Novel Algorithm for Constrained Optimal Design and Information-based Subdata Selection for Logistic Model

by

Qianshun Cheng
B.A. (Xiamen University) 2011
M.S. (Univrsity of Illinois at Chicago) 2013

Thesis submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Mathematics
in the Graduate College of the
University of Illinois at Chicago, 2017

Chicago, Illinois

Defense Committee:
Min Yang, Chair and Adviser
Dibyen Majumdar
Jie Yang
Ouyang Cheng
George Karabatsos, Department of Educational Psychology

Copyright by

Qianshun Cheng

2017

ACKNOWLEDGMENT

I would like to thank my thesis adviser Min Yang for guiding and helping me with my research. He is really a great mentor. During the past five years, professor Min yang gave me endless support on my graduate study and daily life, guiding me slowly from a fresh new graduate student to a qualified researcher. He spent tremendous effort in training my skills to make me really qualify the PhD degree. His ideas and advice avoided me from going into the wrong direction on my research and are essential to every progress I made on the research projects. I am really really grateful to his fully support on all parts of my graduate life, like my graduate researches, the writing of this thesis... His suggestions and supports will continue to greatly benefit my research career and daily life.

I wish to thank professor Dibyen Majumdar for his financial support on my research and study. His support made me able to focus more on my graduate research.

I hope to thank all my professors in the STAT group of the MSCS department for teaching me knowledge and providing me various academia resources.

Last but not the least, I also like to thank all my defense committee members for serving my defense and share valuable opinions on my projects.

CONTRIBUTION OF AUTHORS

Chapter 1 & 2 borrows part of a published manuscript (Previously published as Cheng, Q., Majumdar, M. and Yang, M. (2016) On Multiple Objective Nonlinear Optimal Designs, mODa 11 - Advances in Model-Oriented Design and Analysis, pp 63-70). All the work described in the borrowed paragraphs and graphs for the first two chapters are done by myself. The work in Chapter 3 & 4 is not published yet.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

CHAPTER **PAGE**

LIST OF TABLES

LIST OF TABLES (Continued)

TABLE **PAGE**

LIST OF FIGURES

SUMMARY

My thesis includes two major parts which are described as follows.

The first part develops a new powerful algorithm for multiple-constrained optimal design problems. Experiments with multiple objectives form a staple diet of modern scientific research. Deriving optimal designs with multiple objectives is a long-standing challenging problem with only a few tools available. The few existing approaches cannot provide a fully satisfactory solution in general: either the computation is very expensive, or a satisfactory solution is not guaranteed. A novel algorithm is proposed to address this literature gap. We prove the convergence of this algorithm, and show in various examples that the new algorithm can derive the true solutions with high speed.

The second part is develops an information-based optimal subdata selection strategy, which can efficiently pick out subsample of fixed size from massive data set with the logistic regression model. Advances in computes technology have enabled an exponential growth in data collection and the size of data sets. For the extraordinary large data sets, proven statistical methods are no longer applicable due to computational limitations. A critical step in Big Data analysis is data reduction. In this thesis, we investigate the sampling approach of selecting subsets under the logistic regression model. For random sampling approaches, it is shown that the information contained in the subdata is limited by the size of the subset. A novel framework of selecting subsets is proposed. The information contained in the subdata based on the new framework increases as size of full data increases. The respective performances of the proposed approaches,

along with some of the widely-applied existing methods, are compared under various criteria based on extensive simulation studies.

CHAPTER 1

ON MULTIPLE-OBJECTIVE OPTIMAL DESIGNS

(Previously published as Cheng, Q., Majumdar, M. and Yang, M. (2016) On Multiple Objective Nonlinear Optimal Designs, mODa 11 - Advances in Model-Oriented Design and Analysis, pp 63-70)

1.1 Introduction

With the development of computational technology, nonlinear models have become more feasible and popular. An optimal/efficient design can improve the accuracy of statistical inferences with a given sample size, or reduce size of sample needed for certain level of accuracy. A major challenge in studying optimal designs of the nonlinear models is that the optimal designs, are depending on the unknown interested model parameters. Thus, a common solution to this challenge is to use the locally optimal designs, which are using the best possible guess of the parameters (1). (Hereafter, for simplicity, word "locally" is omitted.)

However, little progress has been made due to the complexity of design for nonlinear models. References (2; 3; 4; 5; 6) obtained a series of unifying results so called "complete classes" of designs. These results provided big steps towards simplifying design search for nonlinear models, even for multiple-objective design problems.

A research gap, however, still exists. It seems impossible to find the optimal design analytically, and we have to rely on a numerical solution. While we can focus on designs of a simple

form, the numerical computation may still be problematic in terms of time cost. A comprehensive review about classic and newly developed algorithms can be found in the book by Pronzato and Pázman(7). Among these efficient algorithms, Yang, Biedermann, and Tang (8) proposed a general algorithm (the optimal weights exchange algorithm - OWEA) which can identify an optimal design quickly regardless of optimality criteria and parameters of interest. While the new algorithm is for single objective design problem, it provides foundations for deriving the multiple-objective optimal designs.

In practice, it is common for a experimenter to have multiple objectives. A typical example is the multiple comparisons study in (9). There are several ways of formulating the multiple-objective optimal design problems. They include compound optimal design approach, the minimax efficient design approach, the Pareto front approach (10; 11; 12), and so forth. One popular approach formulates the optimality problem as maximizing one objective function subject to all other objective functions satisfying certain efficiencies. The constrained optimization approach provides a clearer and more intuitive interpretation than the compound optimality approach. This has made it a popular method.

However, the constrained optimization approach does not maintain the concave property. The uncertainty of the number of design points and mostly bounded design space also add difficulty to the direct use of any derivative-based approaches. Classical algorithms, like aiming at one single objective, can hardly be extended to multiple objective cases. Consequently, there is no general approach of deriving a constrained optimal design. Fortunately, there is a relationship between the two approaches. Based on the Lagrange multiplier theorem, Clyde

and Chaloner (13) generalized a result of Cookand Wong (14) and showed the equivalence of the constrained optimization approach and the compound optimality approach. A numerical solution for the constrained design problem can be derived by using an appropriate compound optimality criteria. In fact, almost all numerical solutions for constrained design problems use this strategy. However, the major challenge is how to find the corresponding weights for a given constrained optimality problem.

There are two approaches in the literature using this relation; the grid search approach and the sequential approach. The grid search approaches rapidly becomes computationally infeasible as the accuracy increase. And with three objectives, Huang and Wong (15) proposed a sequential approach for finding the weights. The basic idea is to consider the objective functions in pairs and sequentially add more constraints. While they seem to have given reasonable answers in their examples, their approach lacks theoretical justification. Consequently, this approach will generally not yield a satisfactory solution even for the three-objective optimal design problems. Other approaches for constructing contrained optimal design are also available(16; 17). The article (16) considered constructing contrained optimal designs with equality constraints, and (17) focused on finding optimal designs with system of linear contraints on weight vectors of design points. They are different from the settings in this dissertation and are thus not discussed further.

The goal of this dissertation is to propose a novel algorithm for finding the corresponding weights for a given constrained optimality problem, and then to find the corresponding optimal design. Consistency of the algorithm is proved. The performance of the new algorithm is

demonstrated by comparing it with the grid search approaches and sequential approaches. As an example, for a design with four objectives, the new algorithm can find a desired solution within 30 minutes with a laptop. In contrast, the grid search approach will take more than 10 hours and the sequential approach fails to produce a desired solution.

This chapter is organized as follows. In Section 1.2, we introduce the set up and necessary notation. Section 1.3 we briefly describe the previous works on algorithms for the constrained optimization approach. Characterization and convergence properties are presented in Section 1.4. The implementation of the algorithm, as well as its computational cost, are discussed in Section 1.5. Applications of the algorithm to many different nonlinear models, and different number of constrains, along with comparisons of the algorithm with the grid search and the sequential approach are shown in Section 1.6. Section 1.7 provides a brief discussion.

1.2 Set up and Notation

We adapt the same notation as those of (8). Suppose we have a nonlinear regression model for which at each vector point $\mathbf{x}$, the experimenter observes values of dependent response variable Y. We assume that the responses are independent and follow some exponential family distribution with mean $\eta(\mathbf{x},\boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a $(k \times 1)$ vector of unknown parameters. Typically, approximate designs are studied, i.e. designs of the form $\xi = \{(\mathbf{x}_i, \omega_i), i = 1, \ldots, m\}$ with support points $\mathbf{x}_i \in \mathcal{X}$ and weights $\omega_i > 0$, with $\sum_{i=1}^{m} \omega_i = 1$. Here, $\mathcal{X}$ denotes the original design space. The set of all approximate designs on the design region $\mathcal{X}$ is denoted by Ξ.

Denote the information matrix of ξ as $\mathbf{I}_\xi$. Let $\Phi_0(\xi), \ldots, \Phi_n(\xi)$ be the values of $n+1$ smooth objective functions for design ξ. These objective functions are some real-valued functions of

$\mathbf{I}_\xi$ which are formulated such that larger values are desirable. These objectives depend on the optimality criteria and the parameters of interest, and different objectives may have different parameters of interest. For example, $\Phi_0(\xi)$ can be the opposite number of the trace of inverse of the information matrix; and $\Phi_1(\xi)$ can be the opposite number of the determinant of the inverse of the corresponding information matrix when the parameter of interest is restricted to the first two parameters (assuming there are more than two parameters).

Ideally, we hope we can find a deisgn ξ^* which can maximize $\Phi_0(\xi), \ldots, \Phi_n(\xi)$ simultaneously among all possible designs. However, such solution does not exist in general. The constrained optimization approach specifies one objective as the primary criteria and maximizes this objective subject to the constraints on the remaining objectives (14; 13). Formally, this approach can be written as

$$\underset{\xi \in \Xi}{\text{Maximize}}\ \Phi_0(\xi) \text{ subject to } \Phi_i(\xi) \geq c_i,\ i = 1, \ldots, n, \tag{1.1}$$

where $\mathbf{c} = (c_1, \ldots, c_n)$ are user-specified constants which reflect minimally desired levels of performance relative to optimal designs for these n objective functions. To make this problem meaningful, through out this chapter, we assume there is at least one design satisfying all the constraints, which means an optimal solution exists.

Unfortunately, within the restricted optimality set up, there is no direct way of solving the constrained optimization problem. We have to solve (Equation 1.1) through the corresponding compound optimal design. Let

$$L(\xi, \mathbf{U}) = \Phi_0(\xi) + \sum_{i=1}^{n} u_i(\Phi_i(\xi) - c_i), \tag{1.2}$$

where $u_i \geq 0$, $i = 1, \ldots, n$. Let $\mathbf{U} = (u_1, \ldots, u_n)$. For a given $\mathbf{U}$, $L(\xi, \mathbf{U})$ maintains the concavity property without any restrictions. This property is critically important for applying the celebrated equivalence theorem, which enables verification whether a given design is indeed optimal. Once a $\mathbf{U}$ is given, deriving a design maximizing $L(\xi, \mathbf{U})$ can ba based on some existing algorithms, such as PSO (18); the Cocktail algorithm (19); and OWEA (8), among others. As we mentioned earlier, it is not recommended to use the compound optimal design strategy directly due to lack of a meaningful interpretation.

To establish the relationship between constrained optimal design and compound optimal design, we need the following assumptions, which are adapted from (13) . Assume that

(A1) $\Phi_i(\xi)$, $i = 0, \ldots, n$, are concave on Ξ.

(A2) $\Phi_i(\xi)$, $i = 0, \ldots, n$, are differentiable and the directional derivatives are continuous on $\mathbf{x}$.

(A3) If ξ_n converges to ξ, then $\Phi_i(\xi_n)$ converges to $\Phi_i(\xi)$, $i = 0, \ldots, n$.

(A4) There is at least one design ξ in Ξ such that the constraints in (Equation 1.1) are satisfied.

Clyde and Chaloner (13) generalized a result of Cook and Wong (14) and showed the equivalence of the constrained optimization approach and the compound optimality approach.

