the symmetric design space about 0, where $S = [-b, b]$, with $b > 0$. In this case, the support points of the K-optimal design for polynomial models are symmetric about 0 for any $d \geq 1$. The explanation is as follows. Assume that each distinct design point u_i, $i = 1, 2, \ldots N$, in the design space S_N, has its reflection $-u_i$, which also belongs to S_N. Under this assumption, there are symmetric K-optimal designs for all $d \geq 1$.

In equation (2.6), the information matrix of point u_i and its reflection point $-u_i$ is denoted as $\boldsymbol{f}(u_i)\boldsymbol{f}^\top(u_i)$ and $\boldsymbol{f}(-u_i)\boldsymbol{f}^\top(-u_i)$, respectively. Consider a diagonal matrix $\boldsymbol{Q}$ with entries of $1, -1, 1, \ldots, (-1)^d$. Then a reflection design of $\xi(\boldsymbol{w})$ has an information matrix

$$\sum_{i=1}^{N} w_i \boldsymbol{f}(-u_i)\boldsymbol{f}^\top(-u_i) = \sum_{i=1}^{N} w_i \boldsymbol{Q}\boldsymbol{f}(u_i)\boldsymbol{f}^\top(u_i)\boldsymbol{Q}^\top = \boldsymbol{Q}\boldsymbol{A}(\boldsymbol{w})\boldsymbol{Q}^\top.$$

Since $\boldsymbol{Q}\boldsymbol{A}(\boldsymbol{w})\boldsymbol{Q}^\top$ and $\boldsymbol{A(w)}$ have the same eigenvalues(Nicholson, 2019, p.300), the condition numbers for them are the same. This implies that K-optimal designs are symmetric. Table 3.3 lists the K-optimal designs in $S = [-1, 1]$ for $d = 1, \ldots, 5$. Each design has $d+1$ support points that always include the two boundary points. The boundary or near-boundary points tend to receive less weight compared to the interior points. The condition number increases dramatically from 1 at $d = 1$ to about 843 at $d = 5$. Figure 3.2 shows how the condition number changes as the design space $[-b, b]$ increases, for models with degree $d = 1, 2$ and 3, with b ranging from 1 to 50. In the linear model, the condition number stays constant at 1. The quadratic and cubic models both start with high condition numbers when b is small and quickly drop to more stable values as b increases. However, the cubic model shows gaps in the plot, which occur for the same reasons as in the asymmetric design case, either because MATLAB could not solve the design problem or the condition number was too large to include. By comparing Figures 3.1 and 3.2, we can see that the condition number reaches its minimum more quickly in symmetric designs compared to asymmetric designs.

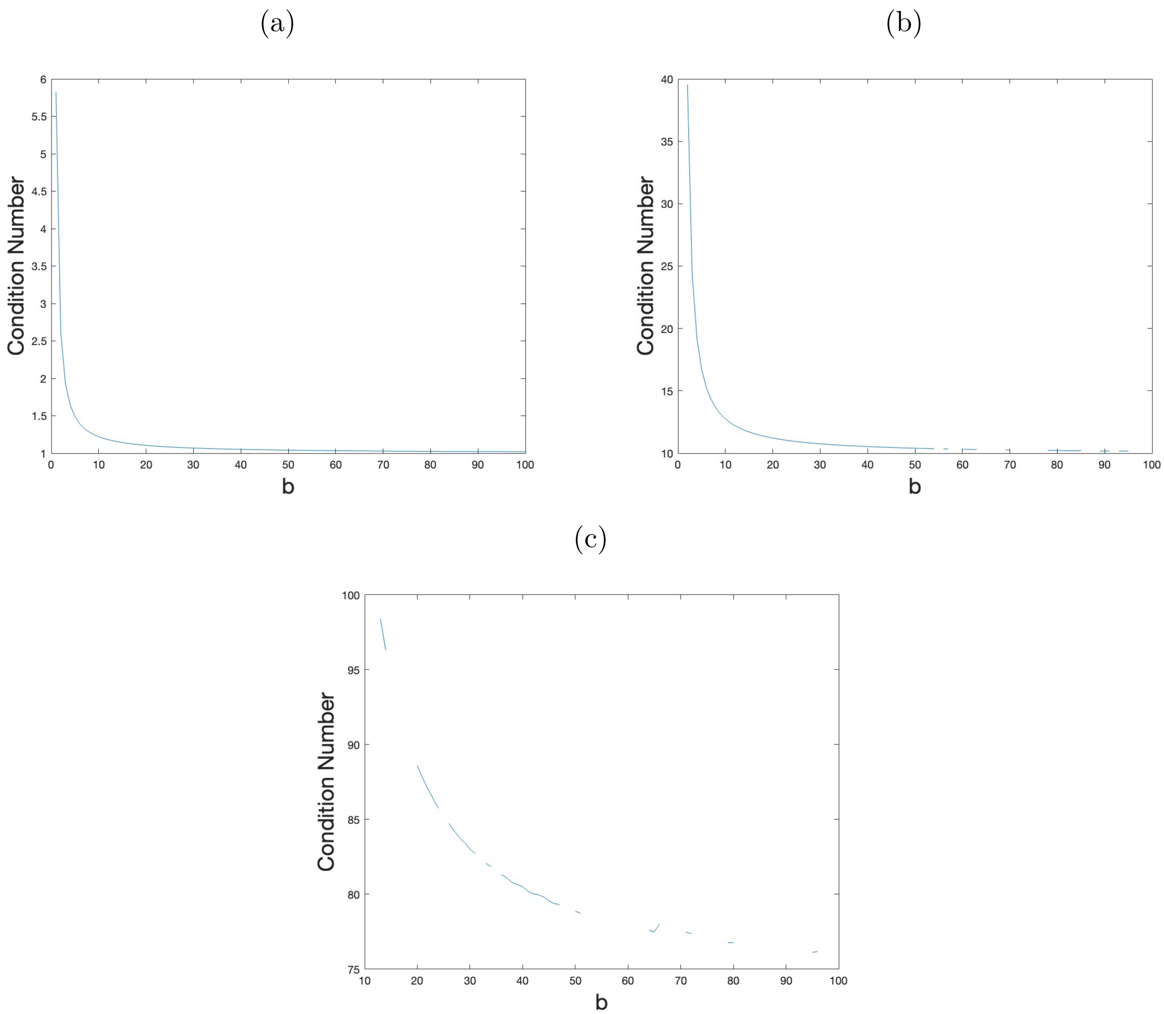


Figure 3.1: Plots of the condition number of K-optimal designs versus b, where b is the boundary point of design space $S = [0, b]$, for three models: (a) linear model, (b) quadratic model, (c) cubic model.

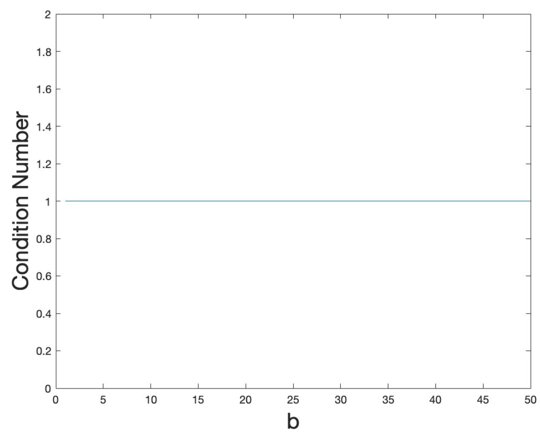
Table 3.1: K-optimal designs for polynomial regression models on $S = [0, b]$ with $N = 1001$, where designs are shown as [support point, weight].

$S = [0, b]$	K-optimal design for linear model	K-optimal design for quadratic model	K-optimal design for cubic model
$b = 1$	$\begin{bmatrix} 0 & 0.6667 \\ 1.0000 & 0.3333 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.4046 \\ 0.4390 & 0.4809 \\ 1.0000 & 0.1145 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.2919 \\ 0.2190 & 0.0001 \\ 0.2200 & 0.4386 \\ 0.7070 & 0.2072 \\ 0.7080 & 0.0002 \\ 1.0000 & 0.0621 \end{bmatrix}$
$b = 2$	$\begin{bmatrix} 0 & 0.8333 \\ 2.0000 & 0.1667 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.5651 \\ 0.6620 & 0.4097 \\ 2.0000 & 0.0252 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.4243 \\ 0.3220 & 0.4962 \\ 1.1540 & 0.0403 \\ 1.1560 & 0.0339 \\ 2.0000 & 0.0053 \end{bmatrix}$
$b = 3$	$\begin{bmatrix} 0 & 0.9091 \\ 3.0000 & 0.0909 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.6341 \\ 0.7650 & 0.3591 \\ 3.0000 & 0.0068 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.4713 \\ 0.3660 & 0.4917 \\ 1.3950 & 0.0364 \\ 3.0000 & 0.0006 \end{bmatrix}$
$b = 4$	$\begin{bmatrix} 0 & 0.9444 \\ 4.0000 & 0.0556 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.6667 \\ 0.8200 & 0.3309 \\ 4.0000 & 0.0024 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.4928 \\ 0.3880 & 0.4838 \\ 1.5440 & 0.0233 \\ 4.0000 & 0.0001 \end{bmatrix}$
$b = 5$	$\begin{bmatrix} 0 & 0.9630 \\ 5.0000 & 0.0370 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.6855 \\ 0.8550 & 0.3135 \\ 5.0000 & 0.0010 \end{bmatrix}$	$\begin{bmatrix} 0 & 0.5052 \\ 0.4050 & 0.4771 \\ 1.6400 & 0.0177 \\ 5.0000 & 0.0000 \end{bmatrix}$

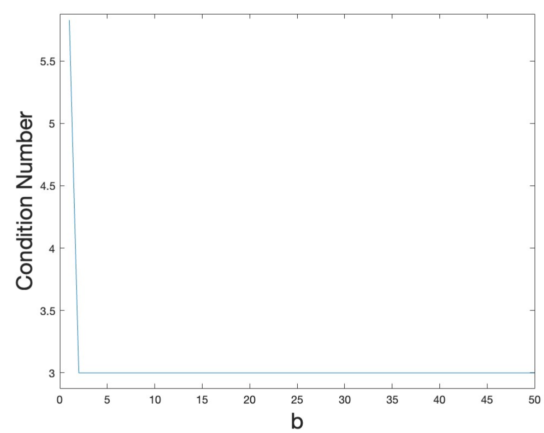
Table 3.2: Running time of each asymmetric design, and minimum condition number can be achieved on $b = 1, 2, \ldots, 100$

degree d	$d = 1$	$d = 2$	$d = 3$
Running time in second	$1 \sim 2$	$2 \sim 3$	$3 \sim 4$
Minimum condition number	1	3	5.8284

(a)



(b)



(c)

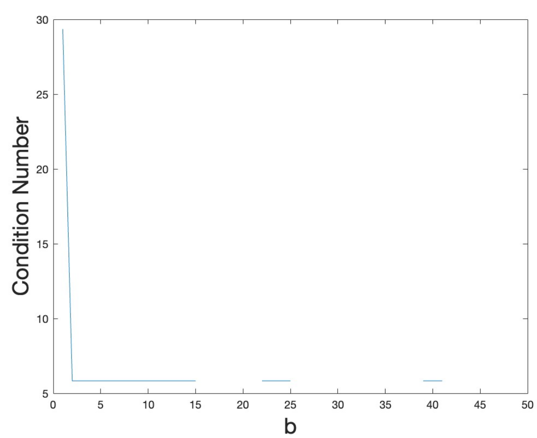


Figure 3.2: Plots of the condition number of K-optimal designs versus b, where b is the boundary point of design space $S = [-b, b]$, for three models: (a) linear model, (b) quadratic model, (c) cubic model.

Table 3.3: K-optimal design for polynomial regression models on $S = [-1, 1]$ with $N = 1001$.

dth degree of polynomial	K-optimal design [support point, weight]	condition number
$d = 1$	$\begin{bmatrix} -1.0000 & 0.5000 \\ 1.0000 & 0.5000 \end{bmatrix}$	1.0000
$d = 2$	$\begin{bmatrix} -1.0000 & 0.1667 \\ 0 & 0.6667 \\ 1.0000 & 0.1667 \end{bmatrix}$	5.8284
$d = 3$	$\begin{bmatrix} -1.0000 & 0.0965 \\ -0.4580 & 0.4035 \\ 0.4580 & 0.4035 \\ 1.0000 & 0.0965 \end{bmatrix}$	29.3553
$d = 4$	$\begin{bmatrix} -1.0000 & 0.0571 \\ -0.6620 & 0.2405 \\ 0 & 0.4048 \\ 0.6620 & 0.2405 \\ 1.0000 & 0.0571 \end{bmatrix}$	160.2101
$d = 5$	$\begin{bmatrix} -1.0000 & 0.0441 \\ -0.7800 & 0.1527 \\ -0.2860 & 0.3032 \\ 0.2860 & 0.3032 \\ 0.7800 & 0.1527 \\ 1.0000 & 0.0441 \end{bmatrix}$	842.6604

3.2 K-optimal designs for trigonometric models

Consider the trigonometric regression model with one predictor variable x, and the model (2.1) becomes

$$y_i = \boldsymbol{f}^\top(x_i)\,\boldsymbol{\theta} + \epsilon_i, \quad i = 1, \ldots, n, \tag{3.2}$$

where $\boldsymbol{f}^\top(x_i) = (1, \sin(x_i), \cos(x_i), \sin(2x_i), \cos(2x_i), \cdots, \sin(dx_i), \cos(dx_i))$, design points $x_i \in S = [a\pi, b\pi]$, and $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_{2d})$, with $d \geq 1$. An equally spaced grid of points is used to construct a discrete design space $S_N = \{u_1, \ldots, u_N\}$, where $u_i = a\pi + (b - a)\pi(i-1)/(N-1)$, $i = 1, \ldots, N$. The information matrix $\boldsymbol{A}(\boldsymbol{w})$ is a square matrix of size $(2d+1) \times (2d+1)$, which is defined in (2.6), with $\boldsymbol{f}^\top(u_i) = (1, \sin(u_i), \cos(u_i), \sin(2u_i),$ $\cos(2u_i), \ldots, \sin(du_i), \cos(du_i))$.

Follow the steps outlined in Section 2.5 to find the K-optimal designs in MATLAB for trigonometric models. We set $N = 201$ and apply a threshold of 10^{-5} to all designs to obtain consistent comparisons. There are two cases to consider for trigonometric models. The first case considers the asymmetric design space $S = [0, a\pi]$. Since the trigonometric model performs periodic behavior, we constrain a in the interval $(0, 2]$ to ensure that the design points remain within and cover a complete period of the trigonometric function. For each model with $d = 1, 2, \ldots, 4$, we construct K-optimal designs for various asymmetric design spaces $[0, a\pi]$, where $a = \frac{1}{8}, \frac{2}{8}, \ldots, \frac{16}{8}$. The minimum condition number can be achieved by the K-optimal design for any trigonometric regression model where $d \geq 1$ is 2 (Yue et al., 2023). Table 3.4 shows that the running time for each design increases as the model becomes complicated, from $1 \sim 2$ seconds for $d = 1$ to $6 \sim 7$ seconds for $d = 6$. Table 3.5 presents the value of a at which the condition number reaches its minimum of 2 for each degree, where the design space is defined as $[0, a\pi]$, along with the corresponding Fisher information matrix. As the degree d increases from 1 to 4, the required values of a also increase. For example, when $d = 1$, the minimum condition number is attained for $a \geq \frac{11}{8}$, whereas for $d = 4$, it is only achieved when $a \geq \frac{15}{8}$. This indicates that increasing the design space helps complex models reach the minimum condition number. Furthermore, the information matrix $\boldsymbol{A(w)}$ of the K-optimal design is a diagonal matrix with the first element on the diagonal being one and the remaining elements on the diagonal being $\frac{1}{2}$. Moreover, when the condition number reaches its minimum value of 2 for each design, the resulting K-optimal design consists of N support points. Each of these design points becomes a support point in the optimal design and is assigned a non-zero weight. Figure 3.3 shows the relationship between the conditional number and various design spaces $[0, a\pi]$, where $a = \frac{1}{8}, \frac{2}{8}, \ldots, \frac{16}{8}$, for the trigonometric model when $d = 1, 2, \ldots, 4$. The condition number is set to not exceed 100 for easy visualization. We can see that the condition number decreases as the design space increases. Additionally, as the model becomes complicated, the design space needs to be large to reach the minimum condition number. Figure 3.3 and Table 3.5 are consistent, each showing the condition number is minimized at the same a.