Theorem 1. *(13). Under assumptions A1 to A4, ξ^* is optimal for constrained optimal design (Equation 1.1) if and only if there exists a non-negative vector $\mathbf{U}^* = (u_1^*, \cdots, u_n^*) \in \Re^n$, such that*

$$\begin{aligned} \xi^* = \arg\max_{\xi \in \Xi} L(\xi, \mathbf{U}^*), \Phi_i(\xi^*) \geq c_i \quad \textit{for } i = 1, \cdots, n \\ \textit{and } \sum_{i=1}^{n} u_i^* (\Phi_i(\xi^*) - c_i) = 0. \end{aligned} \tag{1.3}$$

Theorem 1 provides necessary and sufficient condition for constrained optimal designs (Equation 1.1). It demonstrates that a numerical solution for the constrained design problem (Equation 1.1) can be derived by using an appropriate compound optimality criteria. The big challenge is how to find the desired $\mathbf{U}^*$ for a given constrained design problem (Equation 2.2). Since the exact derivative is not available, direct use of derivative-based algorithms to find this $\mathbf{U}^*$ may not be accurate and may lead to some undesired local roots. Thus they are not discussed here. There are two approaches to handle this: the grid search approach and the sequential approach. Both approaches consider the weighted optimal design, which is equivalent to compound optimal design. Let

$$\Phi_\lambda(\xi) = \sum_{i=0}^{n} \lambda_i \Phi_i(\xi), \tag{1.4}$$

where $\lambda = (\lambda_0, \ldots, \lambda_n)$, $\lambda_0 > 0$, $0 \leq \lambda_i < 1$, $i = 1, \ldots, n$ with $\sum_{i=0}^{n} \lambda_i = 1$. Clearly $\Phi_\lambda(\xi)$ is just a normalized form of $L(\xi, \mathbf{U})$. For given λ, $\Phi_\lambda(\xi)$ also enjoys the concave property as $L(\xi, \mathbf{U})$ does. So deriving a weighted optimal design can be based on the some standard algorithm, or the newly developed OWEA algorithm.

As we discuss in the introduction section, both grid search and the sequential approach (we shall give detailed description later) have their own problems. Consequently, they cannot serve as a general solution for the constrained optimal design problem (Equation 1.1). How can we develop a general and efficient algorithm for the important but largely unsolved problem? The first step is to characterize $\mathbf{U}^*$ in Theorem 1.

1.3 Characterization

For deriving theoretical results purpose, we need to have two assumptions. The first one is

$$\Phi_0 \text{ is a strict concave function on information matrices.} \tag{1.5}$$

The strict concave property means the optimal design is unique in term of information matrix, i.e., if ξ_1^* and ξ_2^* both are optimal designs for $L(\xi, \mathbf{U}^0)$ with a fixed Lagrange multiplier $\mathbf{U}^0$, then the two information matrices of ξ_1^* and ξ_2^* are identical. Assumption (Equation 1.5) is not restrictive. Many optimality objective functions satisfy this assumption. For example, D-, A-, E-, and general ϕ_p-optimality criteria (20) for full parameters satisfy this assumption.

Let ξ^* be the optimal design for a constrained optimal design problem (Equation 1.1). By Theorem 1, ξ^* is also an optimality solution of a compound optimal design problem (Equa-

tion 1.2). Let $\mathbf{U}^* = (u_1^*, \ldots, u_n^*)$ be the Lagrange multiplier of the compound optimal design problem.

In a compound optimal design problem (Equation 1.2), each $u_i > 0$ without upper bound. For an algorithm searching for $\mathbf{U}^*$, it is challenging to establish the convergence property of the algorithm when the search space is infinite. Thus our second assumption is

$$u_i^* \in [0, N_i) \text{ where } N_i \text{ is pre-specified, } i = 1, \ldots, n. \tag{1.6}$$

This assumption is equivalent to the grid size in a weighted optimal design problem (Equation 1.4). Both grid search approach and sequential approach need to choose a grid size. Let the grid size be ϵ, then it means $0 \leq u_i \leq \frac{1-\epsilon}{\epsilon} < \frac{1}{\epsilon}$ for the equivalent compound optimal design (Equation 1.2). We can always choose some reasonable large numbers N_i's such that Assumption (Equation 1.6) is satisfied.

A constraint Φ_i is called active if $u_i^* > 0$; otherwise the constraint will be regarded as inactive. For easy presentation, we denote $\xi_{\mathbf{U}}$ as a design which maximizes the Lagrange function $L(\xi, \mathbf{U})$ for a given weight vector $\mathbf{U} = (u_1, \cdots, u_n)$ and $\hat{\Phi}_i(\xi)$ as $\Phi_i(\xi) - c_i$, $i = 1, \ldots, n$. Before we characterize $\mathbf{U}$ in Theorem 1, we first give an overview of the new algorithm. The detailed description will be given in Section 4.

1.3.1 Overview of the New Algorithm

The new algorithm is designed to search for a satisfied $\mathbf{U}^*$ from the easiest case to the most complex case. It goes through all the possible cases, while following a complexity order until the right combination of active constraints are found:

All constraints are inactive $\longrightarrow$ One constraint is active

$\longrightarrow \cdots \longrightarrow$ All constraints are active.

Now consider that the constrained optimal design problem have a active constraints. Without losing generality, suppose these active constraints are $\Phi_1, \cdots, \Phi_a$. In other words, our efforts now are on finding a weight vector $\mathbf{U} = (u_1, \cdots, u_a, u_{a+1},$
$\cdots, u_n)$ where $u_1, \cdots, u_a$ are positive and $u_{a+1}, \cdots, u_n$ are zero and hopefully $\xi_{\mathbf{U}}$ will satisfy the sufficient condition.

To search for satisfied values for $u_1, \cdots, u_a$, the algorithm uses bisection process for all elements $u_1, \cdots, u_a$ through an iterative procedure. The rest elements $u_{a+1}, \cdots, u_n$ in weight vector $\mathbf{U}$ will be fixed at 0 during this process. Denote bisection result of $\mathbf{U}$ by $\mathbf{U}^* = (u_1^*, \cdots, u_a^*, 0, \cdots, 0)$. Then for any $i \in \{1, \cdots, a\}$, u_i^* will satisfy the following property:

$$\begin{aligned} &\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) > 0, \text{ then } u_i^* = 0; \\ &\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) < 0, \text{ then } u_i^* = N_i; \\ &\text{if } \hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0, \text{ then } u_i^* \in [0, N_i]. \end{aligned} \tag{1.7}$$

This property will be quoted frequently in the theorems stated later.

For example, take $a = 2$, which means only u_1 and u_2 are supposed to be nonzero. In this case, the algorithm first fixes u_2 as $u_2^0 = \frac{0+N_2}{2}$. Then the value for u_1 will be updated to u_1^0 using bisection and u_1^0 will satisfy Property (Equation 1.7) with $\mathbf{U}^0 = (u_1^0, u_2^0, 0, \cdots, 0)$. Now check $\hat{\Phi}_2(\xi_{U^0})$. If $\hat{\Phi}_2(\xi_{\mathbf{U}^0}) \neq 0$, adjust the value for u_2 through one time bisection to get u_2^1 such that $\hat{\Phi}_2(\xi_{\mathbf{U}^1})$ is closer to 0. For the new fixed $u_2 = u_2^1$, again update u_1 to u_1^1 using bisection to make u_1^1 satisfy Property (Equation 1.7) with $U^1 = (u_1^1, u_2^1, 0, \cdots, 0)$. Check $\hat{\Phi}_2(\xi_{U^1})$ and update u_2 to u_2^2 if $\hat{\Phi}_2(\xi_{U^1}) \neq 0$. Continue this process until a satisfied $\mathbf{U}^* = (u_1^*, u_2^*, 0, \cdots, 0)$ is found which guarantees that u_1^* and u_2^* both satisfy Property (Equation 1.7).

For a general a active constraints case, similar to $a = 2$ case, we first fix u_a as $u_a^0 = \frac{0+N_a}{2}$. Similar to the recursive procedure mentioned for 2 active constraints case, derive the corresponding values $u_1^0, \cdots, u_{a-1}^0$ for the element u_1 to u_{a-1} using bisections approach such that they satisfy Property (Equation 1.7), with $\mathbf{U}^0 = \{u_1^0, \cdots, u_a^0, 0, \cdots, 0\}$. Check whether $\hat{\Phi}_a(\xi_{\mathbf{U}^0}) = 0$ and update u_a to u_a^1. Continue this process until a desired $\mathbf{U}^* = (u_1^*, \cdots, u_a^*, 0, \cdots, 0)$ is found with all $u_1^*, \cdots, u_a^*$ satisfied Property (Equation 1.7).

In order to guarantee the bisection technique is valid and the desired Property (Equation 1.7) can be achieved for $u_1, \cdots, u_a$ through the bisection process, we need to characterize the property of the multiplier $\mathbf{U}$. The characterizations in this section allow us to propose a new algorithm which guarantees the convergence and speed.

1.3.2 Properties

Theorem 2. *For any* $a \in \{1, \cdots, n\}$, $S \subsetneq \{1, \cdots, n\} \backslash \{a\}$ *and* $S' = \{1, \cdots, n\} \backslash (S \bigcup \{a\})$, *define* $\mathbf{U}_S = \{u_i | i \in S\}$ *and* $\mathbf{U}_{S'} = \{u_i | i \in S'\}$. *Then* $\hat{\Phi}_a(\xi_{\mathbf{U}})$ *is a non-decreasing function of* u_a *if* $\mathbf{U}_{S'}$ *is pre-fixed and* $\mathbf{U}_S$ *satisfies one of the following two conditions:*

$$\begin{aligned} &\hat{\Phi}_i(\xi_{\mathbf{U}}) \geq 0 \text{ and } u_i \hat{\Phi}_i(\xi_{\mathbf{U}}) = 0 \text{ for } i \in S_1, \text{ or} \\ &u_i = N_i \text{ and } \Phi_i(\xi_{\mathbf{U}}) < 0 \text{ for } i \in S_2, \end{aligned} \tag{1.8}$$

where $S_1 \cup S_2 = S$ *and* $S_1 \cap S_2 = \emptyset$ *and* $\mathbf{U}$ *is the combination of* $\mathbf{U}_S$, u_a, *and* $\mathbf{U}_{S'}$ *by their corresponding indexes.*

The main purpose of Theorem 2 is to guarantee that the recursive bisection technique can be properly implemented. Condition (Equation 1.8) implies that u_i, for $i \in S$, satisfy Property given by (Equation 1.7). Suppose there are a active constraints and they are $\Phi_1, \cdots, \Phi_a$. When we search for the proper value of u_i ($i \leq a-1$), $u_{i+1}, \cdots, u_a$ and the zero-element $u_{a+1}, \cdots, u_n$ can be regarded as fixed, which correspond to $\mathbf{U}_{S'}$ in the Theorem. And since it is a recursive procedure, for $u_1, \cdots, u_{i-1}$, the value will be updated first according to the value assigned to u_i each time and fixed $u_{i+1}, \cdots, u_n$. Thus $(u_1, \cdots, u_{i-1})$ is $\mathbf{U}_S$ in this case. After $u_1, \cdots, u_{i-1}$ is

updated for the given u_i, $\hat{\Phi}_i(\xi_{\mathbf{U}})$ should be a monotone increasing function of u_i by Theorem 2. Due to the monotone property, three cases may occur when we search for u_i:

Case 1 $\hat{\Phi}_i(\xi_{\mathbf{U}}) = 0$ and $u_i \in [0, N_i]$;

Case 2 $\hat{\Phi}_i(\xi_{\mathbf{U}}) < 0$ and $u_i = N_i$;

Case 3 $\hat{\Phi}_i(\xi_{\mathbf{U}}) > 0$ and $u_i = 0$.

The three possible cases are equivalent to Property given by Equation 1.7. Under all of these possible cases that may occur when the bisection technique is applied to the former elements, Theorem 2 makes clear that the monotone increasing property holds for the next element to which the bisection technique is applied.

Now suppose the active constraints are Φ_i with $i \in S \subseteq \{1, \cdots, n\}$. A weight vector $\mathbf{U}_S^*$ for active constraints can be found through the bisection technique. One can always construct a complete weight vector $\mathbf{U}^* = (u_1^*, \cdots, u_n^*)$ as follows:

For any $i \in \{1, \cdots, n\}$

- If $i \in S$, take u_i^* as the corresponding value in $\mathbf{U}_S^*$;
- If $i \notin S$, $u_i^* = 0$.

For simplicity, we denote such constructed full weight vector $\mathbf{U}$ as $\{\mathbf{U}_S, 0\}$.