The second case considers the symmetric design space $S = [-a\pi, a\pi]$. We constrain a in the interval $(0, 1]$ for the same reason as in the first case. Using the algorithm in Section 2.5, the K-optimal design resulted in symmetry about zero if the design space is symmetric. Table 3.6 gives the K-optimal design for the trigonometric model in two specific scenarios: when $a = \frac{1}{2}$ for $d = 1, 2, \ldots, 4$, and when $a = \frac{3}{4}$ for $d = 2, 3$, and 4. Only non-negative design points and their corresponding weights are presented in the table. The condition number decreases as the design space is expanded. However, it increases as the model gets complicated. The running time for each design increases with model complexity, but adjustments to the design spaces have only a minor impact on it. For completeness, we also implemented additional cases that were not reported in Table 3.6. In particular, when $d = 1$ on $S = [\frac{-3\pi}{4}, \frac{3\pi}{4}]$, the minimum condition number of 2 is attained, which leads to N support points. Since this produces a very large number of support points, it is impractical to display them within the table, and this is discussed here instead. A similar pattern occurs whenever the condition number reaches 2 where the information matrix has the same size and structure as shown in Table 3.5 and all N design points are used with nonzero weights. In particular, this situation also occurs for $d = 1, 2, \ldots, 6$ in the largest symmetric design space $S = [-\pi, \pi]$.

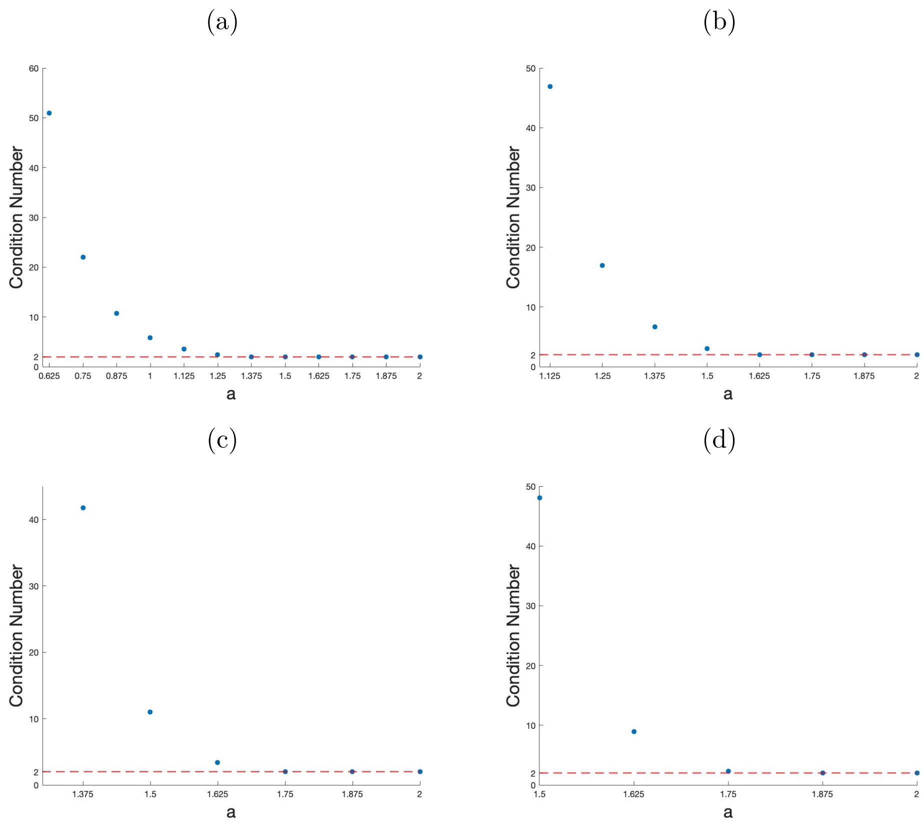


Figure 3.3: Plots of the condition number of K-optimal designs versus a, where a decides the boundary point of design space $S = [0, a\pi]$, for trigonometric models where: (a) $d = 1$, (b) $d = 2$, (c) $d = 3$, (d) $d = 4$.

Table 3.4: Running time of each K-optimal design for trigonometric regression models on $S = [0, a\pi]$ where $a \in (0, 2]$ with $N = 201$.

d	$d=1$	$d=2$	$d=3$	$d=4$	$d=5$	$d=6$
Running time in second	$1 \sim 2$	$2 \sim 3$	$3 \sim 4$	$4 \sim 5$	$4 \sim 5$	$6 \sim 7$

Table 3.5: The information matrix of K-optimal design for trigonometric regression models when minimum condition number at the value of 2 is achieved for various cases. Each case is represented by the value of d and the value of a.

d	Design space $[0,\ \mathrm{a}\pi]$	Information Matrix
$d=1$	$a=\frac{11}{8},\frac{12}{8},\ldots,\frac{16}{8}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} \end{bmatrix}$
$d=2$	$a=\frac{13}{8},\frac{14}{8},\ldots,\frac{16}{8}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}$
$d=3$	$a=\frac{14}{8},\frac{15}{8}$ and $\frac{16}{8}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}$
$d=4$	$a=\frac{15}{8}$ and $\frac{16}{8}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}$

Table 3.6: K-optimal design for various values of d in trigonometric regression models on $S = [-a\pi, a\pi]$ where $a = \frac{1}{2}$ or $\frac{3}{4}$ with $N = 201$.

a	d	Condition number	Non-negative design points for the K-optimal design with corresponding weights in parentheses	Running time (s)
$a = \frac{1}{2}$	$d = 1$	5.8284	$0\,(0.3333)$, $\frac{\pi}{2}\,(0.3333)$	$2 \sim 3$
	$d = 2$	140.4179	$0\,(0.2201)$, $\frac{8\pi}{25}\,(0.2249)$, $\frac{\pi}{2}\,(0.1651)$	$3 \sim 4$
	$d = 3$	4198.6561	$0\,(0.2054)$, $\frac{21\pi}{100}\,(0.1320)$, $\frac{21\pi}{50}\,(0.1510)$, $\frac{\pi}{2}\,(0.1144)$	$4 \sim 5$
	$d = 4$	135134.3149	$0\,(0.1304)$, $\frac{33\pi}{200}\,(0.1358)$, $\frac{8\pi}{25}\,(0.1105)$, $\frac{9\pi}{20}\,(0.1165)$, $\frac{\pi}{2}\,(0.0720)$	$6 \sim 7$
$a = \frac{3}{4}$	$d = 2$	3.0370	$0\,(0.1509)$, $\frac{81\pi}{200}\,(0.0191)$, $\frac{33\pi}{80}\,(0.1776)$, $\frac{3\pi}{4}\,(0.2279)$	$3 \sim 4$
	$d = 3$	11.0126	$0\,(0.1169)$, $\frac{57\pi}{200}\,(0.1492)$, $\frac{111\pi}{200}\,(0.1355)$, $\frac{3\pi}{4}\,(0.1568)$	$4 \sim 5$
	$d = 4$	48.0864	$0\,(0.1030)$, $\frac{87\pi}{400}\,(0.0995)$, $\frac{171\pi}{400}\,(0.1273)$, $\frac{249\pi}{400}\,(0.0055)$, $\frac{63\pi}{100}\,(0.1108)$, $\frac{3\pi}{4}\,(0.1055)$	$5 \sim 6$

3.3 K-optimal designs for second-order response models

Consider the second-order response model with d variables,

$$y_i = \theta_0 + \sum_{j=1}^{d} \theta_{jj} x_{ij}^2 + \sum_{j=1}^{d} \theta_j x_{ij} + \sum_{1 \leq j < l < d} \theta_{jl} x_{ij} x_{il} + \epsilon_i, \quad i = 1, \ldots, n, \tag{3.3}$$

where x_{ij} is the ith design point for design variable x_j, $j = 1, \ldots, d$ and $d \geq 1$. The model includes d linear regressors x_j, d quadratic regressors x_j^2, $\frac{d(d-1)}{2}$ interactions $x_j x_l$, and a total of $\frac{(d+1)(d+2)}{2}$ unknown parameters that are expressed as $\boldsymbol{\theta}^\top = \Big(\theta_0, \theta_{11}, \ldots, \theta_{dd}, \theta_1, \ldots, \theta_d, \theta_{12}, \ldots, \theta_{(d-1)(d)}\Big)$. Consider dividing the interval $[-a, a]$ into r equally spaced points. These points are expressed as the set $P = \{p_1, p_2, \ldots, p_r\}$ where $p_h = -a + \frac{2a(h-1)}{r-1}$, $h = 1, \ldots, r$. The d dimensional discrete design space S_N is formed as the Cartesian product of P repeated d times. Each element in S_N is a vector $\boldsymbol{u_i} = (u_{i1} \ldots, u_{id})^\top, u_{ij} \in P$. Therefore, S_N contains all possible combinations of d dimensional points where each coordinate comes from P and the total number of design points in S_N is $N = r^d$. The information matrix $\boldsymbol{A(w)}$ is a square matrix of size $\frac{(d+1)(d+2)}{2} \times \frac{(d+1)(d+2)}{2}$, which is defined in (2.6), with $\boldsymbol{f(u_i)}^\top = \Big(1, u_{i1}^2, \ldots, u_{id}^2, u_{i1}, \ldots, u_{id}, u_{i1}u_{i2}, \ldots, u_{i(d-1)}u_{id}\Big)$.

Yue et al.(2023) presented interesting findings on the K-optimal design for the second-order response model with $d = 2$. They only focused on symmetric designs, as Theorem 3 in their paper shows that symmetric K-optimal designs always have a condition number that is as good as or better than asymmetric designs. This result also applies to models with $d \geq 3$. Therefore, we only consider symmetric designs for the model (3.3) with $d = 3$. Following the steps outlined in Section 2.5 to find the K-optimal designs in MATLAB for second-order response models. A threshold of 10^{-5} is set for all designs to obtain consistent comparisons. In the first scenario, we are interested in the model with all interaction terms included. Examples of K-optimal design in the design space $S = [-1, 1]^3$ with total design points $N = 11^3$ and $N = 13^3$ are given in Table 3.7. All support points with one zero coordinate share the same weight, those with two zero coordinates share another weight, and the center point (0,0,0) has the highest weight among all design points. The condition number is the same for both cases as well, and the value is 8. Both designs have the same support points, but the weights are slightly different. Since the information matrix follows the same structure as explained below, the optimal support points remain unchanged. The small differences in weights are likely due to numerical rounding caused by the change in the number of candidate design points. The form of the information matrix is the same for both cases and follows the structure described in Yue et al.(2023), as outlined below,

$$\mathbf{A}(\mathbf{w}^*) = \begin{bmatrix} 1 & b_1 & b_1 & b_1 \\ b_1 & b_2 & b_3 & b_3 \\ b_1 & b_3 & b_2 & b_3 \\ b_1 & b_3 & b_3 & b_2 \end{bmatrix} \oplus \operatorname{diag}\left(b_1, b_1, b_1, b_3, b_3, b_3\right),$$

where $b_1 = b_2 = 0.4$ and $b_3 = 0.2$. Figure 3.4 is a 3D plot of the K-optimal design for the second-order response model with $d = 3$. Support points with the same weight are shown in the same color and the circle size is proportional to the weight assigned to each design point, where larger circles indicate higher weights.

The second scenario is for second-order response models with $d = 3$, but only with one interaction term x_1x_2. Examples of K-optimal design in the design space $[-1, 1]^3$ with total design points $N = 11^3$ and $N = 13^3$ are given in Table 3.8. Like in the first scenario, the support points of the designs for both cases are the same, while their weights are slightly different. The design remains symmetric about the origin $(0, 0, 0)$; however, the weights of the support points are different depending on which coordinate plane or axis they are on. Specifically, support points with one zero coordinate where those in the x_1x_2 plane, such as

$(\pm1, \pm1, 0)$, receive higher weights than those in the x_1x_3 or x_2x_3 planes, such as $(\pm1, 0, \pm1)$ or $(0, \pm1, \pm1)$. For support points with two zero coordinates, those lying on the x_1 or x_2-axes, such as $(\pm1, 0, 0)$ and $(0, \pm1, 0)$, receive higher weights than those on the x_3-axis, such as $(0, 0, \pm1)$. This indicates that the algorithm gives more weight to the design points that provide more information about the parameters. The condition number is 7.7727 for both cases. The information matrix for the designs for both cases is the same and has the form as follows,

$$\mathbf{A}(\mathbf{w}^*) = \begin{bmatrix} 1 & b_1 & b_1 & b_1' \\ b_1 & b_2 & b_3 & b_3' \\ b_1 & b_3 & b_2 & b_3' \\ b_1' & b_3' & b_3' & b_2' \end{bmatrix} \oplus \operatorname{diag}\left(b_1, b_1, b_1', b_3\right),$$

where $b_1 = \sum_{i=1}^N w_i u_{i1}^2 = \sum_{i=1}^N w_i u_{i2}^2 = 0.3837, b_1' = \sum_{i=1}^N w_i u_{i3}^2 = 0.3134, b_2 = \sum_{i=1}^N w_i u_{i1}^4 = \sum_{i=1}^N w_i u_{i2}^4 = 0.3837, b_2' = \sum_{i=1}^N w_i u_{i3}^4 = 0.3134$, and $b_3 = \sum_{i=1}^N w_i u_{i1}^2 u_{i2}^2 = 0.1918, b_3' = \sum_{i=1}^N w_i u_{i1}^2 u_{i3}^2 = \sum_{i=1}^N w_i u_{i2}^2 u_{i3}^2 = 0.1481$. The entries related to x_3 in the information matrix are smaller than those related to x_1 or x_2, which is consistent with the design that places more weight on the support points involving x_1 or x_2. The same pattern observed with the interaction x_1x_2 also applies when the model includes only x_1x_3 or x_2x_3. In each case, the design places more weight on support points that involve the variables in the interaction term, and this pattern is also reflected in the structure of the information matrix. Figure 3.5 is a 3D plot of the K-optimal design for the second-order response model with $d = 3$, including only the interaction x_1x_2, over the design space $[-1, 1]^3$ with the design points $N = 11^3$. Support points with equal weight are colored the same, while the circle size changes with weight, where larger circles mean higher weights.

The third scenario considers second-order response models with $d = 3$ that include only two interaction terms x_1x_2 and x_1x_3. Examples of K-optimal design in the design space $[-1, 1]^3$ with total design points $N = 11^3$ and $N = 13^3$ are given in Table 3.9. The support points are the same in both designs, but their weights differ depending on N and on how relevant each point is to the interaction terms included in the model. The designs remain symmetric around the origin $(0, 0, 0)$, and more weight is given to points that involve x_1 because it appears in both interaction terms. For example, for one-zero coordinate points, $(\pm 1, \pm 1, 0)$ and $(\pm 1, 0, \pm 1)$, which lie in the $x_1 x_2$ and $x_1 x_3$ planes respectively, receive more weights than other points like $(0, \pm 1, \pm 1)$, which lie in the $x_2 x_3$ plane where $x_1 = 0$. This pattern also appears at points where two of the coordinates are zero. For example, the points $(\pm 1, 0, 0)$ where $x_1 \neq 0$ are more weighted than $(0, \pm 1, 0)$ and $(0, 0, \pm 1)$ where $x_1 = 0$. This shows that the model has more focus on x_1 because it is shared between both interaction terms. The condition number for both cases is 7.8990. The information matrix for the designs for both cases is the same and has the form as follows,

$$\mathbf{A}(\mathbf{w}^*) = \begin{bmatrix} 1 & b_1' & b_1 & b_1 \\ b_1' & b_2' & b_3' & b_3' \\ b_1 & b_3' & b_2 & b_3 \\ b_1 & b_3' & b_3 & b_2 \end{bmatrix} \oplus \operatorname{diag}\left(b_1', b_1, b_1, b_3', b_3'\right),$$

where $b_1 = \sum_{i=1}^N w_i u_{i2}^2 = \sum_{i=1}^N w_i u_{i3}^2 = 0.3603, b_1' = \sum_{i=1}^N w_i u_{i1}^2 = 0.4413, b_2 = \sum_{i=1}^N w_i u_{i2}^4 = \sum_{i=1}^N w_i u_{i3}^4 = 0.3603, b_2' = \sum_{i=1}^N w_i u_{i1}^4 = 0.4413$, and $b_3 = \sum_{i=1}^N w_i u_{i2}^2 u_{i3}^2 = 0.1620, b_3' = \sum_{i=1}^N w_i u_{i1}^2 u_{i2}^2 = \sum_{i=1}^N w_i u_{i1}^2 u_{i3}^2 = 0.1984$. Because both interaction terms involve x_1, design points where $x_1 = 0$ contribute less to the estimation of the model, resulting in smaller weights. This weighting pattern is directly reflected in the information matrix, where the entries are larger because the higher contribution from points with $x_1 \neq 0$, compared to those with $x_1 = 0$. The same pattern seen with two interactions $x_1 x_2$ and $x_1 x_3$ also occurs when the model includes only $x_1 x_2$ and $x_2 x_3$ or $x_1 x_3$ and $x_2 x_3$. In each case, the design places more weight on points involving the variable common to both interaction terms, and this is also reflected in the information matrix. Figure 3.6 is a 3D plot of the K-optimal design for the second-order response model with $d = 3$, including two interaction terms x_1x_2 and x_1x_3 when $N = 11^3$ on $[-1, 1]^3$. Support points that have the same weight are given the same color, and the size of each circle reflects its weight, with larger circles showing higher weights.

Table 3.7: K-optimal design for second-order response models with $d = 3$ on $[-1, 1]^3$ with total design points $N = 11^3$ and $N = 13^3$. Notation $(\pm1, \pm1, \pm1)$: 0.0131 means that there are eight design points and each point has a weight 0.0131.

Total design point	K-optimal design (Support points): weight	Condition number	Information matrix
$N = 11^3$	$(\pm1, \pm1, \pm1) : 0.0131$ $(0, \pm1, \pm1) : 0.0239$ $(\pm1, 0, \pm1) : 0.0239$ $(\pm1, \pm1, 0) : 0.0239$ $(0, 0, \pm1) : 0.0523$ $(0, \pm1, 0) : 0.0523$ $(\pm1, 0, 0) : 0.0523$ $(0, 0, 0) : 0.2955$	8	$\begin{bmatrix} 1.0000 & 0.4000 & 0.4000 & 0.4000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4000 & 0.4000 & 0.2000 & 0.2000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4000 & 0.2000 & 0.4000 & 0.2000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4000 & 0.2000 & 0.2000 & 0.4000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.4000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.4000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.4000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2000 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2000 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2000 \end{bmatrix}$
$N = 13^3$	$(\pm1, \pm1, \pm1) : 0.0160$ $(0, \pm1, \pm1) : 0.0180$ $(\pm1, 0, \pm1) : 0.0180$ $(\pm1, \pm1, 0) : 0.0180$ $(0, 0, \pm1) : 0.0640$ $(0, \pm1, 0) : 0.0640$ $(\pm1, 0, 0) : 0.0640$ $(0, 0, 0) : 0.2719$	8	$\begin{bmatrix} 1.0000 & 0.4000 & 0.4000 & 0.4000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4000 & 0.4000 & 0.2000 & 0.2000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4000 & 0.2000 & 0.4000 & 0.2000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.4000 & 0.2000 & 0.2000 & 0.4000 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.4000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.4000 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.4000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2000 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2000 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2000 \end{bmatrix}$

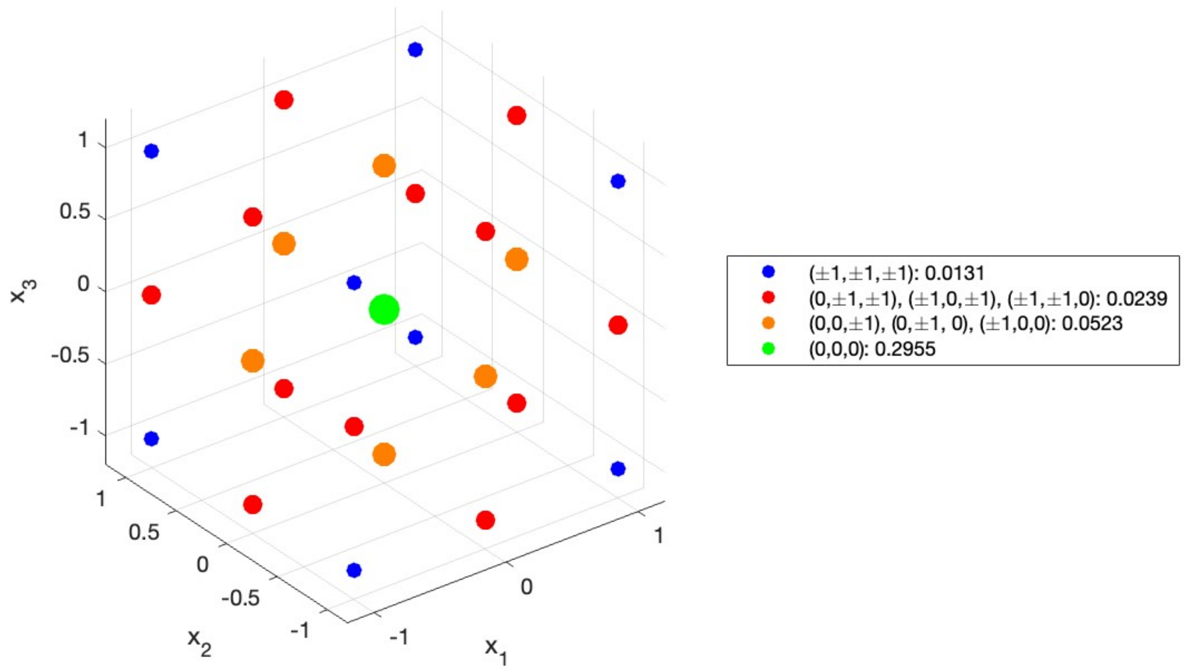


Figure 3.4: Plot of the K-optimal design for the second-order response model with $d = 3$ and $N = 11^3$ on $[-1, 1]^3$. The design points with the same weight are in the same color. Notation $(\pm 1, \pm 1, \pm 1)$: 0.0131 means that there are eight design points and each point has a weight of 0.0131.

Table 3.8: K-optimal design for second-order response models with $d = 3$, including only one interaction term x_1x_2, are constructed on the domain $[-1, 1]^3$ with design points $N = 11^3$ and $N = 13^3$. Notation $(\pm1, \pm1, \pm1)$: 0.0092 means that there are eight design points and each point has a weight 0.0092.

Total design point	K-optimal design (Support points): weight	Condition number	Information matrix
$N = 11^3$	$(\pm1, \pm1, \pm1) : 0.0092$ $(0, \pm1, \pm1) : 0.0187$ $(\pm1, 0, \pm1) : 0.0187$ $(\pm1, \pm1, 0) : 0.0296$ $(0, 0, \pm1) : 0.0453$ $(0, \pm1, 0) : 0.0586$ $(\pm1, 0, 0) : 0.0586$ $(0, 0, 0) : 0.3339$	7.7727	$\begin{bmatrix} 1.0000 & 0.3837 & 0.3837 & 0.3134 & 0 & 0 & 0 & 0 \\ 0.3837 & 0.3837 & 0.1918 & 0.1481 & 0 & 0 & 0 & 0 \\ 0.3837 & 0.1918 & 0.3837 & 0.1481 & 0 & 0 & 0 & 0 \\ 0.3134 & 0.1481 & 0.1481 & 0.3134 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.3837 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.3837 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.3134 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1918 \end{bmatrix}$
$N = 13^3$	$(\pm1, \pm1, \pm1) : 0.0086$ $(0, \pm1, \pm1) : 0.0198$ $(\pm1, 0, \pm1) : 0.0198$ $(\pm1, \pm1, 0) : 0.0307$ $(0, 0, \pm1) : 0.0431$ $(0, \pm1, 0) : 0.0564$ $(\pm1, 0, 0) : 0.0564$ $(0, 0, 0) : 0.3384$	7.7727	$\begin{bmatrix} 1.0000 & 0.3837 & 0.3837 & 0.3134 & 0 & 0 & 0 & 0 \\ 0.3837 & 0.3837 & 0.1918 & 0.1481 & 0 & 0 & 0 & 0 \\ 0.3837 & 0.1918 & 0.3837 & 0.1481 & 0 & 0 & 0 & 0 \\ 0.3134 & 0.1481 & 0.1481 & 0.3134 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.3837 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.3837 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.3134 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1918 \end{bmatrix}$

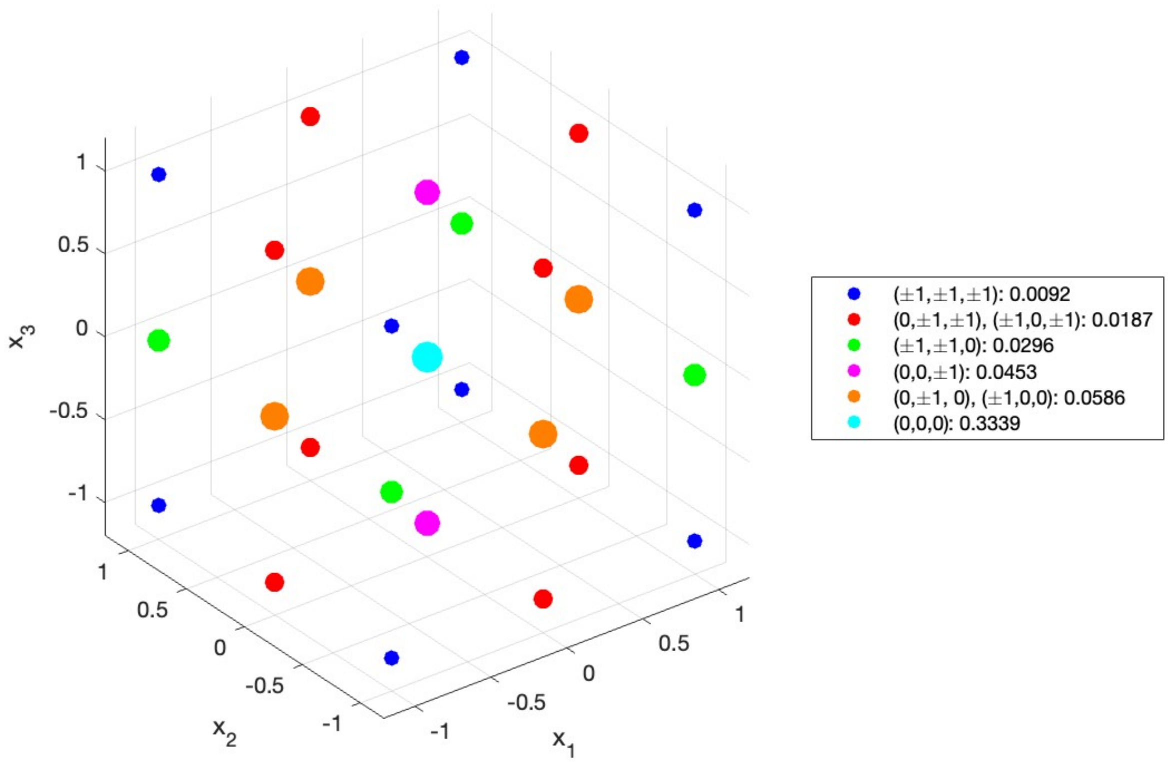


Figure 3.5: Plot of the K-optimal design for the second-order response model with $d = 3$, one interaction term x_1x_2 and $N = 11^3$ in $[-1, 1]^3$. The design points with the same weight are in the same color. Notation $(\pm1, \pm1, \pm1)$: 0.0092 means that there are eight design points and each point has a weight of 0.0092.

Table 3.9: K-optimal design for second-order response models with $d = 3$, including two interaction terms x_1x_2 and x_1x_3, are constructed on the domain $[-1, 1]$ with design points $N = 11^3$ and $N = 13^3$. Notation $(\pm1, \pm1, \pm1)$: 0.0121 means that there are eight design points and each point has a weight 0.0121.

Total design point	K-optimal design (Support points): weight	Condition number	Information matrix
$N = 11^3$	$(\pm1, \pm1, \pm1) : 0.0121$ $(0, \pm1, \pm1) : 0.0162$ $(\pm1, 0, \pm1) : 0.0253$ $(\pm1, \pm1, 0) : 0.0253$ $(0, 0, \pm1) : 0.0485$ $(0, \pm1, 0) : 0.0485$ $(\pm1, 0, 0) : 0.0708$ $(0, 0, 0) : 0.2996$	7.8990	$\begin{bmatrix} 1.0000 & 0.4413 & 0.3603 & 0.3603 & 0 & 0 & 0 & 0 & 0 \\ 0.4413 & 0.4413 & 0.1984 & 0.1984 & 0 & 0 & 0 & 0 & 0 \\ 0.3603 & 0.1984 & 0.3603 & 0.1620 & 0 & 0 & 0 & 0 & 0 \\ 0.3603 & 0.1984 & 0.1620 & 0.3603 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.4413 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.3603 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.3603 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1984 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1984 \end{bmatrix}$
$N = 13^3$	$(\pm1, \pm1, \pm1) : 0.0108$ $(0, \pm1, \pm1) : 0.0189$ $(\pm1, 0, \pm1) : 0.0280$ $(\pm1, \pm1, 0) : 0.0280$ $(0, 0, \pm1) : 0.0432$ $(0, \pm1, 0) : 0.0432$ $(\pm1, 0, 0) : 0.0654$ $(0, 0, 0) : 0.3105$	7.8990	$\begin{bmatrix} 1.0000 & 0.4413 & 0.3603 & 0.3603 & 0 & 0 & 0 & 0 & 0 \\ 0.4413 & 0.4413 & 0.1984 & 0.1984 & 0 & 0 & 0 & 0 & 0 \\ 0.3603 & 0.1984 & 0.3603 & 0.1620 & 0 & 0 & 0 & 0 & 0 \\ 0.3603 & 0.1984 & 0.1620 & 0.3603 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.4413 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.3603 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.3603 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1984 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.1984 \end{bmatrix}$

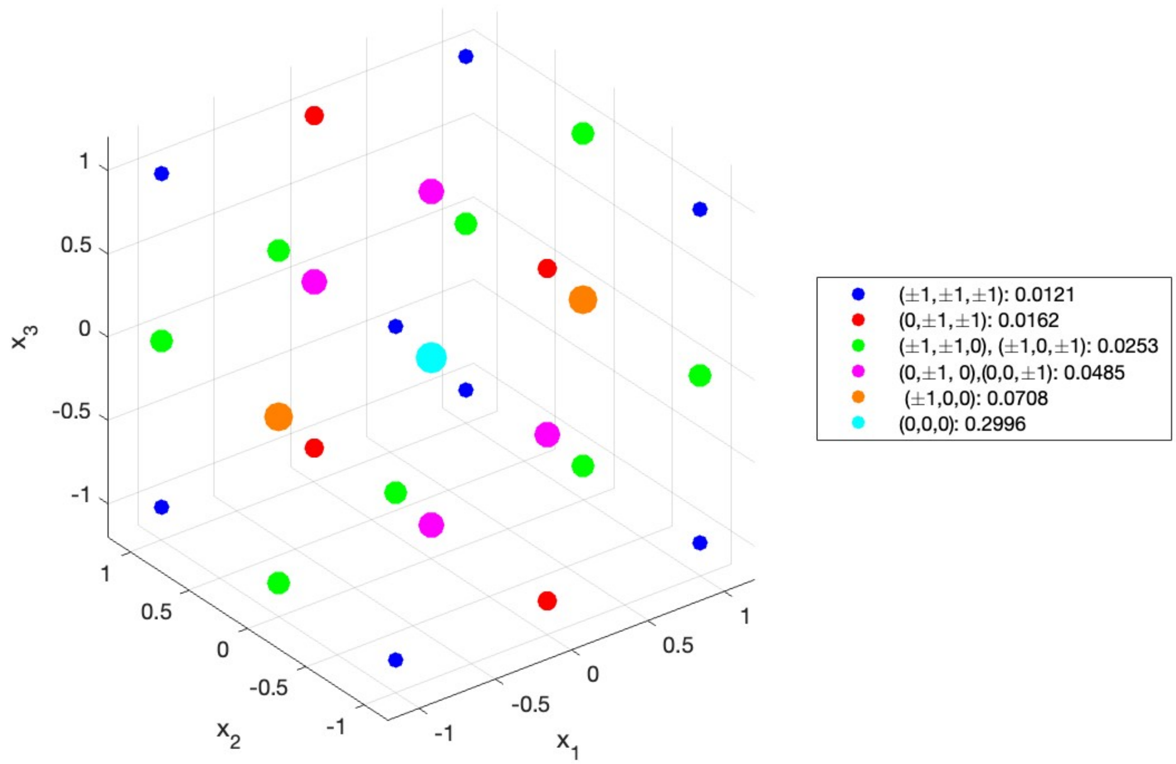


Figure 3.6: Plot of the K-optimal design for the second-order response model with $d = 3$, two interaction terms x_1x_2 and x_1x_3, and $N = 11^3$ in $[-1, 1]^3$. The design points with the same weight are in the same color. Notation $(\pm1, \pm1, \pm1)$: 0.0121 means that there are eight design points and each point has a weight of 0.0121.