Theorem 3. *Define $S \subsetneq \{1, \cdots, n\}$ as the active constraints indexes set. For two non-zero value sets $\mathbf{U}_S^0$ and $\mathbf{U}_S^1$ of the corresponding weight vector $\mathbf{U}_S$, let $\mathbf{U}^0 = \{\mathbf{U}_S^0, 0\}$ and $\mathbf{U}^1 = \{\mathbf{U}_S^1, 0\}$, then*

$\xi_{\mathbf{U}^0}$ *will be equivalent to* $\xi_{\mathbf{U}^1}$*, i.e., they have the same information matrix, if the two designs both satisfy*

$$\hat{\Phi}_i(\xi) = 0, i \in S. \tag{1.9}$$

Now suppose $\Phi_1, \cdots, \Phi_a$ are active constraints. Theorem 3 shows all the possible weight vectors, that satisfy $\hat{\Phi}_i(\xi_{\mathbf{U}}) = 0$, $i \in \{1, \cdots, a\}$, are equivalent. Thus if we find $\mathbf{U}^* = (u_1^*, \cdots, u_a^*, 0, \cdots, 0)$ with $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0$ for $i \in \{1, \cdots, a\}$, $\mathbf{U}^*$ can represent all the possible satisfied weight vectors since they are all equivalent. For such $\mathbf{U}^*$, if $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) \geq 0$ for $i = 1, \cdots, n$, then $\mathbf{U}^*$ will be the desired weight vector. Otherwise the assumption is not valid and two cases need to be considered:

Case 1 There are still a active constraints but we need to pick another combination of constraints of size a and re-do the searching process.

Case 2 If all combinations of sized a constraints have been tested and a desired $\mathbf{U}^*$ cannot be found, then it implies the constrained optimal design problem has more than a active constraints and $a + 1$ active constraints cases should be considered.

However, the bisection technique may return a weight vector with some elements, say i-th element, taking value at lower bound 0 or upper bound N_i, while the corresponding $\hat{\Phi}_i \neq 0$. In this situation, the following theorem guarantees that the assumed active constraint set is not valid, and then we can move to a new active constraints set according to the two cases mentioned above.

Theorem 4. *For any* $S \subset \{1, \cdots, n\}$*, suppose that* $\mathbf{U}^0 = \{\mathbf{U}_S^0, 0\}$ *satisfies the following two conditions*

$$\begin{aligned} &\text{(i) } \hat{\Phi}_i(\xi_{\mathbf{U}^0}) \geq 0 \text{ for } i \in S_1 \text{ and } \sum_{i \in S_1} u_i \hat{\Phi}_i(\xi_{\mathbf{U}^0}) = 0. \\ &\text{(ii) } \hat{\Phi}_i(\xi_{\mathbf{U}^0}) < 0 \text{ and } u_i = N_i \text{ for } i \in S_2. \end{aligned} \tag{1.10}$$

where $S_1 \cup S_2 = S$ *and* $S_1 \cap S_2 = \emptyset$*. If there exists at least one element in* S*, say* i*, such that* $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) \neq 0$*, then there does not exist a positive value set* $\mathbf{U}_S^+ = \{u_i \in (0, N_i) | i \in S\}$*, such that* $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ *for* $i \in S$*, where* $\mathbf{U}^+ = \{\mathbf{U}_S^+, 0\}$*.*

Now we are ready to present the new algorithm.

1.4 Algorithm

For a given constrained optimal design problem (Equation 1.1), the new algorithm is to find the desired $\mathbf{U}^*$. In each step, we need to derive an optimal design for a compound optimal design problem (Equation 1.2) with $\mathbf{U}$ being given. We first introduce such algorithm.

1.4.1 Deriving Compound Optimal Design with given U

The paper (8) proposed the optimal weight exchange algorithm (OWEA), a general and efficient algorithm for deriving optimal designs. OWEA can be applied to commonly used optimality criteria regardless of the parameters of interest, and also enjoys high speed. This algorithm was originally designed for one objective optimal design problems. Fortunately, OWEA can be extended for deriving $\xi_{\mathbf{U}} = \mathrm{argmax}_{\xi} L(\xi, \mathbf{U})$ where $\mathbf{U}$ is given. A detailed description about OWEA algorithm can be found in the supplemental material.

Now, we are ready to present the main algorithm, which searchs for the satisfied $\mathbf{U}^*$.

1.4.2 The Main Algorithm

The algorithm searches from the simplest case (no constraint is active) to the most complicated case (all constraints are active). For each of these cases, the algorithm will implement a recursive bisection procedure. The algorithm can be described as following:

Step 1 Set $a = 0$, derive $\xi^* = \underset{\xi}{\operatorname{argmax}}\ \Phi_0(\xi)$ and check whether $\Phi_i(\xi^*) \geq c_i$ for $i = 1, \cdots, n$. If all constraints are satisfied, stop, and then ξ^* is the desired design. Otherwise, set $a = 1$ and go to Step 2.

Step 2 Set $i = 1$, consider $\xi^* = \underset{\xi}{\operatorname{argmax}}\ \Phi_0(\xi) + u_i\Phi_i(\xi)$. Adjust the value of u_i using the bisection technique on $[0, N_i]$ to obtain u_i^* such that $\hat{\Phi}_i(\xi^*) = 0$. During the bisection process, the upper bound, instead of the median, of the final bisection interval will be picked as the right value for u_i^*. If $\hat{\Phi}_i(\xi^*) > 0$ when $u_i = 0$, set $u_i^* = 0$. If $\hat{\Phi}_i(\xi^*) < 0$ when $u_i = N_i$, set $u_i^* = N_i$. For $\xi^* = \underset{\xi}{\operatorname{argmax}}\ \Phi_0(\xi) + u_i^*\Phi_i(\xi)$, check whether $\hat{\Phi}_j(\xi^*) \geq 0$ for $j = 1, \cdots, n$. If all constraints are satisfied, stop and ξ^* is the desired design; otherwise change i to $i+1$ and repeat this process. After $i = n$ is tested and no desired ξ^* is found, then set $a = 2$ and proceed to Step 3.

Step 3 Find all subsets of $\{1, \cdots, n\}$ of size a, and then choose one out of these subsets. Denote that subset as S.

Step 4 Let $(s_1, \ldots, s_a)$ be the indexes of the elements in $\mathbf{U}_S$. To find the right value $\mathbf{U}_S^*$ for $\mathbf{U}_S$, we follow a recursive procedure. For each time a given value of u_{s_a}, first use

bisection technique to find the corresponding $u_{s_1}, \cdots, u_{s_{a-1}}$. The full weight vector $\mathbf{U}$ can be constructed with $u_{s_1}, \cdots, u_{s_a}$ by setting all the other weight elements in $\mathbf{U}$ as 0's, which we later denote by $\mathbf{U} = \{\mathbf{U}_S, 0\}$. Then adapt the value of u_{s_a} as follows :

- If $\hat{\Phi}_{s_a}(\xi_{\mathbf{U}}) > 0$ when u_{s_a} is assigned as 0, set $u^*_{s_a} = 0$.
- If $\hat{\Phi}_{s_a}(\xi_{\mathbf{U}}) < 0$ when u_{s_k} is assigned as N_a, set $u^*_{s_a} = N_a$.
- Otherwise use the bisection technique to find $u^*_{s_a}$ such that $\hat{\Phi}_{s_a}(\xi_{\mathbf{U}}) = 0$.

Record $u^*_{s_a}$ and the corresponding values for $\{u^*_{s_1}, \cdots, u^*_{s_{a-1}}\}$ as U^*_S. For the bisection process in each dimension, the upper bound of the final bisection interval will be picked as the right value for the corresponding element in weight vector $\mathbf{U}^*_S$. Then the full weight vector $\mathbf{U}^*$ can be constructed using $\mathbf{U}^* = \{\mathbf{U}^*_S, 0\}$.

Step 5 For the U^*_S and $\xi_{\mathbf{U}^*}$ derived in Step 4, check $\hat{\Phi}_i(\xi_{\mathbf{U}^*})$, $i = 1, \ldots, n$. If all constraints are satisfied, stop and $\xi_{\mathbf{U}^*}$ is the desired design. Otherwise, pick another a-element subset in Step 3, and go through Step 4 to Step 5 again. If all a-element subsets are tested, then go to Step 6.

Step 6 Change a to $a+1$, go through Step 3 to Step 5, until $a = n$. If no suitable design $\xi_{\mathbf{U}^*}$ is found, the implication is that there is no solution for the constrained optimal design (Equation 1.1).

We demonstrate this algorithm through an optimal design problem with two constraints. Denote the target objective function by Φ_0 and two constrained objective functions by Φ_1 and

Φ_2. The algorithm will search for a desired weight vector $\mathbf{U}^* = (u_1^*, u_2^*)$ and desired design $\xi_{\mathbf{U}^*}$ according to the following process:

Step 1 Suppose there is no active constraint. Then $\mathbf{U}^*$ in this case will be $(0, 0)$ and $\xi_{\mathbf{U}^*}$ is also an optimal design for Φ_0. If $\xi_{\mathbf{U}^*}$ satisfies all the constraints, then $\xi_{\mathbf{U}^*}$ is the desired design. Otherwise go to Step 2.

Step 2 Suppose there is one active constraint. First suppose Φ_1 is active. Derive u_1^* through bisection technique such that $\hat{\Phi}_1(\xi_{\mathbf{U}^*}) = 0$, where $\mathbf{U}^* = (u_1^*, 0)$. If $\xi_{\mathbf{U}^*}$ satisfies all the constraints, $\xi_{\mathbf{U}^*}$ is the desired design. Otherwise suppose Φ_2 is active and repeat this process. If both fail to find the desired $\xi_{\mathbf{U}^*}$, that means there are more than one active constraint. Go to Step 3.

Step 3 Now suppose all constraints are active. Derive $\mathbf{U}^* = (u_1^*, u_2^*)$ through bisection technique such that $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0$ for $i = 1, 2$. If such $\mathbf{U}^*$ can be derived, then $\xi_{\mathbf{U}^*}$ is the desired design. If it fails to produce a satisfied $\mathbf{U}^*$, then there are two possible reasons as follows:

Case 1 The predefined upper bound vector N_1 and N_2 are improper. The true u_i^* fall out of the interval $[0, N_i)$ for at least one of i's, $i = 1, 2$,

Case 2 There is no solution for the constrained optimal design problem.

1.4.3 Convergence and Computational Cost

Whether or not an algorithm is successful depends mainly on two properties, namely, convergence and computational cost. We first establish the convergence of the proposed algorithm.

Theorem 5. *For the constrained optimal design problem (Equation 1.1), under Assumptions (Equation 1.5) and (Equation 1.6), the proposed algorithm converges to* ξ^*.

Next we shall compare the computational cost of the new algorithm with those of the grid search and the sequential approach. Both the grid search and the sequential approach are based on weighted optimal design problem (Equation 1.4), which is equivalent to a compound optimal design problem with $u_i = \frac{\lambda_i}{\lambda_0}$, $i = 1, \ldots, n$. All three approaches are based on identifying a satisfied multiplier of a compounded optimal design problem and the computational cost of each approach is proportional to the number of multiplier the approach tests.

The grid search approach considers all possible combinations of $\lambda_1, \cdots, \lambda_n$ on $[0, 1]^n$ with given mesh grid size. The combination must satisfy that $\sum_{i=1}^{n} \lambda_i < 1$ and λ_0 is set as $1 - \sum_{i=1}^{n} \lambda_i$. Suppose the grid size is ϵ in a grid search. Let T_G be the number of all possible combinations. Direct computation shows that

$$T_G = \sum_{k=0}^{n} \binom{n}{k} \binom{\lfloor \frac{1}{\epsilon} \rfloor - 1}{k} = \binom{n + \lfloor \frac{1}{\epsilon} \rfloor - 1}{n}, \tag{1.11}$$

where $\lfloor . \rfloor$ refers to floor function.