Chapter 4

Computing Locally K-optimal Designs for Nonlinear Models

In this chapter, we consider nonlinear regression models to construct locally K-optimal designs. Model (2.1) becomes

$$y_i = \boldsymbol{\eta}\left(x_i, \boldsymbol{\theta}\right) + \epsilon_i, \quad i = 1, \cdots, n, \tag{4.1}$$

where $\boldsymbol{\eta}\left(x_i, \boldsymbol{\theta}\right)$ is a nonlinear function of model parameters $\boldsymbol{\theta}$. The least squares estimator is used to estimate the unknown parameters $\boldsymbol{\theta}$ and defined as

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} \left(y_i - \boldsymbol{\eta}\left(x_i, \boldsymbol{\theta}\right)\right)^2.$$

The covariance matrix of $\hat{\boldsymbol{\theta}}$ is approximately given by

$$\mathrm{Cov}(\hat{\boldsymbol{\theta}}) = \frac{\sigma^2}{n}\left(n^{-1}\sum_{i=1}^{n}\left(\nabla\boldsymbol{\eta}\left(x_i, \boldsymbol{\theta}\right)\right)\left(\nabla\boldsymbol{\eta}\left(x_i, \boldsymbol{\theta}\right)\right)^\top\right)^{-1} = \sigma^2\left(\sum_{i=1}^{n}\left(\nabla\boldsymbol{\eta}\left(x_i, \boldsymbol{\theta}\right)\right)\left(\nabla\boldsymbol{\eta}\left(x_i, \boldsymbol{\theta}\right)\right)^\top\right)^{-1},$$

where $\nabla\boldsymbol{\eta}(x_i,\boldsymbol{\theta})$ is a vector of the partial derivatives of $\boldsymbol{\eta}(x_i,\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta} = (\theta_1,\theta_2,\ldots,\theta_q)$. Therefore, the Fisher information matrix is

$$\boldsymbol{A} = \frac{1}{n}\sum_{i=1}^{n}\left(\nabla\boldsymbol{\eta}(x_i,\boldsymbol{\theta})\right)\left(\nabla\boldsymbol{\eta}(x_i,\boldsymbol{\theta})\right)^{\top}.$$

If an approximate design on a discrete design space $S_N = \{u_1,\ldots,u_N\}$ is used, the Fisher information matrix for nonlinear model becomes

$$\boldsymbol{A}(\boldsymbol{w}) = \sum_{i=1}^{N} w_i\left(\nabla\boldsymbol{\eta}(u_i,\boldsymbol{\theta})\right)\left(\nabla\boldsymbol{\eta}(u_i,\boldsymbol{\theta})\right)^{\top}. \tag{4.2}$$

It can be seen that $\boldsymbol{A}(\boldsymbol{w})$ in (4.2) depends on unknown parameters $\boldsymbol{\theta}$. Therefore, the optimal design also depends on the unknown parameters. This type of optimal design is called locally optimal (Atkinson et al., 2007).

In this chapter, we compute locally K-optimal designs for various nonlinear models, including the Michaelis-Menten model, the compartmental model, and the Peleg model. We also include the generalized linear model, specifically the logistic regression model. We present and discuss the interesting features and results of locally K-optimal designs. Representative MATLAB code is provided in the Appendix.

4.1 Locally K-optimal designs for Michaelis-Menten model

The nonlinear Michaelis-Menten (MM-) model is widely used in enzyme kinetic studies. It describes how the rate of an enzyme-catalyzed reaction depends on the concentration of the enzyme and its substrate. In 1903, physical chemist Victor Henri discovered that enzyme reactions could be explained by the interaction between the enzyme and the substrate (Tomczak and Węglarz-Tomczak, 2019). Leonor Michaelis and Maud Menten further developed Victor's work, and in 1913, they proposed a mathematical model to describe how the reaction rate depends on enzyme and substrate concentrations (Michaelis and Menten, 1913). The model is defined as follows,

$$y_i = \frac{\theta_1 x_i}{\theta_2 + x_i} + \epsilon_i, \quad i = 1, \ldots, n, \tag{4.3}$$

where y_i is the reaction velocity, x_i is the substrate concentration with $x_i \geq 0$, θ_1 is the maximum velocity achieved by the system, and θ_2 is the Michaelis-Menten constant, which represents the value of x_i at which the reaction rate is half the maximum velocity. Both θ_1 and θ_2 are positive. Several optimal designs have been developed for the MM-model. For example, Kamruzzaman (2011) applied three different types of optimal designs based on D-optimal criteria for fixed-effects and mixed-effects MM-model. Dette and Wong (1999) constructed locally E-optimal designs for the MM-model. They provided examples and compared the performance of the E-optimal designs with the D-optimal designs for the MM-model.

In this thesis, we construct locally K-optimal designs for the MM-model and discuss the key features and findings. An equally spaced grid of points is used to construct a discrete design space, $S_N = \{u_1, \ldots, u_N\}$, where $u_i = a + \frac{i-1}{N-1}(b-a)$, $i = 1, \ldots, N$. As $x_i \geq 0$, the design space is defined as $S = [a, b]$, where $a = 0$ and $b > 0$. The information matrix $\boldsymbol{A}(\boldsymbol{w})$ is a square matrix of size 2×2, as defined in (4.2), with $\nabla \boldsymbol{\eta}\left(u_i, \boldsymbol{\theta}\right) = \left(\frac{u_i}{\theta_2 + u_i}, \frac{-\theta_1 u_i}{(\theta_2 + u_i)^2}\right)^\top$. Following the algorithm in Section 2.5 with $\boldsymbol{f}(u_i) = \nabla \boldsymbol{\eta}\left(u_i, \boldsymbol{\theta}\right)$, we compute locally K-optimal designs for the MM-model. We will analyze two cases. The first scenario is where $\boldsymbol{\theta} = [1, 1]$ and b only takes integer values. For consistent comparisons, the sequence u_i in the design space $[0, b]$ for $b > 1$ includes all the points from the previous design space $[0, b-1]$ and keeps the same step size of 0.01, so the total number of points is $N = 100b + 1$.

Table 4.1 shows the results for various design spaces. Each locally K-optimal design for $b = 1, \ldots, 5$ has two support points, which are the first nonzero value of u_i and the value of b. Most design weights are concentrated at the first nonzero value of u_i. The condition number of the locally K-optimal design decreases as the design space expands. Each design has been run 100 times, with an average running time of $1 \sim 2$ seconds. The running time remains unchanged as the design spaces vary. Figure 4.1 shows that for $b = 1, \ldots, 50$ and $N = 100b + 1$, the condition number of the locally K-optimal design decreases as the design space increases for the MM-model, which is consistent with the results in Table 4.1.

The parameters for the second case are adopted from Dette and Wong (1999). The design space is $S = [0, 200]$, with $N = 1001$. Six sets of values for $\boldsymbol{\theta}$ are compared in Table 4.2 for various optimal designs. We first examine the K-optimal design in detail. Each locally K-optimal design has two support points, which are the first nonzero value of u_i and the value of b. For the first four sets of $\boldsymbol{\theta}$, when $\theta_2 = 1$, the design weight for the first nonzero value of u_i decreases significantly as θ_1 increases. Furthermore, the condition number first decreases and then increases as θ_1 increases, which is consistent with the graph (b) of Figure 4.2. For the last three sets of $\boldsymbol{\theta}$, when $\theta_1 = 100$, the design weight for the first nonzero value of u_i increases dramatically as θ_2 increases. Furthermore, the condition number increases with θ_2, as is also shown in graph (a) of Figure 4.2. Each K-optimal design has been run 100 times, with an average running time of $1 \sim 2$ seconds, which remains unchanged with varies $\boldsymbol{\theta}$. Table 4.2 also summarizes the design, the corresponding condition number, and the average running time based on 100 runs for each design under the D- and A-optimality criteria, which are provided for comparison with the K-optimality results for the MM-model. The locally K-optimal design achieves the smallest condition number compared to the D-optimality and A-optimality. For example, when $\boldsymbol{\theta} = [100, 1]$, its condition number is 2.6774, compared to 645.0810 for the D-optimality and 28.7967 for the A-optimality. Additionally, it requires significantly less running time, completing within 1 to 2 seconds in all cases. In contrast, the locally D-optimal design takes 30 to 50 seconds, and the A-optimal design requires 20 to 40 seconds depending on the scenario.

Table 4.1: Locally K-optimal design for Michaelis-Menten models where $\boldsymbol{\theta} = [1, 1]$ on design space $S = [0, b]$.

Design space	N	Locally K-optimal design [support point, weight]	Condition number	Running time (s)
$[0, 1]$	101	$\begin{bmatrix} 0.0100 & 0.9994 \\ 1.0000 & 0.0006 \end{bmatrix}$	39.1968	$1 \sim 2$
$[0, 2]$	201	$\begin{bmatrix} 0.0100 & 0.9996 \\ 2.0000 & 0.0004 \end{bmatrix}$	18.3500	$1 \sim 2$
$[0, 3]$	301	$\begin{bmatrix} 0.0100 & 0.9997 \\ 3.0000 & 0.0003 \end{bmatrix}$	13.2900	$1 \sim 2$
$[0, 4]$	401	$\begin{bmatrix} 0.0100 & 0.9997 \\ 4.0000 & 0.0003 \end{bmatrix}$	11.1065	$1 \sim 2$
$[0, 5]$	501	$\begin{bmatrix} 0.0100 & 0.9997 \\ 5.0000 & 0.0003 \end{bmatrix}$	9.9060	$1 \sim 2$

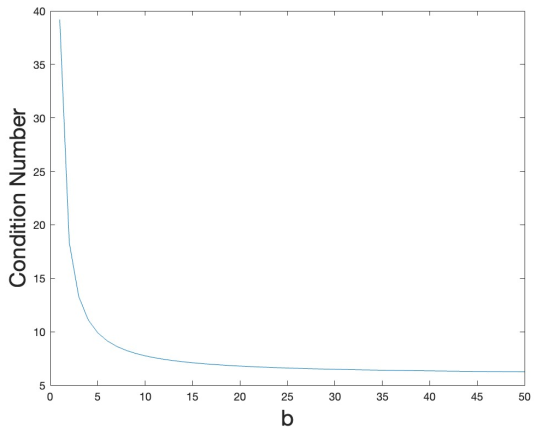


Figure 4.1: Plot of the condition number of locally K-optimal designs versus b for Michaelis-Menten model when $\theta = [1, 1]$ and $N = 100b + 1$, where b is the boundary point of design space $S = [0, b]$.

(a) (b)

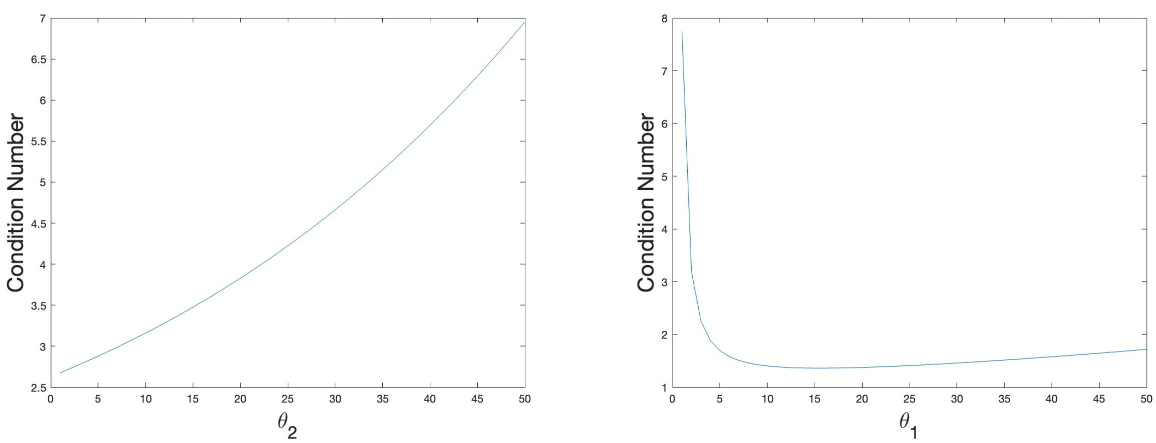


Figure 4.2: Plot of the condition number of locally K-optimal designs versus one of the value of $\boldsymbol{\theta}$, where $N = 1001$ and $S = [0, 200]$, for Michaelis-Menten model: (a) set $\theta_1 = 100$ and $\theta_2 = 1, \ldots, 50$, (b) set $\theta_2 = 1$ and $\theta_1 = 1, \ldots, 50$.

Table 4.2: Various optimal designs for Michaelis-Menten models on $S = [0, 200]$ and $N = 1001$.

$\boldsymbol{\theta}$	Locally K-optimal design			Locally D-optimal design			Locally A-optimal design		
	Design [support point, weight]	Condition number	Running time (s)	Design [support point, weight]	Condition number	Running time (s)	Design [support point, weight]	Condition number	Running time (s)
$[10, 1]$	$\begin{bmatrix} 0.2000 & 0.3365 \\ 200.0000 & 0.6635 \end{bmatrix}$	1.4048	$1 \sim 2$	$\begin{bmatrix} 1.0000 & 0.5000 \\ 200.0000 & 0.5000 \end{bmatrix}$	7.1148	$40 \sim 50$	$\begin{bmatrix} 0.8000 & 0.1801 \\ 1.0000 & 0.1030 \\ 200.0000 & 0.7169 \end{bmatrix}$	2.8614	$20 \sim 30$
$[15, 1]$	$\begin{bmatrix} 0.2000 & 0.1856 \\ 200.0000 & 0.8144 \end{bmatrix}$	1.3632	$1 \sim 2$	$\begin{bmatrix} 1.0000 & 0.5000 \\ 200.0000 & 0.5000 \end{bmatrix}$	15.1049	$40 \sim 50$	$\begin{bmatrix} 0.8000 & 0.0109 \\ 1.0000 & 0.1979 \\ 200.0000 & 0.7912 \end{bmatrix}$	4.0727	$20 \sim 30$
$[50, 1]$	$\begin{bmatrix} 0.2000 & 0.0213 \\ 200.0000 & 0.9787 \end{bmatrix}$	1.7197	$1 \sim 2$	$\begin{bmatrix} 1.0000 & 0.5000 \\ 200.0000 & 0.5000 \end{bmatrix}$	161.6914	$40 \sim 50$	$\begin{bmatrix} 0.8000 & 0.0766 \\ 200.0000 & 0.9234 \end{bmatrix}$	13.2223	$20 \sim 30$
$[100, 1]$	$\begin{bmatrix} 0.2000 & 0.0064 \\ 200.0000 & 0.9936 \end{bmatrix}$	2.6774	$1 \sim 2$	$\begin{bmatrix} 1.0000 & 0.5000 \\ 200.0000 & 0.5000 \end{bmatrix}$	645.0810	$40 \sim 50$	$\begin{bmatrix} 0.8000 & 0.0431 \\ 200.0000 & 0.9569 \end{bmatrix}$	28.7967	$20 \sim 30$
$[100, 10]$	$\begin{bmatrix} 0.2000 & 0.9675 \\ 200.0000 & 0.0325 \end{bmatrix}$	3.1609	$1 \sim 2$	$\begin{bmatrix} 9.0000 & 0.4211 \\ 9.2000 & 0.0789 \\ 200.0000 & 0.5000 \end{bmatrix}$	10.1417	$40 \sim 50$	$\begin{bmatrix} 7.4000 & 0.2983 \\ 200.0000 & 0.7017 \end{bmatrix}$	4.9771	$30 \sim 40$
$[100, 20]$	$\begin{bmatrix} 0.2000 & 0.9975 \\ 200.0000 & 0.0025 \end{bmatrix}$	3.8293	$1 \sim 2$	$\begin{bmatrix} 16.6000 & 0.5000 \\ 200.0000 & 0.5000 \end{bmatrix}$	6.3456	$40 \sim 50$	$\begin{bmatrix} 13.0000 & 0.0001 \\ 13.2000 & 0.4416 \\ 200.0000 & 0.5582 \end{bmatrix}$	5.3476	$20 \sim 30$

4.2 Locally K-optimal designs for compartmental model

The nonlinear compartmental model is a mathematical framework used to analyze complex problems by dividing them into a finite number of interconnected compartments (Enderle and Bronzino, 2012, p.379). The general form of the d-compartmental model is given as

$$\eta(x_i, \boldsymbol{\theta}) = \sum_{j=1}^{d} \theta_j e^{\left(-\theta_{d+j} x_i\right)},$$

where $x_i \geq 0$ and $0 < \theta_{d+1} < \theta_{d+2} < \cdots < \theta_{2d}$. Compartmental models have been applied in various fields. For example, Brauer (2008) used them in epidemiology to model disease transmission. In pharmacokinetics, they are the most commonly used modeling technique (Chaubet et al., 2019, p.440). Saghir et al. (2024) highlighted their use in understanding and predicting how xenobiotics and their metabolites are processed within the body, where a xenobiotic is a foreign chemical substance, such as a drug. In addition, Pornphol et al. (2024) developed compartmental models in environmental science to capture the dynamics of pollution in various water bodies and solve these models to better understand and manage water pollution. Many studies have also explored optimal designs for compartmental models. For instance, López-Fidalgo et al. (2008) used locally c-optimal designs on compartmental models to determine the optimal times for bioassays following accidental radioactivity exposure. Dette et al. (2006) constructed the locally D-optimal designs for compartmental models when $d = 1, 2$ and 3. Wong and Zhou (2019) extended Dette's work by computing the locally D-optimal designs for $d = 4$.