For the new algorithm, since $u_i = \frac{\lambda_i}{\lambda_0}$, the upper bound of the corresponding u_i is $1/\epsilon$. To guarantee the new algorithm has at least the same accuracy (ϵ) on interval $[0, 1/\epsilon]$ as that of grid search, one needs $\lceil -2 \log_2 \epsilon + 2 \rceil$ times bisection technique. Here $\lceil . \rceil$ refers the ceiling

TABLE I

COMPARISONS OF COMPUTATIONAL COST

	Three Objectives		Four Objectives	
Mesh Grid Size	0.01	0.001	0.01	0.001
Grid Search	5050	500500	171700	167167000
New Algorithm	289	529	4913	12167

Note: Numbers in the table are counts of weighted optimal designs calculated to solve the multiple-objective design problem for each technique.

function. Let T_L be the number of times compound optimal designs calculated during the searching process, then

$$T_L = \sum_{k=0}^{n} \binom{n}{k} \lceil -2\log_2\epsilon + 2 \rceil^k = \lceil -2\log_2\epsilon + 3 \rceil^n. \tag{1.12}$$

As for the sequential approach, the computational cost is significantly less than those of the grid search and the new algorithm. However, as we demonstrate in the next section, the sequential approach in general cannot find a desired solution.

Table I shows the comparison of computational cost between new algorithm and grid search under different grid sizes and different numbers of constraints.

1.5 Numerical Examples

In this section, we will compare the performance (accuracy and the computing time) of the new algorithm, against the grid search and the sequential approach in terms of accuracy and computing time.

All three approaches utilize the OWEA algorithm to derive optimal designs for given weighted optimal design problems. For all examples, the design space has been discretized uniformly into 1000 design points. The cut-off value for checking optimality in $L(\xi, \mathbf{U})$ for given $\mathbf{U}$ was chosen to be $\Delta = 10^{-6}$. All other set ups of OWEA are the same as those of (8).

For new algorithm and grid search, we require the algorithms to produce the best possible design while guarantee that the constraints are satisfied exactly. For the sequential approach, since it doesn't guarantee to produce a proper design and may fail during the searching process, a tolerance value $\epsilon = 0.01$ is adopted. This means that during the sequential approach process, if a design ξ_0 have $\Phi_i(\xi_0) \geq c_i - \epsilon$ for some i, then the design ξ_0 will still be regarded as a proper design which satisfies the constraint for objective function Φ_i. The grid size is 0.01 for all the examples in this section. The pre-specified upperbound N in the new algorithm is 100. All the algorithms are implemented in SAS software using a Lenovo laptop with Intel Core 2 duo CPU 2.27 HZ.

1.5.1 Three-objective Optimal Designs

Now, we compare the performance of the grid search approach, the sequential approach, and the new algorithm in term of deriving optimal designs with three objectives.

Example I Consider the nonlinear model given by

$$y = \beta_1 e^{-\theta_1 x} + \beta_2 e^{-\theta_2 x} + \epsilon. \tag{1.13}$$

This model is commonly used to compare the progression of a drug between different compartments. Here y denotes the concentration level of the drug in compartments, x denotes the sampling time, and ϵ is assumed to follow normal distribution with mean zero and variance σ^2. In a PK/PD study, Notari (21) used (Equation 1.13) to model the concentration of a drug taken at different time. The estimates of the parameters are $\theta_0 = (\theta_1, \theta_2, \beta_1, \beta_2) = (1.34, 0.13, 5.25, 1.75)$. Under these parameter estimations, Huang and Wong (15) studied three-objective optimal design with design space $x \in [0, 15]$.

Let $B = \text{diag}\{\frac{1}{\theta_1^2}, \frac{1}{\theta_2^2}, \frac{1}{\beta_1^2}, \frac{1}{\beta_2^2}\}$; $W = \int_2^{10} f(x) f^t(x) \nu(dx)$, where $f(x)$ is the linearized function of the model function using Taylor expansion at θ_0^T; $\xi_0^* = \text{argmin}_\xi \text{tr}(I^{-1}(\xi)B)$; $\xi_1^* = \text{argmin}_\xi |I^{-1}(\xi)|$; and $\xi_2^* = \text{argmin}_\xi \text{tr}(I^{-1}(\xi)W)$. The three objective functions can be written as follows:

$$\begin{aligned}
\Phi_0(I(\xi)) &= -\frac{\text{tr}(I^{-1}(\xi)B)}{\text{tr}(I^{-1}(\xi_0^*)B)}, \\
\Phi_1(I(\xi)) &= -\left(\frac{|I^{-1}(\xi)|}{|I^{-1}(\xi_1^*)|}\right)^{\frac{1}{4}}, \text{ and} \\
\Phi_2(I(\xi)) &= -\frac{\text{tr}(I^{-1}(\xi)W)}{\text{tr}(I^{-1}(\xi_2^*)W)}.
\end{aligned}$$

Define $\text{Effi}_{\Phi_i(\xi)} = -\frac{1}{\Phi_i(I(\xi))}$. Clearly $\text{Effi}_{\Phi_i(\xi)}$, $i = 0, 1, 2$ are consistent with the definitions of efficiency of design ξ under the corresponding optimality criteria. For example, $\text{Effi}_{\Phi_1(\xi)}$ refers the D-efficiency. Such definition will be used in the subsequent examples.

The three-objective optimal design problem considered in (15) is given by:

$$\begin{aligned} &\underset{\xi}{\text{Maximize}} \quad \text{Effi}_{\Phi_0(\xi)} \\ &\text{subject to} \quad \begin{cases} \text{Effi}_{\Phi_1(\xi)} \geq 0.9, \\ \text{Effi}_{\Phi_2(\xi)} \geq 0.8. \end{cases} \end{aligned}$$

Notice that the constraints $\text{Effi}_{\Phi_1(\xi)} \geq 0.9$ and $\text{Effi}_{\Phi_2(\xi)} \geq 0.8$ are obviously equivalent to $\Phi_1(I(\xi)) \geq -10/9$ and $\Phi_2(I(\xi)) \geq -5/4$, respectively. In the subsequent examples, we will use a similar efficiency setup without specifying their equivalence to the corresponding objective functions.

TABLE II

EXAMPLE I: THE RELATIVE EFFICIENCIES OF ξ_0^*, ξ_1^*, ξ_2^*, AND ξ^*

	Efficiency		
Design Type	Φ_0	Φ_1	Φ_2
ξ_0^*	1	0.7315	0.7739
ξ_1^*	0.6677	1	0.5576
ξ_2^*	0.6959	0.4166	1
ξ^*	0.8692	0.9000	0.8001

The efficiency of ξ_1^*, ξ_2^*, and ξ_3^* under each of the three objective functions is shown in Table II. Clearly, the optimal design based on one single optimal criteria has bad performance under other optimal criteria. These efficiencies are consistent with the corresponding efficiencies provided in Table 4 of (15). The new algorithm is applied to the three-objective optimal design problem. With the new algorithm, the corresponding Lagrange function is given by:

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 4.2053\Phi_1 + 2.5085\Phi_2.$$

The efficiencies of the derived constrained optimal design ξ^* are also shown in Table II. It shows that ξ^* has high efficiency on Φ_0 while guarantees the other two efficiencies are above the acceptable level.

The grid search and the sequential approach are also applied to this optimal design problem. The sequential result is also consistent with that of (15).

TABLE III

EXAMPLE I: RELATIVE EFFICIENCIES OF CONSTRAINED OPTIMAL DESIGNS BASED ON DIFFERENT TECHNIQUES

	Efficiency			
Techniques	Φ_0	Φ_1	Φ_2	Computational Cost (Seconds)
Grid Search	0.8658	0.9009	0.8000	1834
Sequential Approach	0.8917	0.8900	0.8040	52
New Algorithm	0.8692	0.9000	0.8001	103

Table III shows the efficiencies and computational time comparisons of the constrained optimal designs derived using the grid search, the sequential approach and the new algorithm.

It shows that the three approaches are essentially equivalent in this sense. The sequential approach gains highest efficiency on Φ_0 by sacrificing a little bit on constrained efficiencies. The new algorithm and grid search have slightly dropped on target efficiency to guarantee that the two constraints are exactly satisfied. The sequential approach is faster. However, the computational time in the table for sequential approach is just for one possible order. In many cases, one may need to check many possible orders to produce a satisfied solution. Thus, the computational time will tremendously increase in that case. Also in the next a few examples, however, sequential approach fails to provide a desired design.

Example II E_{max} model is commonly used in dose-finding studies. This model can be written as

$$y = \beta_0 + \frac{\beta_1 x}{\beta_2 + x} + \epsilon, \tag{1.14}$$

where x represents the dose level, ϵ is assumed to follow the normal distribution with mean zero and variance σ^2, β_0 represents the response when the dose level is at 0, $\beta_1(E_{max})$ is the maximum effect of the drug and $\beta_2(ED_{50})$ can be regarded as the dose level which produces half of E_{max}. In a dose finding study, Dette et.al (22) used Model (Equation 1.14) to find optimal design for the minimum effective dose level (MED) under parameter estimates $\beta_0 = 0$, $\beta_1 = 0.4760$, and $\beta_2 = 25$, where the relevant difference Δ is set as 0.2. Now suppose a researcher is interested

in estimating $h_0(\beta) = \beta_2$, $h_1(\beta) = \beta_1$, and $h_2(\beta) = MED = \beta_2 log(\frac{\beta_1+\Delta}{\beta_1})$. Let $c_i = \frac{\partial h_i(\beta)}{\partial \beta}$ and $\xi_i^* = argmin_\xi tr(c_i^T I^{-1}(\xi)c_i)$, $i = 0, 1, 2$. The corresponding objective functions are:

$$\Phi_0(I(\xi)) = -\frac{tr(c_0^T I^{-1}(\xi)c_0)}{tr(c_0^T I^{-1}(\xi_0^*)c_0)}, \text{ and}$$
$$\Phi_i(I(\xi)) = -\frac{tr(c_i^T I^{-1}(\xi)c_i)}{tr(c_i^T I^{-1}(\xi_i^*)c_i)}, i = 1, 2.$$

Consider the three-objective optimal design problem given by:

$$\underset{\xi}{\text{Maximize}} \quad \text{Effi}_{\Phi_0(\xi)}$$
$$\text{subject to} \quad \begin{cases} \text{Effi}_{\Phi_1(\xi)} \geq 0.7, \\ \text{Effi}_{\Phi_2(\xi)} \geq 0.65. \end{cases}$$

Utilizing the new algorithm, we find that the corresponding Lagrange function is given by:

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.4944\Phi_1 + 0.2258\Phi_2.$$

The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, and the constrained optimal design ξ^* under each of different optimal criteria are shown in Table IV.

Table V shows the efficiencies and computational time comparisons of the constrained optimal designs derived using the grid search, the sequential approach and the new algorithm. The table shows that the new algorithm produces a desired design. Grid search also produces

TABLE IV

EXAMPLE II: RELATIVE EFFICIENCIES OF ξ_0^*, ξ_1^*, ξ_2^*, AND ξ^*

	Efficiency		
Design Type	Φ_0	Φ_1	Φ_2
ξ_0^*	1.0000	0.5891	0.6670
ξ_1^*	0.0001	1.0000	0.0001
ξ_2^*	0.0028	0.0006	1.0000
ξ^*	0.9609	0.7008	0.6505

a satisfied solution, although the computational time is around fifteen times of that of the new algorithm. A notable fact is that the sequential approach could not produce a proper solution. For sequential approach, all possible orders are tested and they all fail to produce a proper design.

Sequential approach results based on different orders are shown in Table VI. ξ_{ijk}^* is the derived design based on the order $\Phi_i \rightarrow \Phi_j \rightarrow \Phi_k$ using the sequential approach. Since by the sequential approach procedure, ξ_{012}^* will be equivalent to ξ_{102}^* and ξ_{021}^* is equivalent to ξ_{201}^*, only four different orders are shown on the table. From Table VI, we can see that ξ_{210}^* performs relative well. However the efficiency for Φ_2 for ξ_{210}^* is 0.6726, while the corresponding constraint value is 0.65. This indicates ξ_{210}^* does not identify the active objective function Φ_2.