In this thesis, we focus on $d = 1$ for the locally K-optimal design of compartmental models, as when $d \geq 2$, the designs are challenging to solve numerically. Equally spaced grid points are used to define the discrete design space $S_N = \{u_1, \ldots, u_N\}$, where $u_i = a + \frac{i-1}{N-1}(b-a)$,

$i = 1, \dots, N$. Since $x_i \geq 0$, the design space is $S = [a, b]$ with $a = 0$ and $b > 0$. For $d = 1$, the compartmental model is given as $\eta(x_i, \boldsymbol{\theta}) = \theta_1 e^{(-\theta_2 x_i)}$. The information matrix $\boldsymbol{A}(\boldsymbol{w})$ is a 2×2 square matrix as defined in (4.2), with $\nabla \boldsymbol{\eta}(u_i, \boldsymbol{\theta}) = \left(e^{(-\theta_2 u_i)}, -\theta_1 u_i e^{(-\theta_2 u_i)}\right)^\top$. Using the algorithm in Section 2.5, with $\boldsymbol{f}(u_i) = \nabla \boldsymbol{\eta}(u_i, \boldsymbol{\theta})$, we compute locally K-optimal designs for the compartmental model with $d = 1$. The chosen values of $\boldsymbol{\theta}$ and b are from Wong and Zhou (2019), where $\boldsymbol{\theta} = [1, 0.1]$ and $b = 10$. For comparison, we also test with $\theta_2 = 0.05$. When $N = 51, 101, 201, 501$, and 801, the K-optimal designs remain consistent for each $\boldsymbol{\theta}$, and the results are shown in Table 4.3. The support points of the K-optimal designs for both sets of $\boldsymbol{\theta}$ are the same, 0 and 10, which are the boundary points of the design space $S = [0, 10]$. Most of the design weight is concentrated at the first support point, 0. Furthermore, the design weight at the first support point decreases as θ_2 decreases. Although the condition number remains unchanged as N varies for each $\boldsymbol{\theta}$, it decreases as θ_2 increases. Each design was run 100 times, with an average running time of $1 \sim 2$ seconds on all designs. For each set value of $\boldsymbol{\theta}$, the running time does not change while N and $\boldsymbol{\theta}$ changes. Under the setting of $\theta_1 = 1$, $N = 801$, $b = 10$, and $\theta_2 = 0.01, 0.02, \dots, 0.1$, the support points for each K-optimal design are 0 and 10. Also, Figure 4.3 illustrates that the condition number decreases as θ_2 increases, while the design weight of 0 increases with θ_2. These results are consistent with Table 4.3. Table 4.4 compares the condition number and the average running time of 100 runs for various optimal designs applied to the compartmental model with $d = 1$ in $S = [0, 10]$ and $\boldsymbol{\theta} = [1, 0.1]$, for different values of N. The results show that the locally K-optimal design achieves the smallest condition number and requires less running time compared to the locally D-optimal and A-optimal designs. The condition numbers of the K-, D-, and A-optimal designs do not change with N. Specifically, the condition number is 1.0274 for K-optimality, 739.1767 for D-optimality, and 27.1907 for A-optimality. However, the running times of all three designs increase as N increases.

Table 4.3: Locally K-optimal design for compartmental models where $d = 1$ on $S = [0, 10]$. The same designs were obtained for each $\boldsymbol{\theta}$ for various values of N, $N = 51, 101, 201, 501$ and 801.

$\boldsymbol{\theta}$	Locally K-optimal design [support point, weight]	Condition number	Running time (s)
$[1, 0.1]$	$\begin{bmatrix} 0 & 0.9986 \\ 10.0000 & 0.0014 \end{bmatrix}$	1.0274	$1 \sim 2$
$[1, 0.05]$	$\begin{bmatrix} 0 & 0.9963 \\ 10.0000 & 0.0037 \end{bmatrix}$	1.0763	$1 \sim 2$

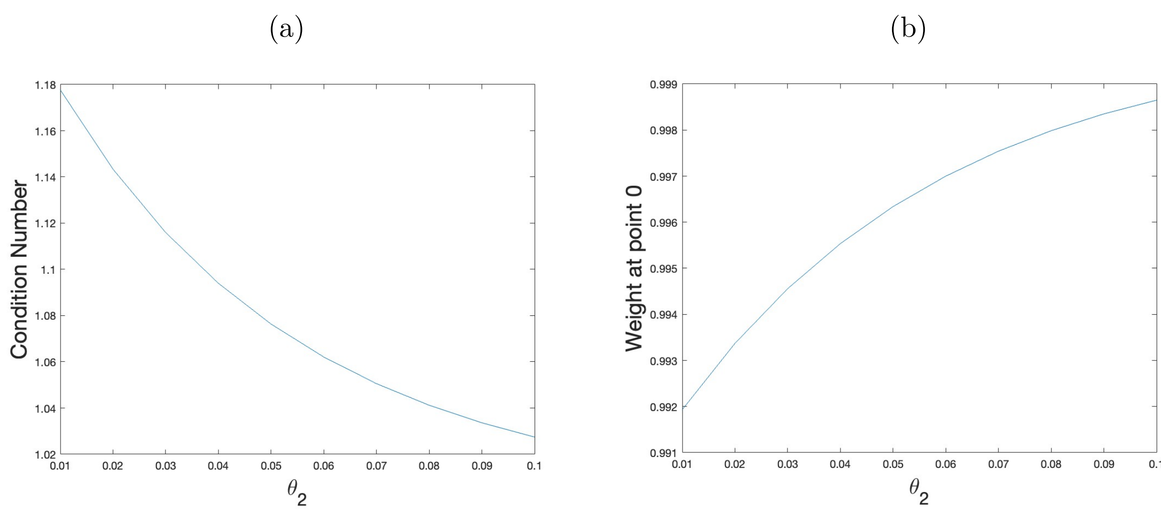


Figure 4.3: (a) plot of condition number versus θ_2, (b) plot of weight of 0 versus θ_2, for the locally K-optimal designs of compartmental model where $d = 1$ on $S = [0, 10]$, $N = 801$ and $\theta_1 = 1$.

Table 4.4: Condition number and average running time of various optimal designs for compartmental models with $d = 1$ on $S = [0, 10]$ and $\boldsymbol{\theta} = [1, 0.1]$.

N	Locally K-optimal design		Locally D-optimal design		Locally A-optimal design	
	Condition number	Running time (s)	Condition number	Running time (s)	Condition number	Running time (s)
51	1.0274	$1 \sim 2$	739.1767	$1 \sim 2$	27.1907	$1 \sim 2$
101	1.0274	$2 \sim 3$	739.1767	$3 \sim 4$	27.1907	$3 \sim 4$
201	1.0274	$2 \sim 3$	739.1767	$4 \sim 5$	27.1907	$5 \sim 6$
501	1.0274	$6 \sim 7$	739.1767	$9 \sim 10$	27.1907	$8 \sim 9$
801	1.0274	$9 \sim 10$	739.1767	$15 \sim 16$	27.1907	$12 \sim 13$

4.3 Locally K-optimal designs for Peleg model

The nonlinear empirical Peleg model is a two-parameter model proposed by Peleg (1988) to describe the sorption curve, representing the relationship between moisture content and time. The model is defined as

$$y_i = y_o + \frac{x_i}{\theta_1 + \theta_2 x_i} + \epsilon_i, \quad i = 1, 2, \cdots, n,$$

where y_i is the moisture content at time $x_i \in S = [0, b]$, y_0 is the initial moisture content, $\theta_1 > 0$ is the Peleg rate constant related to the sorption rate, and $\theta_2 > 0$ is the Peleg capacity constant associated with maximum water absorption. The Peleg model has been widely used to study the hydration behavior of various food products (Montanuci et al., 2013). In addition, it is commonly used to analyze the water absorption of products during soaking, such as cereal grains (Sopade et al., 1992), beans and chickpeas (Shafaei et al., 2016), bambara seeds (Jideani and Mpotokwana, 2009), and black beans (Li et al., 2020). In addition, optimal designs for the Peleg model have been investigated in varying studies. For example, Paquet-Durand et al. (2015) explored different locally optimal designs, including A-, E-, and D-optimality, and compared their performance to ordinary equidistant designs based on parameter estimation errors. Their findings demonstrated that estimation errors for the parameters of the Peleg model could be significantly reduced using locally optimal designs. Furthermore, Paquet-Durand et al. (2015) conducted experiments on the water sorption process in wheat grains, employing locally D-optimal design on a modified Peleg model. Yeh (2018) also examined locally optimal experimental designs for the Peleg model using second-order least squares estimator, including D-, A-, and c-optimality.

The parameter values for the locally K-optimal design in the Peleg model are based on those studied in Yeh (2018). Equally spaced grid points are used to form a discrete design space, $S_N = \{u_1, \ldots, u_N\}$, where $u_i = a + \frac{i-1}{N-1}(b-a)$, $i = 1, \ldots, N$. Since variable time $x_i \geq 0$, the design space is $S = [a, b]$, where $a = 0$ and $b > 0$. The information matrix $\boldsymbol{A}(\boldsymbol{w})$ is a 2×2 square matrix defined in (4.2), with $\nabla \boldsymbol{\eta}\,(u_i, \boldsymbol{\theta}) = \left(\frac{-u_i}{(\theta_1+\theta_2 u_i)^2}, \frac{-(u_i)^2}{(\theta_1+\theta_2 u_i)^2},\right)^\top$. Using the algorithm in Section 2.5, with $\boldsymbol{f}(u_i) = \nabla \boldsymbol{\eta}\,(u_i, \boldsymbol{\theta})$, we computed locally K-optimal designs for the Peleg model. Three different design spaces $S = [0, b]$ were considered, with $b = 120, 180$ and 240, across various sets of $\boldsymbol{\theta}$ values. For consistent comparisons, set $N = b + 1$. The results are shown in Table 4.5, which reveals several interesting findings. The support points of the locally K-optimal design for the Peleg model are the first nonzero design point and the upper boundary point b of the design space. Within the same design space, the design weight of the first nonzero point decreases as θ_2 increases but increases as θ_1 increases. When $\boldsymbol{\theta}$ remains constant, the design weight of the first nonzero point increases as the design space expands. When the design space remains unchanged, the condition number of the locally K-optimal design does not change with variations in $\boldsymbol{\theta}$. However, the condition number decreases as the design space expands. Each design was run 100 times, and the running time was similar for all of them, between 3 and 4 seconds. Therefore, the design space and the value of $\boldsymbol{\theta}$ have minimal impact on computational time. Figure 4.4 illustrates results consistent with Table 4.5. The plot (a) shows that when the design space is fixed at $S = [0, 180]$ and $\theta_1 = 0.5$, the design weight assigned to the first support point decreases as θ_2 increases. Plot (b) presents the opposite scenario with the design space fixed at $S = [0, 180]$ and $\theta_2 = 0.5$, the weight of the first support point increases as θ_1 increases. Plot (c) keeping $\boldsymbol{\theta} = [0.5, 0.5]$, the design weight of the first support point slightly increases as the design space expands. Finally, plot (d) shows that under the same fixed parameter values as plot (c), the condition number decreases when the design space expands. Table 4.6 compares the condition number and the average running time of 100 runs for various optimal designs applied to the Peleg model on $S = [a, b]$ with $a = 0$, $b = 180$, and $N = b + 1$. The locally K-optimal design achieves the smallest condition number and requires less running time compared to the locally D-optimal and A-optimal designs. The average running time ranging between 3 $\sim$ 4 seconds for the K-optimal designs, between 4 $\sim$ 5 seconds for D-optimal designs, and between 3 $\sim$ 5 seconds for A-optimal designs. The condition numbers for D- and A-optimal designs vary with changes in $\boldsymbol{\theta}$, while the condition number for K-optimal designs stays unchanged.

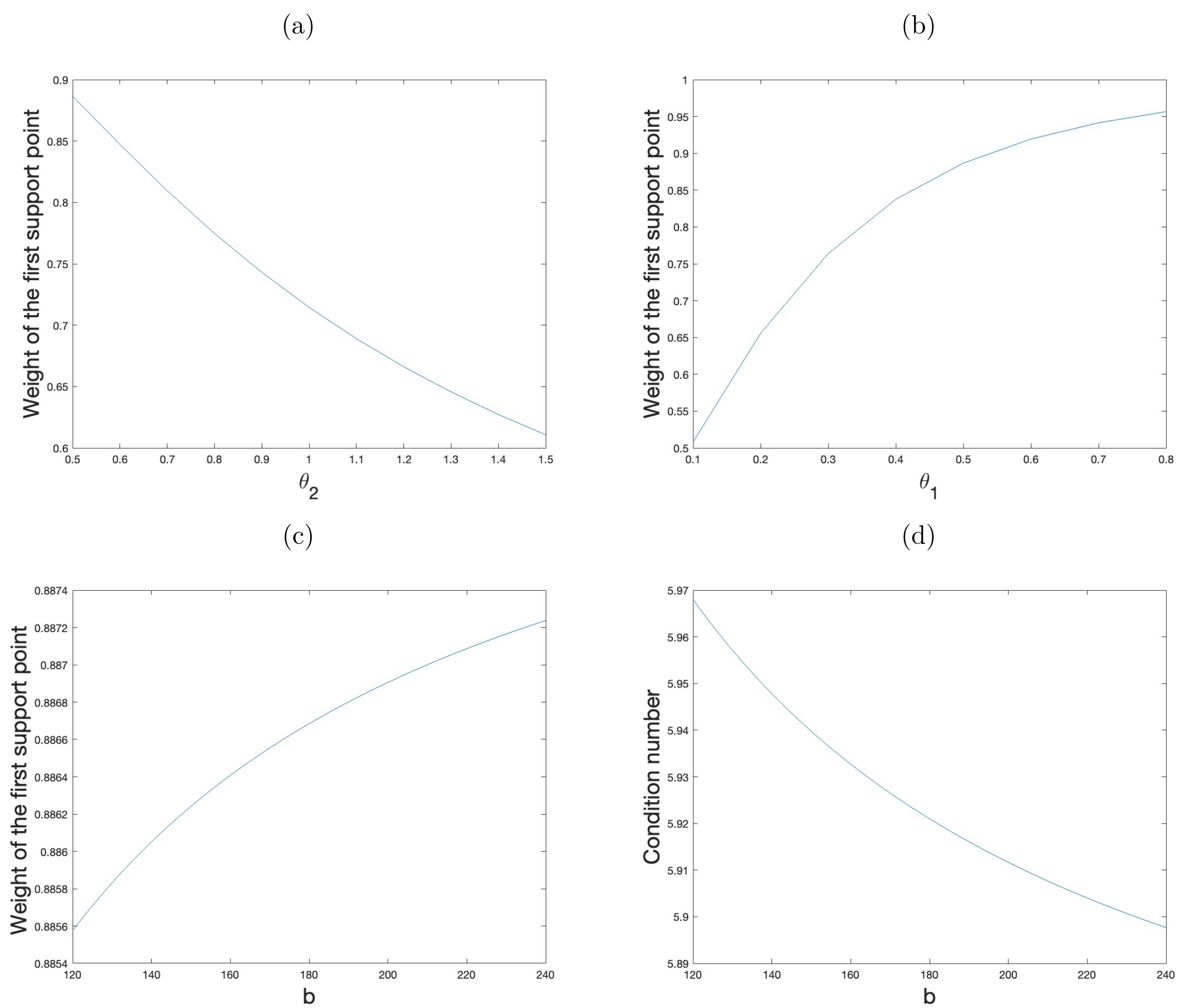


Figure 4.4: Locally K-optimal design on Peleg model: (a) Plot of design weight of the first support point versus θ_2 on $S = [a, b]$ with $a = 0$, $b = 180$, and $N = b + 1$, $\theta_1 = 0.5$ and $\theta_2 = 0.5, 0.6, \ldots, 1.5$. (b) Plot of design weight of the first support point versus θ_1 on $S = [a, b]$ with $a = 0$, $b = 180$, and $N = b + 1$, $\theta_2 = 0.5$ and $\theta_1 = 0.1, 0.2, \ldots, 0.8$. (c) Plot of design weight of the first support point versus b when $\boldsymbol{\theta} = [0.5, 0.5]$, $b = 120, 121, \ldots, 240$, and $N = b + 1$. (d) Plot of condition number versus b when $\boldsymbol{\theta} = [0.5, 0.5]$, $b = 120, 121, \ldots, 240$, and $N = b + 1$.