TABLE V

EXAMPLE II: RELATIVE EFFICIENCIES OF CONSTRAINED OPTIMAL DESIGNS BASED ON DIFFERENT APPROACHES

	Efficiency			
Techniques	Φ_0	Φ_1	Φ_2	Time Cost (Seconds)
Grid Search	0.9604	0.7000	0.6529	502
Sequential Approach		Failed		
New Algorithm	0.9609	0.7008	0.6505	34

Example III Atkinson et.al (23) derived Bayesian designs for a compartmental model, which can be written as

$$y = \theta_3(e^{-\theta_1 x} - e^{-\theta_2 x}) + \epsilon = \eta(x, \theta) + \epsilon. \tag{1.15}$$

where ϵ is assumed to follow the normal distribution with mean zero and variance σ^2 and y represents the concentration level of the drug at time point x. Clyde and Chaloner (13) derived multiple-objective optimal designs under this model with parameter values $\theta^T = (\theta_1, \theta_2, \theta_3) = (0.05884, 4.298, 21.80)$ and design space $[0, 30]$. It is of interest to estimate θ as well as the following quantities:

- Area under the curve (AUC),

$$h_1(\theta) = \frac{\theta_3}{\theta_1} - \frac{\theta_3}{\theta_2}$$

TABLE VI

EXAMPLE II: EFFICIENCIES OF THE DERIVED DESIGNS BASED ON DIFFERENT ORDERS USING SEQUENTIAL APPROACH

	Efficiency		
Designs	Φ_0	Φ_1	Φ_2
ξ^*_{120}	0.9036	0.6992	0.6854
ξ^*_{210}	0.9437	0.6995	0.6726
ξ^*_{102}		Fails	
ξ^*_{201}		Fails	

- Maximum concentration,

$$c_m = h_2(\theta) = \eta(t_{max}, \theta),$$

where $t_{max} = 1.01$.

Let $\xi^*_0 = \mathrm{argmin}|I^{-1}(\xi)|$, c_i be the gradient vector of $h_i(\theta)$ according to parameter vector θ and $\xi^*_i = \mathrm{argmin}tr(c_i^T I^{-1}(\xi)c_i)$, $i = 1, 2$. The corresponding objective functions can be written as follows:

$$\Phi_0(I(\xi)) = -(\frac{|I^{-1}(\xi)|}{|I^{-1}(\xi^*_0)|})^{\frac{1}{3}}, \text{ and}$$
$$\Phi_i(I(\xi)) = -\frac{tr(c_i^T I^{-1}(\xi)c_i)}{tr(c_i^T I^{-1}(\xi^*_i)c_i)}, i = 1, 2.$$

Consider the following three-objective optimal design problem:

$$\begin{aligned} &\underset{\xi}{\text{Maximize}} && \text{Effi}_{\Phi_0(\xi)} \\ &\text{subject to} && \text{Effi}_{\Phi_i(\xi)} \geq 0.4, i = 1, 2. \end{aligned}$$

Utilizing the new algorithm, we find that the corresponding Lagrange function is given by:

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.0916\Phi_1 + 0.0854\Phi_2.$$

The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, and the constrained optimal design ξ^* under different optimality criteria are shown in Table VII.

TABLE VII

EXAMPLE III: RELATIVE EFFICIENCIES OF ξ_0^*, ξ_1^*, ξ_2^*, AND ξ^*

	Efficiency		
Design Type	Φ_0	Φ_1	Φ_2
ξ_0^*	1.0000	0.3431	0.3634
ξ_1^*	0.0036	1.0000	0.0000
ξ_2^*	0.0042	0.0000	1.0000
ξ^*	0.9761	0.4008	0.4046

Table VIII compares the efficiencies and computational time comparisons of the constrained optimal designs derived using the grid search, the sequential approach and the new algorithm. The table clearly shows both new algorithm and grid search produce a satisfied solution. However, the grid search approach takes nearly eighteen times the computational time compared to that of the new algorithm. On the other hand, the sequential approach again fails to produce a satisfied solution. For the sequential approach, all possible orders are tested and results are shown in Table IX. ξ^*_{ijk} is the sequential optimal design based on order $\Phi_i \rightarrow \Phi_j \rightarrow \Phi_k$. Table IX shows sequential approach with order $\Phi_1 \rightarrow \Phi_0 \rightarrow \Phi_2$ and order $\Phi_2 \rightarrow \Phi_0 \rightarrow \Phi_1$ fails to produce a design which satisfies all the constraints. For optimal designs derived with the other two orders, although constraints are satisfied, the efficiency of the target objective function Φ_0 is far below the results from the new algorithm and the grid search. All of these results indicate that the sequential approach may not be proper for finding multiple-objective optimal design problems.

1.5.2 Four-Objective and Five-Objective Optimal Designs

In this subsection, we mainly focus on the performance of the new algorithm when there are four or five objectives. The sequential approach is dropped due to its unstable performance. The grid search is not considered either due to its lengthy computational time.

TABLE VIII

EXAMPLE III: RELATIVE EFFICIENCY OF CONSTRAINED OPTIMAL DESIGN BASED ON DIFFERENT TECHNIQUES

	Efficiency			
Techniques	Φ_0	Φ_1	Φ_2	Time Cost (Seconds)
Grid Search	0.9761	0.4042	0.4009	1047
Sequential Approach		Fails		
New Algorithm	0.9761	0.4008	0.4046	59

Example IV Under the same set up as that of Example III, we consider another parameter of interest, time to maximum concentration t_m, where

$$t_m = h_3(\theta) = \frac{\log(\theta_2) - \log(\theta_1)}{\theta_2 - \theta_1}.$$

The corresponding objective function is

$$\Phi_3(I(\xi)) = -\frac{tr(c_3^T I^{-1}(\xi) c_3)}{tr(c_3^T I^{-1}(\xi_3^*) c_3)},$$

TABLE IX

EXAMPLE III: EFFICIENCIES OF THE DERIVED DESIGNS BASED ON DIFFERENT ORDERS USING SEQUENTIAL APPROACH

	Efficiency		
Designs	Φ_0	Φ_1	Φ_2
ξ^*_{120}	0.5797	0.3908	0.5981
ξ^*_{210}	0.4537	0.6135	0.3904
ξ^*_{102}		Fails	
ξ^*_{201}		Fails	

where c_3 is the gradient vector of $h_3(\theta)$ according to vector θ and $\xi^*_3 = \mathtt{argmin}_\xi \mathtt{tr}(c_3^T I^{-1}(\xi) c_3)$.

Clyde and Chaloner (13) studied the following four-objective optimal design problem

$$\underset{\xi}{\text{Maximize}} \quad \text{Effi}_{\Phi_0(\xi)}$$

$$\text{subject to} \quad \begin{cases} \text{Effi}_{\Phi_1(\xi)} \geq 0.4, \\ \text{Effi}_{\Phi_2(\xi)} \geq 0.4, \\ \text{Effi}_{\Phi_3(\xi)} \geq 0.4. \end{cases}$$

is considered.

Utilizing the new algorithm, we find that the corresponding Lagrange funcion is

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.0916\Phi_1 + 0.0854\Phi_2.$$

This indicates that only two out of the three constrains are active, which are the objective functions Φ_1 and Φ_2. The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, and the constrained optimal design ξ^* under different optimal criteria are shown in Table X. The computational time is around 56 seconds.

TABLE X

EXAMPLE IV: THE RELATIVE EFFICIENCIES OF ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^* AND ξ^*

	Efficiency			
Design Type	Φ_0	Φ_1	Φ_2	Φ_3
ξ_0^*	1.0000	0.3431	0.3634	0.6464
ξ_1^*	0.0036	1.0000	0.0000	0.0000
ξ_2^*	0.0042	0.0000	1.0000	0.0002
ξ_3^*	0.0785	0.0001	0.0007	1.0000
ξ^*	0.9761	0.4008	0.4046	0.5143

Example V Based on the same settings as Example IV, we add another objective function:

$$\Phi_4(I(\xi)) = -\frac{tr(I^{-1}(\xi))}{tr(I^{-1}(\xi_4^*))}.$$

Here $\xi_4^* = \mathbf{argmin tr}(I^{-1}(\xi))$. Then five-objective optimal design problem

$$\begin{array}{ll} \underset{\xi}{\text{Maximize}} & \text{Effi}_{\Phi_0(\xi)} \\ \text{subject to} & \begin{cases} \text{Effi}_{\Phi_i(\xi)} \geq 0.4, i = 1,2,3 \\ \\ \text{Effi}_{\Phi_4(\xi)} \geq 0.75 \end{cases} \end{array}$$

is considered.

Result from new algorithm indicates that the corresponding Lagrange function is given by:

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.3052\Phi_1 + 0.8362\Phi_4.$$

In this case, only the objective functions Φ_1 and Φ_4 are active. The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* and the constrained optimal design ξ^* under different optimal criteria are shown in Table XI. It takes 2 minutes and 27 seconds for the new algorithm to find ξ^*.

Example VI

Consider Equation 1.13 in Example I. Suppose that we want to maximize the efficiency of D-optimal, while guaranteeing that the efficiency of C-optimal for each parameter is above 0.7. All other settings are as the same as those of example I. Let $\xi_0^* = \mathbf{argmin}|I^{-1}(\xi)|$ and $\xi_i^* = \mathbf{argmin tr}(e_i^T I^{-1}(\xi) e_i)$, $i = 1,2,3,4$, where e_i is the unit vector with i-th element equal to 1.

TABLE XI

EXAMPLE V: THE RELATIVE EFFICIENCIES OF ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* AND ξ^*

	Efficiency				
Design Type	Φ_0	Φ_1	Φ_2	Φ_3	Φ_4
ξ_0^*	1.0000	0.3431	0.3634	0.6464	0.7044
ξ_1^*	0.0036	1.0000	0.0000	0.0000	0.0000
ξ_2^*	0.0042	0.0000	1.0000	0.0002	0.0005
ξ_3^*	0.0785	0.0001	0.0007	1.0000	0.0010
ξ_4^*	0.7904	0.1138	0.6460	0.5895	1.0000
ξ^*	0.9616	0.4013	0.4184	0.4945	0.7501

The corresponding objective functions can be written as follows:

$$\Phi_0(I(\xi)) = -\left(\frac{|I^{-1}(\xi)|}{|I^{-1}(\xi_0^*)|}\right)^{\frac{1}{3}}, \text{and}$$
$$\Phi_i(I(\xi)) = -\frac{tr(e_i^T I^{-1}(\xi) e_i)}{tr(e_i^T I^{-1}(\xi_i^*) e_i)}, i = 1, 2, 3, 4.$$

Consider the following five-objective optimal design problem

$$\begin{aligned} \underset{\xi}{\text{Maximize}} \quad & \text{Effi}_{\Phi_0(\xi)} \\ \text{subject to} \quad & \text{Effi}_{\Phi_i(\xi)} \geq 0.7, i = 1, 2, 3, 4. \end{aligned}$$

Results from the new algorithm show that the corresponding Lagrange function is

$$L(\xi, \mathbf{U}^*) = \Phi_0 + 0.0183\Phi_1 + 0.3540\Phi_2 + 0.0305\Phi_4.$$

In this case, only the objective function Φ_3 is inactive. The efficiencies of ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* and the constrained optimal design ξ^*, under different optimal criteria, are shown in Table XII. It takes about 37 minutes on a laptop.

TABLE XII

EXAMPLE VI: THE RELATIVE EFFICIENCIES OF ξ_0^*, ξ_1^*, ξ_2^*, ξ_3^*, ξ_4^* AND ξ^*

	Efficiency				
Design Type	Φ_0	Φ_1	Φ_2	Φ_3	Φ_4
ξ_0^*	1.0000	0.8323	0.4461	0.6326	0.5967
ξ_1^*	0.9141	1.0000	0.3294	0.6234	0.6136
ξ_2^*	0.3849	0.1964	1.0000	0.3353	0.6422
ξ_3^*	0.1471	0.0006	0.0232	1.0000	0.0051
ξ_4^*	0.6044	0.4260	0.6867	0.6230	1.0000
ξ^*	0.9259	0.7009	0.7007	0.7212	0.7027

1.6 Discussion

While the importance of multiple-objective optimal designs is well recognized in scientific studies, their applications are still undeveloped due to a lack of a general and efficient algorithm.

The combination of the OWEA algorithm for the compound optimal design problem and the new algorithm, provides an efficient and stable framework for finding the general multiple-objective optimal designs. And Examples show remarkable improvement on computational cost, compare to the grid search approach.

For optimal designs with no more than four objective functions, the new algorithm can efficiently derive the desired solution. When there are five or more objective functions, it is unlikely that all constraints are active. If only less than four constraints are active, then the new algorithm can still solve the optimal design efficiently. However, in a rare situation where there are four or more active constraints, the computation time can become lengthy. More research works are needed to study these cases.