Table 4.5: Locally K-optimal design and condition number $\kappa\left(\boldsymbol{A}\left(\boldsymbol{w}\right)\right)$ of the design for Peleg models on $S=[a,b]$ with $a=0$, $b=120,180,240$, and $N=b+1$.

$\boldsymbol{\theta}$	Locally K-optimal design on $S=[0,120]$ [support point, weight]	Locally K-optimal design on $S=[0,180]$ [support point, weight]	Locally K-optimal design on $S=[0,240]$ [support point, weight]
$[0.5, 0.5]$	$\begin{bmatrix} 1.0000 & 0.8856 \\ 120.0000 & 0.1144 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.8867 \\ 180.0000 & 0.1133 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.8872 \\ 240.0000 & 0.1128 \end{bmatrix}$
$[0.5, 1]$	$\begin{bmatrix} 1.0000 & 0.7134 \\ 120.0000 & 0.2866 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.7146 \\ 180.0000 & 0.2854 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.7151 \\ 240.0000 & 0.2849 \end{bmatrix}$
$[0.5, 1.5]$	$\begin{bmatrix} 1.0000 & 0.6098 \\ 120.0000 & 0.3902 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.6107 \\ 180.0000 & 0.3893 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.6111 \\ 240.0000 & 0.3889 \end{bmatrix}$
$[0.1, 0.5]$	$\begin{bmatrix} 1.0000 & 0.5074 \\ 120.0000 & 0.4926 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.5079 \\ 180.0000 & 0.4921 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.5082 \\ 240.0000 & 0.4918 \end{bmatrix}$
$[0.8, 0.5]$	$\begin{bmatrix} 1.0000 & 0.9559 \\ 120.0000 & 0.0441 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.9566 \\ 180.0000 & 0.0434 \end{bmatrix}$	$\begin{bmatrix} 1.0000 & 0.9570 \\ 240.0000 & 0.0430 \end{bmatrix}$
$\kappa(\boldsymbol{A}(\boldsymbol{w}))$	5.9680	5.9210	5.8977

Table 4.6: Condition number and average running time of various optimal designs for Peleg models on $S = [a, b]$ with $a = 0$, $b = 180$, and $N = b + 1$.

$\boldsymbol{\theta}$	Locally K-optimal design		Locally D-optimal design		Locally A-optimal design	
	Condition number	Running time (s)	Condition number	Running time (s)	Condition number	Running time (s)
$[0.5, 0.5]$	5.9210	$3 \sim 4$	18.0737	$4 \sim 5$	8.3050	$3 \sim 4$
$[0.5, 1]$	5.9210	$3 \sim 4$	7.7875	$4 \sim 5$	6.3661	$3 \sim 4$
$[0.1, 0.5]$	5.9210	$3 \sim 4$	5.9231	$4 \sim 5$	5.9215	$3 \sim 4$
$[0.8, 0.5]$	5.9210	$3 \sim 4$	50.2856	$4 \sim 5$	11.8895	$4 \sim 5$

4.4 Locally K-optimal designs for logistic regression model

In regression analysis, when the response variable y for a given x does not follow a normal distribution, generalized linear models (GLMs) can be considered to study the relationship between the response variable and regressors. In particular, they are useful when the dependent variable is categorical. GLMs extend linear regression to allow for noncontinuous response variables, such as binary or count data (Rabe-Hesketh and Skrondal, 2010). GLMs use a link function to connect predictors to the response variable (Zhao, 2012). This thesis focuses on a specific type of GLMs, logistic regression. In logistic regression, the response variable y_i has only two possible outcomes, meaning that y_i follows a Bernoulli distribution with probability p instead of a normal distribution (Freund et al., 2010). Since $y_i \in \{0, 1\}$, a linear equation might produce predictions outside the range. To address this issue, we model $p(\boldsymbol{x}, \boldsymbol{\theta}) = p(y = 1)$, where

$$p(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{e^{h(\boldsymbol{x}, \boldsymbol{\theta})}}{1 + e^{h(\boldsymbol{x}, \boldsymbol{\theta})}}, \tag{4.4}$$

where p represents the probability of the response variable taking the value 1, and $h(\boldsymbol{x}, \boldsymbol{\theta})$ being a linear function of $\boldsymbol{\theta}$, i.e., $h(\boldsymbol{x}, \boldsymbol{\theta}) = \theta_0 + \theta_1 x_1 + \ldots + \theta_d x_d$, where $x_1, \ldots, x_d$ are regressors. The function $h(\boldsymbol{x}, \boldsymbol{\theta})$, also known as the logit link function, is expressed as

$$h(\boldsymbol{x}, \boldsymbol{\theta}) = \text{logit}(p(\boldsymbol{x}, \boldsymbol{\theta})) = \log \frac{p(\boldsymbol{x}, \boldsymbol{\theta})}{1 - p(\boldsymbol{x}, \boldsymbol{\theta})}.$$

van den Berg (2018), in Section 14.2 explains how the logit link function transforms the linear combination of explanatory variables into probabilities and constructs the logistic function. Logistic regression uses Maximum Likelihood Estimation to estimate the parameters. Since

y_i has only two possible outcomes, such as 0 and 1, let the probability of $y_i = 1$ be p and $y_i = 0$ be $1 - p$. The likelihood function is given by

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} p(\boldsymbol{x_i}, \boldsymbol{\theta})^{y_i} \left(1 - p(\boldsymbol{x_i}, \boldsymbol{\theta})\right)^{(1-y_i)},$$

where $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_d)^\top$. Taking natural logarithm simplifies maximization, yielding the log-likelihood function given as

$$\begin{aligned} l\left(\boldsymbol{\theta}\right) &= \sum_{i=1}^{n} y_i \log p\left(\boldsymbol{x_i}, \boldsymbol{\theta}\right) + (1 - y_i) \log(1 - p\left(\boldsymbol{x_i}, \boldsymbol{\theta}\right)) \\ &= \sum_{i=1}^{n} y_i \log \frac{p\left(\boldsymbol{x_i}, \boldsymbol{\theta}\right)}{1 - p\left(\boldsymbol{x_i}, \boldsymbol{\theta}\right)} + \sum_{i=1}^{n} \log(1 - p\left(\boldsymbol{x_i}, \boldsymbol{\theta}\right)). \end{aligned} \tag{4.5}$$

The Fisher information matrix for logistic regression is defined as (Nadarajah, 2004)

$$\boldsymbol{A} = E\left(\left(\frac{\partial l(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)\left(\frac{\partial l(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^\top\right),$$

where $E[\cdot]$ denotes the expectation with respect to the Bernoulli distribution of Y, with mean $p(x_i, \boldsymbol{\theta})$. Using equation (4.5), the Fisher information matrix can be expressed as

$$\boldsymbol{A} = \sum_{i=1}^{N} \boldsymbol{w_i} p(\boldsymbol{x_i}, \boldsymbol{\theta})(1 - p(\boldsymbol{x_i}, \boldsymbol{\theta}))\boldsymbol{z_i}\boldsymbol{z_i}^\top, \tag{4.6}$$

where $\boldsymbol{z_i}^\top = (1, x_{1i}, x_{2i}, \ldots, x_{di})$. Since the information matrix depends on unknown parameters, the optimal design for logistic regression is a locally optimal. There are many applications involving optimal designs for logistic regression. For example, Mathew and Sinha (2001) developed locally D- and A-optimal designs to estimate multiple parameter pairs in binary data under the two-parameter logistic regression model. Haines et al. (2007) studied locally D-optimal designs for logistic regression models with two constrained explanatory variables and no interaction terms. Wong and Zhou (2019) constructed locally

D-optimal designs for logistic regression models with two constrained explanatory variables and an interaction term.

We use the logistic model and parameter values in Wong and Zhou (2019) to construct the locally K-optimal design. Considering the logistic model with two variables and an interaction term, model (4.4) becomes

$$p(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{e^{(\theta_0+\theta_1 x_1+\theta_2 x_2+\theta_3 x_1 x_2)}}{1+e^{(\theta_0+\theta_1 x_1+\theta_2 x_2+\theta_3 x_1 x_2)}}, \tag{4.7}$$

where $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_3)^\top$ is the parameter vector, $x_1 \in [a, b_1]$ and $x_2 \in [a, b_2]$ with $a = 0$ and $b_1, b_2 > 0$. The interval $[a, b_1]$ is divided into N_1 equally spaced points and the interval $[a, b_2]$ is divided into N_2 equally spaced points. These sets are expressed as $S_{N_1} = \{u_{11}, \ldots, u_{1N_1}\}$, with $u_{1i} = a + \frac{i-1}{N_1-1}(b_1 - a)$, $i = 1, \ldots, N_1$, and $S_{N_2} = \{u_{21}, \ldots, u_{2N_2}\}$, with $u_{2i} = a + \frac{i-1}{N_2-1}(b_2 - a)$, $i = 1, \ldots, N_2$. The points in S_{N_1} and S_{N_2} can be combined to form a two-dimensional grid of the total points of $N = N_1 N_2$ in the design space S_N. The information matrix of model (4.7) can be derived from equation (4.6) and is given as

$$\boldsymbol{A}(\boldsymbol{w}) = \sum_{i=1}^{N} w_i \frac{e^{\boldsymbol{z_i}^\top \boldsymbol{\theta}}}{(1+e^{\boldsymbol{z_i}^\top \boldsymbol{\theta}})^2} \boldsymbol{z_i} \boldsymbol{z_i}^\top,$$

where $\boldsymbol{z_i}^\top = (1, u_{1i}, u_{2i}, u_{1i}u_{2i})$ and $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_3)^\top$. Using the algorithm in Section 2.5, with $\boldsymbol{f}(u_i) = \frac{\sqrt{e^{z_i^\top \boldsymbol{\theta}}}}{1+e^{z_i^\top \boldsymbol{\theta}}} \boldsymbol{z_i}$, we compute locally K-optimal designs for the logistic model in (4.7). Several scenarios are considered. The first scenario considers $x_1 \in [0, 2]$, $x_2 \in [0, 1]$, and $N = 21^2$ or $N = 51^2$. The two-dimensional design space forms a rectangular shape. Different values of $\boldsymbol{\theta}$ are used based on $\boldsymbol{\theta} = (-2, 3, 2, 1)$ from Wong and Zhou (2019). In each case, only one entry of $\boldsymbol{\theta}$ is changed while the others remain the same. K-optimal designs are shown in Table 4.7 for $N = 21^2$. The running time for each case is similar, ranging between 12 and 14 seconds. Each K-optimal design has four support points, $(0, 0)$, $(0, 1)$, $(2, 0)$, and $(2, 1)$, corresponding to the four corners of the rectangular design space and remain the same for different values of $\boldsymbol{\theta}$. However, the weight of each support point changes with the value of $\boldsymbol{\theta}$. Keeping the other values of $\boldsymbol{\theta}$ fixed, as θ_1 increases, the weights of points $(0,0)$ and $(0,1)$ decrease, while the weights of points $(2,0)$ and $(2,1)$ increase. Similarly, as any one of θ_2, θ_3 or θ_4 increases, the weights of the points $(0,0)$, $(0,1)$, and $(2,0)$ decrease, while the weight of the point $(2,1)$ increases. The condition number of each K-optimal design in Table 4.7 is 15.2590 and does not vary with different values of $\boldsymbol{\theta}$. For all sets of $\boldsymbol{\theta}$ tested in Table 4.7, the K-optimal designs and their condition numbers are the same for $N = 21^2$ and $N = 51^2$. However, the running time for $N = 51^2$ is five to seven times longer than for $N = 21^2$, ranging between 60 and 90 seconds.

The second scenario considers different design spaces for x_1 and x_2, which are given in Table 4.8, with $\boldsymbol{\theta} = (-2, 3, 4, 1)$ and $N = 21^2$ or $N = 51^2$. The same as in the first scenario, for all K-optimal designs in Table 4.8, the designs and their condition numbers are the same for both N values, but the running time for $N = 51^2$ is five to seven times longer than that for $N = 21^2$. The results are shown in Table 4.8 for $N = 21^2$. The running time for each design does not change much when the design spaces vary, ranging between 11 and 13 seconds. The condition number increases as the design space for x_1 becomes small and decreases as the design space for x_2 expands. The two-dimensional design space forms a rectangular or square shape. Each K-optimal design has four support points that correspond to the four corners of the design space.

The third scenario considers a two-dimensional design space constrained to form a triangular, where $x_1 \in [0,1]$, $x_2 \in [0,1]$, and $x_1 + x_2 \leq 1$. With $\boldsymbol{\theta} = (-2, 3, 4, 1)$, the K-optimal design and its condition number are the same for $N = 21^2$ and $N = 51^2$. However, the running time for $N = 51^2$ is five to six times longer than for $N = 21^2$. Table 4.9 presents the results of the locally K-optimal design for $N = 21^2$. For comparison, D-optimality is also evaluated under the same parameter values and design space. The D-optimal design has one additional design point compared to the K-optimal design. It takes longer to generate the

D-optimal design, and its condition number is more than twice that of the K-optimal design. To visualize the designs, Figure 4.5 illustrates both the locally K-optimal and D-optimal designs. The design space is the triangular inside in the dashed lines. The support points are circled in the plot, and their corresponding weights are presented beside them.

This method is quite general and can be used to construct K-optimal designs for other GLMs. The main change needed is in the calculation of the information matrix $\boldsymbol{A}(\boldsymbol{w})$.

Table 4.7: K-optimal designs for the logistic model in (4.7) where $x_1 \in [0, 2]$, $x_2 \in [0, 1]$ and $N = 21^2$ with different values of $\boldsymbol{\theta}$.

$\boldsymbol{\theta}$	Running time (s)	Weight at each support point			
		(0,0)	(0,1)	(2,0)	(2,1)
$\boldsymbol{\theta} = (-2, 3, 2, 1)$	$12 \sim 13$	0.0297	0.0062	0.0353	0.9289
$\boldsymbol{\theta} = (1, 3, 2, 1)$	$13 \sim 14$	0.0008	0.0018	0.0353	0.9621
$\boldsymbol{\theta} = (-5, 3, 2, 1)$	$12 \sim 13$	0.8472	0.0623	0.0057	0.0847
$\boldsymbol{\theta} = (-2, 1, 2, 1)$	$12 \sim 13$	0.5295	0.1112	0.0445	0.3148
$\boldsymbol{\theta} = (-2, 5, 2, 1)$	$13 \sim 14$	0.0006	0.0001	0.0353	0.9640
$\boldsymbol{\theta} = (-2, 3, 4, 1)$	$12 \sim 13$	0.0043	0.0021	0.0051	0.9885
$\boldsymbol{\theta} = (-2, 3, 7, 1)$	$12 \sim 13$	0.0002	0.0017	0.0003	0.9978
$\boldsymbol{\theta} = (-2, 3, 2, -1)$	$12 \sim 13$	0.3341	0.0702	0.3972	0.1986
$\boldsymbol{\theta} = (-2, 3, 2, 3)$	$12 \sim 13$	0.0006	0.0001	0.0007	0.9986

Table 4.8: K-optimal designs for the logistic model in (4.7) where $\boldsymbol{\theta} = (-2, 3, 4, 1)$ and $N = 21^2$ with different design spaces for x_1 and x_2.

Design space for x_1	Design space for x_2	Running time (s)	Condition number	K-optimal design $\begin{bmatrix} \text{Support point} \\ \text{Weight} \end{bmatrix}$
$[0, 2]$	$[0, 1]$	$11 \sim 12$	15.2590	$\begin{bmatrix} (0,0)^\top & (0,1)^\top & (2,0)^\top & (2,1)^\top \\ 0.0043 & 0.0021 & 0.0051 & 0.9885 \end{bmatrix}$
$[0, 1.5]$	$[0, 1]$	$12 \sim 13$	20.3491	$\begin{bmatrix} (0,0)^\top & (0,1)^\top & (1.5,0)^\top & (1.5,1)^\top \\ 0.0199 & 0.0100 & 0.0092 & 0.9609 \end{bmatrix}$
$[0, 1]$	$[0, 1]$	$11 \sim 12$	33.9706	$\begin{bmatrix} (0,0)^\top & (0,1)^\top & (1,0)^\top & (1,1)^\top \\ 0.0806 & 0.0403 & 0.0215 & 0.8576 \end{bmatrix}$
$[0, 1]$	$[0, 1.5]$	$11 \sim 12$	20.3491	$\begin{bmatrix} (0,0)^\top & (0,1.5)^\top & (1,0)^\top & (1,1.5)^\top \\ 0.0121 & 0.0222 & 0.0032 & 0.9625 \end{bmatrix}$
$[0, 1]$	$[0, 2]$	$11 \sim 12$	15.2590	$\begin{bmatrix} (0,0)^\top & (0,2)^\top & (1,0)^\top & (1,2)^\top \\ 0.0016 & 0.0133 & 0.0004 & 0.9847 \end{bmatrix}$

Table 4.9: K- and D-optimal designs for the logistic model in (4.7) where $\boldsymbol{\theta} = (-2, 3, 4, 1)$ and $N = 21^2$ with the design space $x_1 \in [0, 1]$, $x_2 \in [0, 1]$ and $x_1 + x_2 \leq 1$.