In order to guarantee the convergence of new algorithm, the strict concavity of the objective function Φ_0 is required. However, various cases are tested and the convergence holds for virtually all situations based on experience. It may be worthwhile to study the theoretical properties for these cases. On the other hand, the new algorithm is implemented under locally optimal designs context for all examples. It is possible to extend these results to other settings, such as the cases discussed in (24). Penalty approaches provides another strategy for finding multiple-objective optimal design. In the penalty approach, each constraint is transfered to a penalty term. Thus, the constrained optimal design problem can be transfered to a compounded optimal design problem with these penalty terms as the new optimal criteria. More research is certainly needed.

Although computer code for this new algorithm is not straightforward, the main body of the code work for all multiple-objective design problems are the same. One only needs to change the information matrix for the specific model and the specific objective functions in a multiple-objective optimal design problem. The SAS IML codes for all examples in this chapter are freely available upon request. These codes can be easily modified for different multiple-objective optimal problems.

CHAPTER 2

SUPPLEMENTAL MATERIALS FOR MULTIPLE OBJECTIVE OPTIMAL DESIGNS

(Previously published as Cheng, Q., Majumdar, M. and Yang, M. (2016) On Multiple Objective Nonlinear Optimal Designs, mODa 11 - Advances in Model-Oriented Design and Analysis, pp 63-70)

2.1 OWEA Algorithm

Since all elements in $\mathbf{U}$ are nonnegative, $L(\xi, \mathbf{U}) = \Phi_0(\xi) + \sum_{i=1}^{n} u_i(\Phi_i(\xi) - c_i)$ can be regarded as a new optimal criteria. For a design $\xi = \{(\mathbf{x}_1, w_1), \cdots, (\mathbf{x}_{m-1}, w_{m-1}), (\mathbf{x}_m, w_m)\}$, let $X = (\mathbf{x}_1, \cdots, \mathbf{x}_m)^T$ and $W = (w_1, \cdots, w_{m-1})^T$. The following algorithm follows the similar procedure as that of OWEA in (8).

Step 1 Set $t = 0$, let the initial design set X^0 take $2k$ design points uniformly from the design space and the corresponding weight to be $1/2k$ for each point.

Step 2 Derive the optimal weight vector W^t for a fixed sample points set X^t.

Step 3 For $\xi^t = (X^t, W^t)$, denote directional derivative of $L(\xi, \mathbf{U})$ at $\mathbf{x}$ as $d_{\mathbf{U}}(\mathbf{x}, \xi^t)$, where $\mathbf{x}$ is any design point from the design space $\mathcal{X}$. The explicit expression can be found in (8).

Step 4 For a small prefixed value $\Delta > 0$, if $\max_{\mathbf{x}\in\mathcal{X}} d_{\mathbf{U}}(\mathbf{x}, \xi^t) \leq \Delta$, ξ^t can be regarded as the optimal design. If $d_{\mathbf{U}}(\mathbf{x}, \xi^t) > \Delta$ for some design point $\mathbf{x}$, let $X^{t+1} = X^t \bigcup \hat{\mathbf{x}}_t$ where $\hat{\mathbf{x}}_t = \underset{\mathbf{x}\in\mathcal{X}}{\operatorname{argmax}}\ d_{\mathbf{U}}(\mathbf{x}, \xi^t)$. Then go through Step 2 to Step 4 again with new X^{t+1}.

In Step 2, the optimal weight vector $\hat{W}$ can be found by Newton's method based on the first derivative and second derivative of $L(\xi, \mathbf{U})$ respect to the weight vector W. These derivatives can be derived using Equation 2.1 and the formula in the Appendix of (8).

$$\begin{aligned} \frac{\partial \Phi_\lambda(\xi)}{\partial W} &= \frac{\partial \Phi_0(\xi)}{\partial W} + \sum_{i=1}^{n} u_i \frac{\partial \Phi_i(\xi)}{\partial W}; \\ \frac{\partial^2 \Phi_\lambda(\xi)}{\partial W W^T} &= \frac{\partial^2 \Phi_0(\xi)}{\partial W W^T} + \sum_{i=1}^{n} u_i \frac{\partial^2 \Phi_i(\xi)}{\partial W W^T}. \end{aligned} \tag{2.1}$$

Based on the exact same argument as (8), this algorithm converges to an optimal design maximizing $L(\xi, \mathbf{U})$. We use the extended OWEA to derive $\xi_{\mathbf{U}}$.

2.2 Sequential Approach Procedures

The sequential procedure for finding the corresponding compound optimal design with the specified order $\{s_1, \cdots, s_{n+1}\}$ can be described as follows:

Step 1 If $\Phi_0 \in \{\Phi_{s_1}, \Phi_{s_2}\}$, say $\Phi_0 = \Phi_{s_1}$. Consider solving the constrained optimal design problem

$$\text{Maximize } \Phi_0 \text{ while } \Phi_{s_2} \geq c_{s_2}.$$

If not, consider solving the constrained optimal design problem

$$\text{Maximize } \Phi_{s_2} \text{ while } \Phi_{s_1} \geq c_{s_1}.$$

Finding the weight vector in the weighted optimal design problem corresponding to the specified constrained optimal design problem, using the grid search with a pre-

fixed grid size. Denote the weight vector by $(1-\beta_2, \beta_2)$. Construct a new objective function $\Phi_{\{s_1,s_2\}}(\xi) = \frac{(1-\beta_2)\Phi_{s_1}(\xi)+\beta_2\Phi_{s_2}(\xi)}{(1-\beta_2)\Phi_{s_1}(\xi_{s_1,s_2})+\beta_2\Phi_{s_2}(\xi_{s_1,s_2})}$, where ξ_{s_1,s_2} is optimal design for $(1-\beta_2)\Phi_{s_1}(\xi)+\beta_2\Phi_{s_2}(\xi)$. If $n \geq 2$, set $k = 3$.

Step 2 For the newly constructed objective function, consider weighted design problem $(1-\alpha)\Phi_{\{s_1,\cdots,s_{k-1}\}} + \alpha\Phi_{s_k}$. Change the value of α by grid search on $[0,1]$ with given grid size. If $\Phi_0 \in \{\Phi_{s_1}, \cdots, \Phi_{s_k}\}$, choose a proper value α such that the corresponding weight design maximizes Φ_0 while guarantees $\Phi_{s_i} \geq c_i$ for $i = 1, \cdots, k$. If not, choose a proper value α such that the corresponding weighted optimal design maximizes Φ_{s_k} while guarantees $\Phi_{s_i} \geq c_i$ for $i = 1, \cdots, k-1$. Denote this value as β_k. If all the possible value for α fails to satisfy the constraints for $\Phi_{s_1}, \cdots, \Phi_{s_k}$, that indicates the sequential approach fails with the specified order. Then quit the algorithm.

Construct new objective function

$$\Phi_{\{s_1,\cdots,s_k\}}(\xi) = \frac{(1-\beta_k)\Phi_{s_1,\cdots,s_{k-1}}(\xi) + \beta_k\Phi_{s_k}(\xi)}{(1-\beta_k)\Phi_{s_1,\cdots,s_{k-1}}(\xi_{s_1,\cdots,s_k}) + \beta_k\Phi_{s_k}(\xi_{s_1,\cdots,s_k})},$$

where $\xi_{s_1,\cdots,s_k}$ is optimal design for $(1-\beta_k)\Phi_{s_1,\cdots,s_{k-1}}(\xi) + \beta_k\Phi_{s_k}(\xi)$. Set $k = k+1$ and repeat Step 2, until $k = n+1$.

Step 3 Transfer $\Phi_{\{s_1,\cdots,s_{n+1}\}}(\xi)$ back to $\sum_{i=0}^{n}\lambda_i\Phi_i(\xi)$ using scalar change. Then $\sum_{i=0}^{n}\lambda_i\Phi_i(\xi)$ will be the weighted optimal design problem found for constrained design problem with the sequential approach based on the specified order.

2.3 Theory and Proof

Constrained optimization approach specifies one objective as the primary criteria and maximizes this objective subject to the constraints on the remaining objectives (14; 13). Formally, this approach can be written as

$$\underset{\xi\in\Xi}{\text{Maximize}}\ \Phi_0(\xi) \text{ subject to } \Phi_i(\xi) \geq c_i,\ i = 1,\ldots,n, \tag{2.2}$$

where $\mathbf{c} = (c_1,\ldots,c_n)$ are user-specified constants which reflect minimally desired levels of performance relative to optimal designs for these n objective functions. To make this problem meaningful, throughout this chapter, we assume that there is at least one design satisfying all the constraints, which means an optimal solution exists.

Let $S \subset \{1,\ldots,n\}$, for easy presentation, we denote $\mathbf{U}_S^T\hat{\Phi}_S(\xi) = \sum_{i\in S} u_i\hat{\Phi}_i(\xi)$. We also denote $\hat{\Phi}(\xi) = (\hat{\Phi}_1(\xi),\ldots,\hat{\Phi}_n(\xi))$.

Proof of Theorem 2. Let $u_a^0 > u_a^1$ be two nonnegative values. Let $\mathbf{U}_S^0$ and $\mathbf{U}_S^1$ be the corresponding value sets for $\mathbf{U}_S$ satisfying the two conditions in the theorem when $u_a = u_a^0$ and u_a^1, respectively. Let $\mathbf{U}^0$ be the combination of $\mathbf{U}_S^0$, u_a^0, and $\mathbf{U}_{S'}$ by their corresponding indexes. Similarly let $\mathbf{U}^1$ be the counterpart of $\mathbf{U}_S^1$, u_a^1, and $\mathbf{U}_{S'}$.

Notice that for $\mathbf{U}_S^0$ and $\mathbf{U}_S^1$, the classification of S_1 and S_2 could be different. This means that elements in S_1 for $\mathbf{U}_S^0$ may fall into S_2 for $\mathbf{U}_S^1$ and versus the same. We just need to check that the two disjoint subsets from S satisfy Condition (Equation 1.8) in the theorem separately.

By the properties of $\xi_{\mathbf{U}^0}$ and $\xi_{\mathbf{U}^1}$, we have

$$\begin{aligned}
&\Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^0}) \geq \Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^0)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^1}), \text{ and} \\
&\Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^1)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^1}) \geq \Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^1)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^0}).
\end{aligned} \tag{2.3}$$

Notice that

$$\begin{aligned}
(\mathbf{U}^0)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^0}) &= (\mathbf{U}^0_S)^\intercal \hat{\Phi}_S(\xi_{\mathbf{U}^0}) + u^0_a \hat{\Phi}_a(\xi_{\mathbf{U}^0}) + (\mathbf{U}_{S'})^\intercal \hat{\Phi}_{S'}(\xi_{\mathbf{U}^0}), \\
(\mathbf{U}^0)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^1}) &= (\mathbf{U}^0_S)^\intercal \hat{\Phi}_S(\xi_{\mathbf{U}^1}) + u^0_a \hat{\Phi}_a(\xi_{\mathbf{U}^1}) + (\mathbf{U}_{S'})^\intercal \hat{\Phi}_{S'}(\xi_{\mathbf{U}^1}), \\
(\mathbf{U}^1)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^0}) &= (\mathbf{U}^1_S)^\intercal \hat{\Phi}_S(\xi_{\mathbf{U}^0}) + u^1_a \hat{\Phi}_a(\xi_{\mathbf{U}^0}) + (\mathbf{U}_{S'})^\intercal \hat{\Phi}_{S'}(\xi_{\mathbf{U}^0}), \text{ and} \\
(\mathbf{U}^1)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^1}) &= (\mathbf{U}^1_S)^\intercal \hat{\Phi}_S(\xi_{\mathbf{U}^1}) + u^1_a \hat{\Phi}_a(\xi_{\mathbf{U}^1}) + (\mathbf{U}_{S'})^\intercal \hat{\Phi}_{S'}(\xi_{\mathbf{U}^1}).
\end{aligned} \tag{2.4}$$

Adding up the two inequalities in Equation 2.3 and utilizing Equation 2.4, we have

$$(u^0_a - u^1_a)(\hat{\Phi}_a(\xi_{\mathbf{U}^0}) - \hat{\Phi}_a(\xi_{\mathbf{U}^1})) + (\mathbf{U}^0_S - \mathbf{U}^1_S)^\intercal (\hat{\Phi}_S(\xi_{\mathbf{U}^0}) - \hat{\Phi}_S(\xi_{\mathbf{U}^1})) \geq 0. \tag{2.5}$$

Suppose $i \in S_1$ when $u_a = u^0_a$ and $i \in S_2$ when $u_a = u^1_a$. Clearly that $(u^0_i - u^1_i) \leq 0$ while $(\hat{\Phi}_i(\xi_{\mathbf{U}^0}) - \hat{\Phi}_i(\xi_{\mathbf{U}^1})) \geq 0$. The conclusion holds for all other cases through the similar argument. Thus we have, for any $i \in S$, $(u^0_i - u^1_i)(\hat{\Phi}_i(\xi_{\mathbf{U}^0}) - \hat{\Phi}_i(\xi_{\mathbf{U}^1})) \leq 0$. Consequently, we have

$$(\mathbf{U}^0_S - \mathbf{U}^1_S)^\intercal (\hat{\Phi}_S(\xi_{\mathbf{U}^0}) - \hat{\Phi}_S(\xi_{\mathbf{U}^1})) = \sum_{i \in S} (u^0_i - u^1_i)(\hat{\Phi}_i(\xi_{\mathbf{U}^0}) - \hat{\Phi}_i(\xi_{\mathbf{U}^1})) \leq 0, \tag{2.6}$$

which indicates

$$(u_a^0 - u_a^1)(\hat{\Phi}_a(\xi_{\mathbf{U}^0}) - \hat{\Phi}_a(\xi_{\mathbf{U}^1})) \geq 0. \tag{2.7}$$

Thus the conclusion follows. □

Proof of Theorem 3. By the definitions of $\xi_{\mathbf{U}^0}$ and $\xi_{\mathbf{U}^1}$, we have

$$\begin{aligned} &\Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^0}) \geq \Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^0)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^1}), \text{ and} \\ &\Phi_0(\xi_{\mathbf{U}^1}) + (\mathbf{U}^1)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^1}) \geq \Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^1)^\intercal \hat{\Phi}(\xi_{\mathbf{U}^0}). \end{aligned} \tag{2.8}$$

By Equation 1.9, Equation 2.8 can be rewritten as

$$\begin{aligned} &\Phi_0(\xi_{\mathbf{U}^0}) \geq \Phi_0(\xi_{\mathbf{U}^1}), \text{ and} \\ &\Phi_0(\xi_{\mathbf{U}^1}) \geq \Phi_0(\xi_{\mathbf{U}^0}), \end{aligned} \tag{2.9}$$

which implies

$$\Phi_0(\xi_{\mathbf{U}^0}) = \Phi_0(\xi_{\mathbf{U}^1}). \tag{2.10}$$

Thus

$$L(\xi_{\mathbf{U}^0}, \mathbf{U}^0) = \Phi_0(\xi_{\mathbf{U}^0}) = \Phi_0(\xi_{\mathbf{U}^1}) = L(\xi_{\mathbf{U}^1}, \mathbf{U}^0). \tag{2.11}$$

This indicates that $\xi_{\mathbf{U}^0}$ and $\xi_{\mathbf{U}^1}$ both maximize Lagrange function $L(\xi, \mathbf{U}^0)$.

Since Φ_0 is strictly concave function on information matrices, $L(\xi, U^0)$ is also strictly concave function on information matrices. Thus, $\xi_{\mathbf{U}^0}$ and $\xi_{\mathbf{U}^1}$ have the same information matrix, implying $\xi_{\mathbf{U}^0}$ is equivalent to $\xi_{\mathbf{U}^1}$.

□

Proof of Theorem 4. Define $S_{11} = \{i | \hat{\Phi}_i(\xi_{\mathbf{U}^0}) > 0, i \in S_1\}$. By the properties of $\mathbf{U}^0$, clearly we have $u_i^0 = 0$ for $i \in S_{11}$ and $u_i^0 = N_i$ for $i \in S_2$. Suppose there exists a positive value set $\mathbf{U}^+ = \{\mathbf{U}_S^+, 0\}$ with $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ for $i \in S$. Then we have

$$\begin{aligned} &\Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^0}) \geq \Phi_0(\xi_{\mathbf{U}^+}) + (\mathbf{U}^0)^T \hat{\Phi}(\xi_{\mathbf{U}^+}) \text{ and} \\ &\Phi_0(\xi_{\mathbf{U}^+}) + (\mathbf{U}^+)^T \hat{\Phi}(\xi_{\mathbf{U}^+}) \geq \Phi_0(\xi_{\mathbf{U}^0}) + (\mathbf{U}^+)^T \hat{\Phi}(\xi_{\mathbf{U}^0}). \end{aligned} \tag{2.12}$$

Then, the summation of the two inequalities in Equation 2.12 returns

$$\begin{aligned} &(\mathbf{U}^0_{S/(S_{11} \cup S_2)} - \mathbf{U}^+_{S/(S_{11} \cup S_2)})^T (\hat{\Phi}_{S/(S_{11} \cup S_2)}(\xi_{\mathbf{U}^0}) - \hat{\Phi}_{S/(S_{11} \cup S_2)}(\xi_{\mathbf{U}^+})) \\ &+ (\mathbf{U}^0_{S_{11}} - \mathbf{U}^+_{S_{11}})^T (\hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^0}) - \hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^+})) \\ &+ (\mathbf{U}^0_{S_2} - \mathbf{U}^+_{S_2})^T (\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0}) - \hat{\Phi}_{S_2}(\xi_{\mathbf{U}^+})) \geq 0. \end{aligned} \tag{2.13}$$

By Condition Equation 1.10, and our assumption, we have that (i) $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) = 0$ for $i \in S/(S_{11} \cup S_2)$; (ii) $u_i^0 = 0$ for $i \in S_{11}$; and (iii) $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ for $i \in S$. Thus, (Equation 2.13) reduces to

$$(-\mathbf{U}^+_{S_{11}})^T \hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0_{S_2} - \mathbf{U}^+_{S_2})^T (\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0})) \geq 0. \tag{2.14}$$

Notice that, for $i \in S_{11}$, $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) > 0$ and $\mathbf{U}^+_{S_{11}} > 0$. We have

$$(-\mathbf{U}^+_{S_{11}})^T \hat{\Phi}_{S_{11}}(\xi_{U^0}) < 0. \tag{2.15}$$

On the other hand, for $i \in S_2$, $u^0_i = N_i > \mathbf{U}^+_i$ and $\hat{\Phi}_i(\xi_{\mathbf{U}^0}) < 0$, we have

$$(U^0_{S_2} - U^+_{S_2})^T (\hat{\Phi}_{S_2}(\xi_{U^0})) < 0. \tag{2.16}$$

Since $S_{11} \bigcup S_2 \neq \emptyset$, we have

$$(-\mathbf{U}^+_{S_{11}})^T \hat{\Phi}_{S_{11}}(\xi_{\mathbf{U}^0}) + (\mathbf{U}^0_{S_2} - \mathbf{U}^+_{S_2})^T (\hat{\Phi}_{S_2}(\xi_{\mathbf{U}^0})) < 0. \tag{2.17}$$

This is contradiction to Equation 2.14. Thus the conclusion follows. □

Proof of Theorem 5. Since there exists an optimal solution for the constrained optimal design problem (1), there exists an active constraints set. (It could be empty set, which means no active constraints). The new algorithm will search for these active constraints set and identify the Lagrange multiplier of the corresponding compound optimal design problem. The new algorithm starts from the simplest case, i.e., there is no active constraints, to most complex case, i.e., all constraints are active.

For each assumed active constraints set S, by Theorem 2, the algorithm procedure utilizing the bisection technique to guarantee that the derived vector $\mathbf{U}^* = \{\mathbf{U}^*_S, 0\}$ satisfies the two conditions in Equation 1.10. If $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) \neq 0$ for some $i \in S$, Theorem 4 guarantees that there

is no positive value set $\mathbf{U}_S^+$ within the given intervals such that $\hat{\Phi}_i(\xi_{\mathbf{U}^+}) = 0$ for all $i \in S$ where $\mathbf{U}^+ = \{\mathbf{U}_S^+, 0\}$. This means S cannot be the true active constraints set. Otherwise, it contradicts Assumption (6). On the other hand, if $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0$ for all $i \in S$ but $\hat{\Phi}_{i'}(\xi_{\mathbf{U}^*}) < 0$ for some $i' \in \{1, \cdots, n\} \backslash S$, Theorem 3 guarantees that $\hat{\Phi}'_i(\xi_{\mathbf{U}'}) < 0$ for any vector $\mathbf{U}' = \{\mathbf{U}'_S, 0\}$ satisfying $\hat{\Phi}_i(\xi_{\mathbf{U}'}) = 0$ for all $i \in S$. This also means S cannot be the true active constraints set.

Since the new algorithm goes through all possible active constraints combinations, a desired $\mathbf{U}^*$, i.e., $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) = 0$ for all $i \in S$ and $\hat{\Phi}_i(\xi_{\mathbf{U}^*}) \geq 0$ for all $i \in \{1, \cdots, n\} \backslash S$, must be found. Otherwise, it means none of the constraint combinations are active. This contradicts the fact that there is an active constraints set.

For the desired $\mathbf{U}^*$, let $\xi^* = \mathtt{argmax}_\xi L(\xi, \mathbf{U}^*)$. By Theorem 1, ξ^* is the optimal design of the constrained optimal design problem (Equation 1.1). □

CHAPTER 3

THE IBOSS ALGORITHM FOR LARGE-SCALE LOGISTIC REGRESSION

3.1 Introduction

Technological advances have enabled an exponential growth in data collection and the size of data sets. For example, the cross-continental Square Kilometre Array, the next generation of astronomical telescopes, will generate 700 TB of data per second (25). While the extraordinary size datasets provide researchers golden opportunities of scientific discoveries, they also bring tremendous challenges for analyzing these big datasets. Many proven and classical statistical methods are no longer applicable due to computational limitations.

There are some recent advances in statistical analysis to deal with these challenges. Roughly speaking, there are three major directions, namely, (1) the divide and conquer approach, (2) the sequential updating method and (2) the subsampling-based approach.

The divide and conquer approach takes advantage of parallel computing. Data are split into chunks of reasonable size. Then, analysis is implemented separately on each chunk of data, and a specified aggregation method is implemented to merge the piece of information from chunks and produce final analysis. Strategy of analysis on each chunks and aggregation methods varies depends on the structure of the data and model assumptions. For linear regression model, the least square estimate can be directly decomposed into a weighted average of the

least square estimate of each chunk. This has became the standard aggregation method for merging solutions from blocks with linear models. For the nonlinear models, several aggregation methods are proposed. Lin and Xi(26) proposed an approach of approximating the estimating equation using Taylor expansion. Under certain conditions and assumptions, accuracy of the final estimator from this approach is closed to the direct estimator from full data. Chen and Xie(27) considered a divide and conquer type approach for generalized linear models (GLM), where both the number of observations n and the dimension of covariates vector p are large, by incorporating variable selection and using penalized regression into the processing steps of each chunk. The authors show under certain regularity conditions that their combined estimator is model selection consistent and asymptotically consistent with the penalized estimator based on the full dataset.

The sequential updating approach works on streaming data. Since the data is arriving in chunks, some of the logics and ideas from divide and conquer approach can be borrowed. In Schifano et.al(28), an approach similar to divide and conquer approach is proposed. The estimate of the parameters based on current data can be updated using previous estimate of parameter and the new coming data. The consistency of the estimate is guaranteed under certain regularity conditions. Both the divide and conquer approach and the sequential updating approach target on chunk analysis and aggregation of analysis result from each pieces. The two approaches gain efficiency mainly from the implementation of parallel computing. They take advantage of more computational resources available. In some senses, these two approaches do

not provide a true saving of computational cost. In addition, it may still be time-consuming when dealing with extremely large data sets using these techniques.