Results	K-optimality	D-optimality
Optimal design $\begin{bmatrix} \text{Support point} \\ \text{Weight} \end{bmatrix}$	$\begin{bmatrix} (0,0)^\top & (0,1)^\top & (0.5,0.5)^\top & (1,0)^\top \\ 0.0996 & 0.2988 & 0.4421 & 0.1595 \end{bmatrix}$	$\begin{bmatrix} (0,0)^\top & (0,0.7)^\top & (0.55,0.45)^\top & (0.90,0)^\top & (0.95,0)^\top \\ 0.2500 & 0.2500 & 0.2500 & 0.0140 & 0.2360 \end{bmatrix}$
Running time	$6 \sim 7$	$9 \sim 10$
Condition number	105.9906	221.2636

(a) (b)

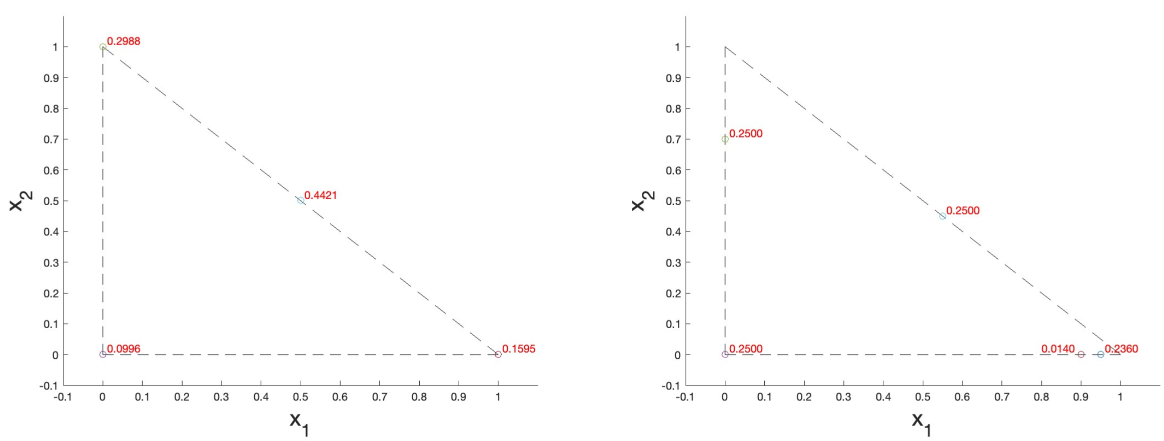


Figure 4.5: (a) plot of the locally K-optimal design, (b) plot of the locally D-optimal design, for the logistic model in (4.7) where $\boldsymbol{\theta} = (-2, 3, 4, 1)$, $N = 21^2$ with the design space $x_1 \in [0, 1]$, $x_2 \in [0, 1]$ and $x_1 + x_2 \leq 1$. The support points are circled with the corresponding weights indicated.

Chapter 5

Conclusion

This thesis aimed to study K-optimal designs for more complex models and settings than previously studied. The motivation was that K-optimality is a relatively new criterion and can help address the practical problem of ill-conditioning in experimental design to achieve a stable solution. Unlike traditional optimality criteria such as A-, D-, and E-optimality, which minimize functions of the inverse of the information matrix, K-optimality instead minimizes the condition number of the information matrix, thereby directly improving numerical stability. Since deriving theoretical results for K-optimal designs is already challenging even for simple models, this thesis focuses on numerical computation. We applied the numerical algorithm developed by Yue et al. (2023) to compute K-optimal designs on discrete design spaces for a variety of models and settings. We extended their work by exploring more complex models and design spaces, including a second-order model in a three-dimensional design space, several nonlinear models, such as the Michaelis-Menten, compartmental, and Peleg models, and a generalized linear model that focuses on logistic regression. We also applied the method to polynomial and trigonometric models in design spaces and parameter settings different from those previously studied. In addition, we compared the performance of K-optimal designs with classical D- and A-optimal designs specifically for nonlinear models and with D-optimal designs for the logistic regression model.

For the models studied in Yue et al. (2023), including polynomial, trigonometric, and second-order response models, applying different design spaces and parameter values yields similar patterns for the K-optimal designs. For polynomial models, symmetric design spaces lead to symmetric designs with better numerical performance and lower condition numbers than asymmetric spaces. As the polynomial degree increases, the condition number rises and designs become harder to compute, but symmetric design spaces help reduce instability and speed up convergence. For trigonometric models, wider design spaces are necessary to capture the periodic nature of the functions. The condition number reaches its minimum value of 2 when the design space is sufficiently wide. Models with more terms require a wider design space to achieve this, while the structure of the information matrix remains consistent. Running time increases with model complexity but is less sensitive to design space size. For second-order response models in three dimensions, the support points remain the same across different interaction structures, but weights vary depending on which interaction terms are included. The information matrix consistently combines a structured 4×4 block with a diagonal matrix. Designs assign more weight to points associated with more influential variables, and condition number changes slightly with model complexity. In general, for linear models, the use of symmetric design spaces for K-optimal designs helps create designs that are balanced, easy to understand, and numerically stable with small condition numbers.

For the nonlinear regression models studied in this thesis, including the MM-model, the compartmental model, and the Peleg model, local K-optimal designs consistently use two support points. For the MM-model and Peleg model, these are the first nonzero design point and the upper bound of the design space, while for the compartmental model, they are the two endpoints of the design space. The weight distribution between these points depends on both the model parameters $\boldsymbol{\theta}$ and the size of the design space. For the MM-model, computation is fast at $1 \sim 2$ seconds and the condition number decreases as the design space expands or as the parameter values change. For the compartmental model, the computation is also fast at $1 \sim 2$ seconds, and the condition number changes with parameter values, but is not affected by changes in N. For the Peleg model, the running time is around $3 \sim 4$ seconds and shorter than for D- and A-optimal designs, and the condition number remains stable across parameter changes for a fixed design space, and it decreases with design space expand.

For the generalized linear model (GLM) considered in this thesis, we focus on the logistic regression model. In all scenarios, local K-optimal designs consistently place support points at the vertices of the design space, with weights that vary depending on the parameter $\boldsymbol{\theta}$ and the design space. In rectangular spaces, the four corners serve as the support points, while in a triangular space, the support points include the three vertices and one additional point along the slanted edge. The condition number remains constant for a fixed design space when $\boldsymbol{\theta}$ changes but varies when the variable ranges change. For all scenarios, the K-optimal designs for different N have the same support points and condition numbers, though the larger N increases the running time. Compared to D-optimal designs, K-optimal designs require less computation time and achieve significantly smaller condition numbers.

Across all models studied in this thesis, K-optimal designs show stable structural patterns, with support points determined mainly by the design space and the type of model. The condition number and the weights of the support points are both influenced by the design space, the type of model, and the parameter values. In all cases, K-optimal designs outperform D-optimal and A-optimal designs by achieving smaller condition numbers, which indicate better numerical stability, while also requiring less computation time.

This study has several limitations. First, only specific parameter values and design spaces were selected for the models investigated, and further research could explore a wider range of parameter values and design space configurations. Second, following the general algorithm, K-optimal designs can be numerically obtained for all models, and future work could extend this to additional model types. Finally, since this thesis focuses on numerical computation, further study could investigate the theoretical properties of K-optimal designs for these models.

Bibliography

[1] Agrawal, A. and Boyd, S. (2020). Disciplined quasiconvex programming. *Optimization Letters*, 14, 1643-1657.

[2] Atkinson, A.C., Donev, A.N. and Tobias, R.D. (2007). *Optimum Experimental Designs, with SAS*. Oxford University Press, Oxford.

[3] Berni, R., Cantone, L., Magrini, A. and Nikiforova, N.D. (2022). Hierarchical optimal designs and modeling for engineering: A case-study in the rail sector. *Applied Stochastic Models in Business and Industry*, 38, 1061–1078.

[4] Bowden, G.D., Pichler, B.J. and Maurer, A. (2019). A design of experiments (DoE) approach accelerates the optimization of copper-mediated ^{18}F-fluorination reactions of arylstannanes. *Scientific Reports*, 9, 1-10.

[5] Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, New York.

[6] Brauer, F. (2008). Compartmental models in epidemiology, in *Mathematical Epidemiology*, eds, Brauer, F., van den Driessche, P. and Wu, J., 19-79, Springer Berlin, Heidelberg.

[7] Bubel, M., Seufert, P., Karpov, G., Schwientek, J., Bortz, M. and Oseledets, I. (2025). Optimal experimental design: from design point to design region. *Statistical Papers*, 66, 1-28.

[8] Chao, J., Ward, E.S. and Ober, R.J. (2016). Fisher information theory for parameter estimation in single molecule microscopy: tutorial. *Journal of the Optical Society of America A*, 33, 36-57.

[9] Charnes, A. and Cooper, W.W. (1962). Programming with linear fractional functionals. *Naval Research Logistics*, 9, 181-186.

[10] Chaubet, F., Rodriguez-Ruiz, V., Boissière, M. and Velasquez, D. (2019). Pharmacology: drug delivery, in *Encyclopedia of Biomedical Engineering*, ed, Narayan, R., 440-453, Elsevier, Amsterdam, Netherlands.

[11] Chernoff, H. (1953). Locally optimal designs for estimating parameters. *The Annals of Mathematical Statistics*, 24, 586-602.

[12] Dette, H. and Sahm, M. (1998). E-optimal designs for the double exponential model, in *MODA 5 — Advances in Model-Oriented Data Analysis and Experimental Design*, eds, Atkinson, A.C., Pronzato, L. and Wynn, H.P., 11-20, Physica, Heidelberg.

[13] Dette, H. and Wong, W.K. (1999). E-optimal designs for the Michaelis–Menten model. *Statistics and Probability Letters*, 44, 405-408.

[14] Dette, H., Melas, V.B. and Wong, W.K. (2006). Locally D-optimal designs for exponential regression models. *Statistica Sinica*, 16, 789-803.

[15] Dette, H., Pepelyshev, A. and Wong, W.K. (2012). Optimal designs for composed models in pharmacokinetic-pharmacodynamic experiments. *Journal of Pharmacokinetics and Pharmacodynamics*, 39, 295-311.

[16] Duarte, B.P.M., Atkinson, A.C. and Oliveira, N.M.C. (2023). Optimum design for illconditioned models: K–optimality and stable parameterizations. *Chemometrics and Intelligent Laboratory Systems*, 239, 1-15.

[17] Enderle, J. and Bronzino, J. (2012). *Introduction to Biomedical Engineering*, Third Edition. Academic Press, Cambridge.

[18] Freund, R.J., Wilson, W.J., Mohr, D.L. (2010). Chapter 13 - Special types of regression, in *Statistical Methods*, Third Edition, eds, Freund, R.J., Wilson, W.J., Mohr, D.L., 663-688, Academic Press, Cambridge.

[19] Grant, M. and Boyd, S. (2020). *The CVX Users' Guide*, Release 2.2. CVX Research, Inc..

[20] Haines, L.M., Kabera, G., Ndlovu, P. and O'Brien, T.E. (2007). D-optimal designs for logistic regression in two variables, in *mODa 8 - Advances in Model-Oriented Design and Analysis*, eds, López-Fidalgo, J., Rodríguez-Díaz, J.M., Torsney, B., 91-98, Physica, Heidelberg.

[21] Higham, D.J. (1995). Condition numbers and their condition numbers. *Linear Algebra and its Applications*, 214, 193-213.

[22] Hoshmand, R. (2006). *Design of Experiments for Agriculture and the Natural Sciences (2nd ed.)*. Chapman and Hall/CRC, New York.

[23] Jacroux, M. and Notz, M. (1983). On the optimality of spring balance weighing designs. *The Annals of Statistics*, 11, 970-978.

[24] Jideani, V.A. and Mpotokwana, S.M. (2009). Modeling of water absorption of Botswana bambara varieties using Peleg's equation. *Journal of Food Engineering*, 92, 182-188.

[25] Jones, B., Allen-Moyer, K. and Goos, P. (2020). A-optimal versus D-optimal design of screening experiments. *Journal of Quality Technology*, 53, 369–382.

[26] Kamruzzaman, M. (2011). *Optimal designs for Michaelis–Menten models.* Master's thesis, Institute of Statistical Research and Training (ISRT), University of Dhaka.

[27] Kaplan, C.R. and Gentry, J.W. (1987). Use of condition numbers for shortcut experimental design. *An Official Publication of the American Institute of Chemical Engineers Journal*, 33, 681-685.

[28] Kiefer, J. (1985). *Jack Carl Kiefer Collected Papers III: Design of experiments*, Springer, New York.

[29] Kiefer, J. (1959). Optimum experimental designs. *Journal of the Royal Statistical Society: Series B (Methodological)*, 21, 272–304.

[30] Kiefer, J. and Wolfowitz, J. (1959). Optimum designs in regression problems. *The Annals of Mathematical Statistics*, 30, 271-294.

[31] Lamidi, S., Kayode, O., Yekinni, A. and Adesina, K. (2024). Design of experiments (DOE): applications and benefits in quality control and assurance, in *Quality Control and Quality Assurance - Techniques and Applications*, eds, Qamar, S.Z. and Al-Hinai, N., IntechOpen, London.

[32] Layton, W.J. and Sussman, M.M. (2020). *Numerical Linear Algebra*, World Scientific, Singapore.

[33] Li, M. (2022). Optimal design, in *Encyclopedia of Ocean Engineering*, eds, Cui, W., Fu, S., Hu, Z., 1269-1278, Springer, Singapore.

[34] Li, P., Li, Y., Wang, L., Zhang, H., Qi, X. and Qian, H. (2020). Study on water absorption kinetics of black beans during soaking. *Journal of Food Engineering*, 283.

[35] López-Fidalgo, J., Martín-Martín, R. and Stehlík, M. (2008). Marginally restricted D-optimal designs for correlated observations. *Journal of Applied Statistics*, 35, 617-632.

[36] Lu, Z. and Pong, T.K. (2011). Minimizing condition number via convex programming. *SIAM Journal on Matrix Analysis and Applications*, 32, 1193-1211.

[37] Mathew, T. and Sinha, B. (2001). Optimal designs for binary data under logistic regression. *Journal of Statistical Planning and Inference*, 93, 295-307.

[38] Michaelis, L. and Menten, M.L. (1913). I93 Die Kinetik der Invertinwirkung. *Biochemische Zeitschrift*, 49, 333-369.

[39] Montanuci, F.D., Jorge, L.M.de M. and Jorge, R.M.M. (2013). Kinetic, thermodynamic properties, and optimization of barley hydration. *Food Science and Technology*, 33, 690-698.

[40] Morgan-Wall, T. and Khoury, G. (2021). Optimal design generation and power evaluation in R: the skpr package. *Journal of Statistical Software*, 99, 1–36.

[41] Müller, C.H. and Schorning, K. (2023). A-optimal designs for state estimation in networks. *Statistical Papers*, 64, 1159–1186.

[42] Nadarajah, S. (2004). Information matrix for logistic distributions. *Mathematical and Computer Modelling*, 40, 953-958.

[43] Nicholson, W.K. (2019). *Linear Algebra with Applications, 2019A*. Lyryx Learning Inc, Calgary, Alberta, Canada

[44] Nyarko, E. and Doku-Amponsah, K. (2022). Approximate and exact optimal designs for paired comparison experiments. *Calcutta Statistical Association Bulletin*, 74, 42-58.

[45] Padmanabhan, S.K., Hsuan, F. and Dragalin, V. (2010). Adaptive penalized D-optimal designs for dose finding based on continuous efficacy and toxicity. *Statistics in Biopharmaceutical Research*, 2, 182–198.

[46] Paquet-Durand, O., Zettel, V., Kohlus, R. and Hitzmann, B. (2015). Optimal design of experiments and measurements of the water sorption process of wheat grains using a modified Peleg model. *Journal of Food Engineering*, 165, 166-171.

[47] Pavithra, C.R. and Deepak, T.G. (2022). Parameter estimation and computation of the Fisher information matrix for functions of phase type random variables. *Computational Statistics and Data Analysis*, 167.

[48] Peleg, M. (1988). An empirical model for the description of moisture sorption curves. *Journal of Food Science*, 53, 1216-1217.

[49] Pornphol, P., Hammachukiattikul, P., Vadivel, R., Adebayo, S.A. and Sathasivam, S. (2024). Compartmental equations hybrid model for modelling water pollution transmission. *Journal of Advanced Research in Fluid Mechanics and Thermal Sciences*, 115, 30-50.

[50] Rabe-Hesketh, S. and Skrondal, A. (2010). Generalized linear mixed models, in *International Encyclopedia of Education (Third Edition)*, eds, Peterson, P., Baker, E., McGaw, B., 171-177, Elsevier, New York.

[51] Rahman, S.M.A., Sharma, P. and Said, Z. (2022). Application of response surface methodology based D-optimal design for modeling and optimisation of osmotic dehydration of zucchini. *Digital Chemical Engineering*, 4, 1-9.

[52] Rempel, M.F. and Zhou, J. (2014). On exact K-optimal designs minimizing the condition number. *Communications in Statistics - Theory and Methods*, 43, 1114-1131.

[53] Rodriguez, L.C., González-Nicolás, A., Ingalls, B., Streck, T., Nowak, W., Xiao, S. and Pagel, H. (2022). Optimal design of experiments to improve the characterisation of atrazine degradation pathways in soil. *European Journal of Soil Science*, 73, 1-18.

[54] Saghir, S.A., Ansari, R.A., Munir, S.T. and Munir, S.T. (2024). Fate of chemicals following exposure V: Pharmacokinetics and toxicokinetics, in, *Encyclopedia of Toxicology (Fourth Edition)*, ed, Wexler, P., 675-684, Academic Press, New York.

[55] Shafaei, S.M., Masoumi, A.A. and Roshan, H. (2016). Analysis of water absorption of bean and chickpea during soaking using Peleg model. *Journal of the Saudi Society of Agricultural Sciences*, 15, 135-144.

[56] Sopade, P.A., Ajisegiri, E.S. and Badau, M.H. (1992). The use of Peleg's equation to model water absorption in some cereal grains during soaking. *Journal of Food Engineering*, 15, 269-283.

[57] Strang, G. (2016). *Introduction to Linear Algebra. Fifth Edition.* Wellesley-Cambridge Press, Wellesley.

[58] Telford, J.K. (2007). A brief introduction to design of experiments. *Johns Hopkins APL Technical Digest*, 27, 224-232.

[59] Todd, J. (1949). The condition of certain matrices. I. *The Quarterly Journal of Mechanics and Applied Mathematics*, 2, 469–472.

[60] Tomczak, J.M. and Węglarz-Tomczak, E. (2019). Estimating kinetic constants in the Michaelis–Menten model from one enzymatic assay using approximate bayesian computation. *FEBS Letters*, 593, 2742-2750.

[61] Turing, A.M. (1948). Rounding-off errors in matrix processes. *The Quarterly Journal of Mechanics and Applied Mathematics*, 1, 287–308.

[62] van den Berg, S.M. (2018). *Analyzing Data using Linear Models (2 ed.)*. University of Twente, Enschede.

[63] Walsh, S.J. and Borkowski, J.J. (2022). Improved G-optimal designs for small exact response surface scenarios: fast and efficient generation via particle swarm optimization. *Mathematics*, 10, 1-17.

[64] Wong, W.K. and Zhou, J. (2019). CVX-based algorithms for constructing various optimal regression designs. *The Canadian Journal of Statistics*, 47, 374–391.

[65] Ye, J.J. and Zhou, J. (2013). Minimizing the condition number to construct design points for polynomial regression models. *SIAM Journal on Optimization*, 23, 666-686.

[66] Ye, J.J., Zhou, J. and Zhou, W. (2016). Computing A-optimal and E-optimal designs for regression models via semidefinite programming. *Communications in Statistics - Simulation and Computation*, 46, 2011–2024.

[67] Yeh, C. (2018). *Optimal regression design under second-order least squares estimator: Theory, algorithm and applications*. M.Sc. thesis, University of Victoria.

[68] Yue, Z., Zhang, X., Driessche, P.v.d. and Zhou J. (2023). Constructing K-optimal designs for regression models. *Statistical Papers*, 64, 205-226.

[69] Zhao, Y. (2012). *R and Data Mining: Examples and Case Studies*. Academic Press, New York.

Appendix A

MATLAB Code

In this appendix, we include samples of MATLAB code. We give an example for each design dimension and regression type: a second-order response model in three dimensions for linear regression, a Michaelis–Menten model in one dimension for nonlinear regression, and a generalized linear model in two dimensions.

A. 1 MATLAB code for second-order response models

```
clear;
clc;
runningtime=cputime;  %record computation time
tol = 10^(-5); %use to determine positive weights in optimal
    designs

%create the discrete design space
d=3; %number of variables/dimension of the design space
equal=11; %grid
N=equal^d;  %number of total design points
```

```
a=-1;    %[a, b] is the design space
b=1;
u=linspace(a,b,equal); %equally spaced N points in [a ,b]
[X1, X2, X3] = ndgrid(u, u, u); %generates 3D grids
para=1+2*d+d*(d-1)/2; %number of parameters

%build information matrices
Ai=zeros(para,para,N); %there are N of 10x10 matrix
u1=X1(:)';
u2=X2(:)';
u3=X3(:)';
for i=1:N
    f=[1 u1(i)^2 u2(i)^2 u3(i)^2 u1(i) u2(i) u3(i) u1(i)*u2(i)
       u1(i)*u3(i) u2(i)*u3(i)];
    Ai(:,:,i)=(f'*f);
end

%Compute the K-optimal design for second-order response model
   with all interaction terms
cvx_begin
    cvx_precision high
    variable v(1,N);
    expression A(para,para);
    for j=1:N
        A = A+Ai(:,:,j)*v(j);
    end
```

```
    minimize lambda_max(A)
    0 <= min(v);
    lambda_min(A) >=1;
cvx_end

%transform solution v* to w*
w=v/sum(v);

%output K-optimal design
design=[u1(find(w(1,:)>tol))' u2(find(w(1,:)>tol))' u3(find(w
   (1,:)>tol))' w(1,find(w(1,:)>tol))']  %display the result

%find condition number
L1=lambda_max(A);
Lp=lambda_min(A);
con_num=L1/Lp

%information matrix
inforMatrix=A/sum(v)

resulttime=cputime-runningtime  %computation time

%plot 3D figure
%Define each group of design points and corresponding weights
%Group 1: ( 1 ,  1 ,  1 ): 8 points
[X1, Y1, Z1] = ndgrid([-1 1], [-1 1], [-1 1]);
```

```
group1 = [X1(:), Y1(:), Z1(:)];
w1 = 0.0131;

%Group 2: (0,  1 ,  1 ): 4 points
[X2, Y2] = ndgrid([-1 1], [-1 1]);
group2 = [zeros(4,1), X2(:), Y2(:)];
w2 = 0.0239;

%Group 3: ( 1 , 0,  1 ): 4 points
group3 = [X2(:), zeros(4,1), Y2(:)];
w3 = 0.0239;

%Group 4: ( 1 ,  1 , 0): 4 points
group4 = [X2(:), Y2(:), zeros(4,1)];
w4 = 0.0239;

%Group 5: (0,0, 1 ): 2 points
group5 = [zeros(2,1), zeros(2,1), [-1;1]];
w5 = 0.0523;

%Group 6: (0, 1 ,0): 2 points
group6 = [zeros(2,1), [-1;1], zeros(2,1)];
w6 = 0.0523;

%Group 7: ( 1 ,0,0): 2 points
group7 = [[-1;1], zeros(2,1), zeros(2,1)];
```

```
w7 = 0.0523;

%Group 8: (0,0,0): 1 point
group8 = [0, 0, 0];
w8 = 0.2955;

%Combine all groups
points = [group1; group2; group3; group4; group5; group6;
   group7; group8];
weights = [repmat(w1, size(group1,1), 1);
           repmat(w2, size(group2,1), 1);
           repmat(w3, size(group3,1), 1);
           repmat(w4, size(group4,1), 1);
           repmat(w5, size(group5,1), 1);
           repmat(w6, size(group6,1), 1);
           repmat(w7, size(group7,1), 1);
           w8];

%Plot
figure;
scatter3(points(:,1), points(:,2), points(:,3), 200*weights, '
   filled','HandleVisibility','off');
hold on;
axis equal;
grid on;
xlabel('x_1'); ylabel('x_2'); zlabel('x_3');
```

```
xlim([-1.2, 1.2]);
ylim([-1.2, 1.2]);
zlim([-1.2, 1.2]);

%Group 1: ( 1 ,  1 ,  1 ), 8 points, weight = 0.0131
[X1, Y1, Z1] = ndgrid([-1 1], [-1 1], [-1 1]);
group1 = [X1(:), Y1(:), Z1(:)];
scatter3(group1(:,1), group1(:,2), group1(:,3), 50, 'b', 'filled', 'DisplayName', '(\pm1,\pm1,\pm1): 0.0131');

%Group 2 4 : 12 points, weight = 0.0239
g2 = [...
    0, 1, 1; 0, -1, 1; 0, 1, -1; 0, -1, -1;    % (0, 1 , 1 )
    1, 0, 1; -1, 0, 1; 1, 0, -1; -1, 0, -1;    % ( 1 ,0, 1 )
    1, 1, 0; -1, 1, 0; 1, -1, 0; -1, -1, 0];   % ( 1 , 1 ,0)
scatter3(g2(:,1), g2(:,2), g2(:,3), 80, 'r', 'filled', 'DisplayName', '(0,\pm1,\pm1), (\pm1,0,\pm1), (\pm1,\pm1,0): 0.0239');

%Group 5 7 : 6 points, weight = 0.0523
g3 = [...
    0, 0, 1; 0, 0, -1;       % (0,0, 1 )
    0, 1, 0; 0, -1, 0;       % (0, 1 ,0)
    1, 0, 0; -1, 0, 0];      % ( 1 ,0,0)
scatter3(g3(:,1), g3(:,2), g3(:,3), 120, [1 0.5 0], 'filled', 'DisplayName', '(0,0,\pm1), (0,\pm1, 0), (\pm1,0,0): 0.0523')
```

```
;

%Group 8: (0,0,0), weight = 0.2955
g4 = [0, 0, 0];
scatter3(g4(:,1), g4(:,2), g4(:,3), 200, 'g', 'filled', '
    DisplayName', '(0,0,0): 0.2955');

%Show legend
legend('Location', 'eastoutside');
```

A. 2 MATLAB code for the Michaelis-Menten model

```
clear;
runningtime=cputime; %record computation time
format short
tol = 10^(-5); %use to determine positive weights in optimal
    designs
N=1001; %number of distinct design points
congroup=[]'; %a matrix stores condition number of each design

theta=[1,1];
for ds=1:50
    %create the discrete design space
    a=  0; %[a, b] is the design space
    b=   ds;
```

```
u=linspace(a,b,N); %equally spaced N points in [a,b]

%model and information matrix build up
Ai=zeros(2,2,N);
for i=1:N
    f=[u(i)/(theta(2)+u(i)) -theta(1)*u(i)/((theta(2)+u(i))
       ^2)];
    Ai(:,:,i)=(f'*f);
end

%Compute the optimal design for Michaelis-Menten model
cvx_begin
   cvx_precision high
   variable v(1,N);
   expression A(2,2);
   for j=1:N
       A = A+Ai(:,:,j)*v(j);
   end

   %K-optimal
   minimize lambda_max(A)
   0 <= v;
   lambda_min(A) >=1;

   %D- and A-optimal
   %minimize (-det_rootn(A))   %D-optimal
```

```
        %minimize( trace_inv(A) )   %A-optimal
        %0 <= v <= 1;
        %sum(v)==1;
    cvx_end

    %transform solution v* to w*
    w=v/sum(v);

    %output K-optimal designs
    design=[u(find(w(1,:)>tol))' w(1,find(w(1,:)>tol))'];

    %find condition number
    Lmax=lambda_max(A);
    Lmin=lambda_min(A);
    con_num=Lmax/Lmin;
    congroup=[congroup,con_num];

    %information matrix
    inforMatrix=A/sum(v)

    %computation time
    resulttime=cputime-runningtime;
end

%create figure for the relationship between condition number
   and different design space
```

```
figure
plot(1:ds,congroup)
xlabel("b", 'FontSize',20)
ylabel("Condition Number", 'FontSize',20)
```

A. 3 MATLAB code for the generalized linear model: logistic regression model

```
clear;
runningtime=cputime;  %record computation time
tol = 10^(-4); %use to determine positive weights in optimal
    designs

theta=[-2,3,4,1];

%create the discrete design space
a1=0; %design space boundary
b1=1;
a2=0;
b2=1;
N1=51;
N2=51;
N=N1*N2; %total design points
u1=linspace(a1,b1,N1);
u2=linspace(a2,b2,N2);
```

```
%Form the 2-dimension grid points
u1=reshape(u1.*ones(N1),1,[]);
u2=repmat(u2,1,N2);

%model and information matrix build up
Ai=zeros(length(theta),length(theta),N);
i=1;
while (i<N+1)
    if u1(i)+u2(i)<=1 %setting boundary for x1+x2<=1
        le=theta(1)+theta(2)*u1(i)+theta(3)*u2(i)+theta(4)*u1(i)*u2(i);
        f=(sqrt(exp(le))/(1+exp(le)))*[1 u1(i) u2(i) u1(i)*u2(i)];
        Ai(:,:,i)=(f'*f);
        i=i+1;
    else
        u1(i)=[];
        u2(i)=[];
        N=N-1;
    end
end

%Compute the optimal design for logistic regression model
cvx_begin
    cvx_precision high
    variable v(1,N);
```

```
    expression A(length(theta),length(theta));
    for j=1:N
        A = A+Ai(:,:,j)*v(j);
    end

    %K-optimal
    %minimize lambda_max(A)
    %0 <= min(v);
    %lambda_min(A) >=1;

    %D-optimal
    minimize (-det_rootn(A))
    0 <= v <= 1;
    sum(v)==1;
cvx_end

%transform solution v* to w*
w=v/sum(v);

%output optimal designs
design=[u1(find(w(1,:)>tol))' u2(find(w(1,:)>tol))' w(1,find(w
   (1,:)>tol))']

%find condition number
L1=lambda_max(A);
Lp=lambda_min(A);
```

```
con_num=L1/Lp

%information matrix
inforMatrix=A/sum(v)

resulttime=cputime-runningtime  %computation time

%build figures
%K-optimal design
x1=[0 0];
x2=[1 0];
x3=[0 1];
figure;
line(x1,x2,'LineStyle','--', 'Color',"k");
hold on
line(x2,x3, 'LineStyle','--', 'Color',"k");
line(x3,x1, 'LineStyle','--', 'Color',"k");
scatter(0,0)
scatter(0,1)
scatter(0.5,0.5)
scatter(1,0)
hold off
axis([-0.1 1.1 -0.1 1.1])
xticks(-0.1:0.1:1)
yticks(-0.1:0.1:1)
text(0.01,0.02,'0.0996','Color',"r")
```

```
text(0.01,1.02,'0.2988','Color',"r")
text(0.51,0.52,'0.4421','Color',"r")
text(1.01,0.02,'0.1595','Color',"r")
xlabel("x_1", 'FontSize',20)
ylabel("x_2", 'FontSize',20)

%D-optimal design
x1=[0 0];
x2=[1 0];
x3=[0 1];
figure;
line(x1,x2,'LineStyle','--', 'Color',"k");
hold on
line(x2,x3, 'LineStyle','--', 'Color',"k");
line(x3,x1, 'LineStyle','--', 'Color',"k");
scatter(0,0)
scatter(0,0.7)
scatter(0.55,0.45)
scatter(0.9,0)
scatter(0.95,0)
hold off
axis([-0.1 1.1 -0.1 1.1])
xticks(-0.1:0.1:1)
yticks(-0.1:0.1:1)
text(0.01,0.02,'0.2500','Color',"r")
text(0.01,0.72,'0.2500','Color',"r")
```

```
text(0.56,0.47,'0.2500','Color',"r")
text(0.80,0.02,'0.0140','Color',"r")
text(0.96,0.02,'0.2360','Color',"r")
xlabel("x_1", 'FontSize',20)
ylabel("x_2", 'FontSize',20)
```