The subsampling approaches, which is to analyze subsamples of the data, downsize the data and reduce the computation burden. Combining the methods of subsampling (29) and bootstrap (30; 31), (32) proposed a novel approach called bags of little bootstrap (BLB) to achieve computational efficiency. Liang et.al(33) and Liang and Kim(34) proposed a mean log-likelihood approach using Monte Carlo averages of estimates from subsamples to approximate the quantities needed for the full data. It seems BLB and mean log-likelihood select the subsamples using simple random sample algorithm. Another line of subsampling approach is based on sampling-based algorithms to select a subsample. In this approach, a sampling probability is assigned to each dataline according to its leveraging score and subsampling procedure is performed based on the assigned sampling probability. Ma and Sun(35) reviewed the existing subsampling methods in the context of linear regression and termed them leveraging algorithms. Ma et.al(36) considered the statistical properties of leveraging algorithms. They systematically derived biases and variances of algorithmic leveraging methods in linear regression models, and then proposed a shrinkage algorithmic leveraging method to improve the performance.

A major limitation of random subsampling approaches is that the amount of information in the subdata based on subsampling approach, generally is proportional to the size of the subdata, which is significantly smaller than the full data size. Wang et.al(37) proved that, under linear models, the variances of estimates based on the random subsampling approach converge to zero at the rate proportional to the size of subdata. Is it possible that the subdata contain

information that is related to the size of full data rather than that of the subdata? Ideally, we choose the subdata with maximum information among all possible subdata sets. However, this is impossible in practice since there are $\binom{n}{r}$ possible ways of selecting a subdata with size r from a full data with size n. This combination number is quickly out of reach even for moderate numbers n and r. An alternative approach need to be employed here. Under linear models, Wang et.al(37) proposed a novel approach called Information-Based Optimal Subdata Selection (IBOSS) to select a subdata. Unlike random subsampling approaches, IBOSS is a deterministic approach. It selects a subdata based on the characterization of the D-optimal design. Under certain conditions, Wang et.al(37) showed that variances of the estimates converge to zero at the rate of the size of full data. The simulation studies demonstrate that the performance of IBOSS is significantly improving from the existing subsampling approaches.

Can the IBOSS strategy be extended to generalized linear models, especially logistic regression model? The logistic regression models, since developed by Cox(38), have played an important role in categorical data analysis. It has been widely used in various fields, such as finance, various medical fields and the social sciences. Unlike the linear models, where closed form estimation is available, there are no closed form estimation of generalized linear models, including logistic regression models. Instead, estimations need to be carried out using an iterative algorithm. The computation complexity is then even more challenging for big dataset. There is relatively little work on how to choose a subdata from a full dataset under generalized linear models. This is perhaps due to the complexity of the nonlinearity features. Wanf et.al(39) proposed the OSMAC algorithm using the leveraging method in order to handle the subsam-

pling work for logistic model. The probability weight is built up based on the A-optimality from optimal design theory (40). Based on different transformations of the the A-optimality, several subsampling strategies are proposed in their work. Among all these approaches, the mVc approach attains almost the best accuracy on the defined criteria under their simulation set up, while maintaining endurable computational cost. However, like many other leveraging approaches for linear regression case, as we shall show in the next section, the information extracted from mVc approach is limited by the subsample size.

Inspired by the IBOSS approach for the linear case, in this paper, we study subdata selection under logistic regression models. A new algorithm of selecting a subdata is proposed. Compared to the existing subsampling approaches, the new algorithm has following advantages: (1) the performance of the algorithm is significantly better; (2) the computational cost is competitive; and (3) the selecting procedure is independent of the response variable. Since "data reduction is perhaps the most critical component in retrieving information in big data" (41), this result is valuable approach in big data analysis.

This chapter is organized as follows. Section 3.2 introduces the notations, gives a summary of existing methods, and present an upper-bound of the information matrix for subsampling-based estimators. A new algorithm as well as its asymptotic properties will be introduced in Section 3.3. Section 3.4 compares the performances of the new algorithm, OSMAC algorithm and simple random sampling using various simulation settings. Section 3.5 proposes an alternative directional derivative approach under un-asymmetric cases. A brief summary of this paper and possible extensions are given in section 3.6.

3.2 Notations and Existing Methods

3.2.1 Notations

Let (Y_i, Z_i), $i = 1, \ldots, n$ denote the full data, where Y_i is a binary response and $Z_i = (x_{i1}, \cdots, x_{im})^T$ is the m dimension explanatory variables. Assume the logistic regression model:

$$\text{Prob}(Y_i = 1|X_i) = \frac{e^{X_i^T \beta}}{1 + e^{X_i^T \beta}}, \tag{3.1}$$

where $\boldsymbol{\beta} = (\beta_0, \cdots, \beta_m)^T$ and $X_i = (1, Z_i^T)^T$. β_0 is called the intercept parameter and $(\beta_1, \cdots, \beta_m)^T$ is the m dimension slope parameters. Denote the covariate matrix $\mathbf{X} = (X_1, \cdots, X_n)^T$ and response vector $\mathbf{Y} = (Y_1, \cdots, Y_n)^T$. Thus, the full data can be represented by $(\mathbf{X}, \mathbf{Y})$. Like linear models, the maximum likelihood estimator is frequently used to approximate $\boldsymbol{\beta}$. However, for logistic regressions, there is no closed-form of the maximum likelihood estimates (MLEs), then iterative algorithms such as Newton-Raphson algorithm (42) are often used to approximately find the maximum likelihood estimator $\widehat{\boldsymbol{\beta}}$. In this paper, denote the MLE of $\boldsymbol{\beta}$ based on the full data as $\widehat{\boldsymbol{\beta}}$. The computational cost of approximating $\widehat{\boldsymbol{\beta}}$ is at the order of $O(\Delta n m^2)$, where Δ is the number of iterations. Thus, for extraordinary large n, the computational cost could be beyond the available computation capacity. We may have to consider analyzing a subdata set instead of the full data. The research question here is: suppose we can only analyze a subdata with size r, how can we choose the subdata such that it contains the most information? Let $\alpha_1, \ldots, \alpha_n$ be the indicator whether the data point is selected or not, i.e., $\alpha_i = 1$ if (Y_i, X_i) is selected in the subdata and 0 otherwise. Let $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ and β_i^{α} be the resulting estimate

from the subdata selected through $\boldsymbol{\alpha}$. Our goal is, under the restriction $\sum_{i=1}^{n} \alpha_i = k$, to find $\boldsymbol{\alpha}$ such that

$$MSE(\boldsymbol{\beta}) = \sum_{i=1}^{m} (\beta_i^{\alpha} - \beta_i)^2 \tag{3.2}$$

is as small as possible.

In literature, many subsampling strategies are designed based on minimizing this MSE term. But most of them target data under linear models, and usually cannot be easily extended to logistic regression cases due to their nonlinearity of the logistic model. The few strategies suitable for logistic models are given by the simple random sampling approach (SRS) and the OSMAC approach from (39).

3.2.2 Existing Subsampling Approaches

When SRS is implemented, each dateline has an equal chance to be selected. It can be regarded as a leveraging type subsampling approach where each dataline has an equal probability of $\frac{1}{n}$ to be picked. It is a widely-used technique for many years to downsize data due to its simplicity and low cost on computational resources. However, in term of the information retrieved from a big dataset, SRS may not be the best choice.

Inspired by the A-optimality criteria from optimal design theories, Wang et.al(39) proposed the novel OSMAC algorithm. The OSMAC algorithm is a leveraging type approach. The probability weights are assigned to each dataline in a way to optimize the A-optimal criteria of the slope parameter with certain pre-specified strategies. According to the use of different

strategies to optimize the A-optimal criteria, the OSMAC approach proposes four different ways of calculating probability weights. They are the mMSE, mEMSE, mVc, and mEVc strategies, respectively. Among all of them, the mMSE strategy directly minimizes the asymptotic MSE of $\hat{\boldsymbol{\beta}}$ and has the best performance in terms of accuracy. However, the computational cost of mMSE strategy for calculating the sampling weights is not endurable. Within the remaining three strategies, mVc strategy's estimation accuracy is the closest to the mMSE strategy, while its computational cost is acceptable. Instead of directly attempting to minimize the asymptotic MSE of $\hat{\boldsymbol{\beta}}$, this strategy targets on finding weights to minimize the asymptotic MSE of $M_X\hat{\boldsymbol{\beta}}$, where $M_X = \frac{1}{n}\sum_{i=1}^{n} w_i(\hat{\boldsymbol{\beta}})X_iX_i^T$. The corresponding sampling probability for each dataline under the mVc strategy is given by

$$\pi_i^{mVc} = \frac{|y_i - p_i(\hat{\boldsymbol{\beta}})|\|X_i\|}{\sum_{j=1}^{n}|y_j - p_j(\hat{\boldsymbol{\beta}})|\|X_j\|}.$$

The other two strategies in Wang et.al(39) are inferior to the mVc strategy on MSE accuracy based on the simulation settings in their paper. Thus they are not discussed here.

3.2.3 Limitations on Estimation Efficiency for Existing Subsampling Strategies

The asymptotic property of the ML estimator $\hat{\boldsymbol{\beta}}$ has been studied for many years. For models within the exponential family, different approaches to prove the consistency of the maximum likelihood estimator (43; 44; 45) is available under various conditions. For the set up of assumptions, one can refer to (43) and (46) , while the MLE consistency of logistic regression

can be regarded as one typical case of the MLE consistency of the exponential family. For the full data MLE $\hat{\boldsymbol{\beta}}$, as $n \to \infty$,

$$(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^0) \sim N(0, (\sum_{i=1}^{n} \Psi(u_i) X_i X_i^T)^{-1}),$$

where $c_i = X_i^T \beta$ and $\Psi(c_i) = \frac{e^{c_i}}{(1+e^{c_i})^2}$. The derivation details and necessary conditions can be found in the next chapter. The variance term $(\sum_{i=1}^{n} \Psi(c_i) X_i X_i^T)^{-1}$ determines how accurate, asymptotically, the MLE $\hat{\boldsymbol{\beta}}$ is on estimating true value $\boldsymbol{\beta}$. It is more convenient to work on its inverse, the information matrix, $I = \sum_{i=1}^{n} \Psi(c_i) X_i X_i^T$. Roughly speaking, the information contains in the data to estimate parameter $\boldsymbol{\beta}$ increases as the size n increases. When the subsampling strategies are implemented, the subsample size r is usually fixed although the data size n might be very large. Does the MLE from subsample still remains the asymptotic consistency? Is the Fisher information matrix $I_{\boldsymbol{\alpha}}$ of the picked subsample bounded by subdata size r despite the fact n goes to infinity?

For SRS, the mle consistency of the sub data can be relatively easily proved. However, it can be shown that, with this approach, the expectation of $I_{\boldsymbol{\alpha}} = \sum_{i=1}^{n} \alpha_i \Psi(c_i) X_i X_i^T$ is bounded when r is bounded (we will show this in the next section). Therefore, the information we can extract from the SRS is limited. This is one major drawback of the SRS strategy, despite its advantage on speed and simplicity. With a much more advanced ranking strategy, does the leveraging type approach, mVc strategy, also have this kind of limitation when size of subdata r is fixed?

For the leveraging type subsampling strategies, the sampling probability of each dataline is based on the rank of the importance among them. This rank provides a way to directly assess the importance of each dataline. To adjust for weight difference across different datalines picked, the weighted maximum likelihood estimator is often used to estimate the parameters with the picked subdata. However, also due to complexity of the probability weight, the weighted MLEs' consistency can no longer be guaranteed. Alternative consistency has been proposed to theoretically support the leveraging type subsample algorithms. Wang et.al(39) proved consistency of leveraging type algorithms between MLEs of subdata and MLEs of full data.

Theorem 6. *(Wang et. al, 2016) Let* n *denote full data size and* r *denote subsample size. The leveraging sampling probability for each dataline is set as* π_i*. Let* F_N *denote the set of* $(\mathbf{Y}, \mathbf{X})$*. Under certain conditions specified in Wang et.al(2016), as* $n, r \to \infty$*, conditional on* F_N*,*

$$V^{-\frac{1}{2}}(\hat{\boldsymbol{\beta}}_{sub} - \hat{\boldsymbol{\beta}}) \to N(0, I)$$

where $\hat{\boldsymbol{\beta}}_{sub}$ *is the weighted maximum likelihood estimation for subdata,* $\hat{\boldsymbol{\beta}}$ *is the maximum likelihood estimation of full data and* $V = M_X^{-1} V_c M_X^{-1}$*. In this formula,* $M_X = \frac{1}{n}\sum_{i=1}^{n} w_i(\hat{\boldsymbol{\beta}}) X_i X_i^T$*,* $V_c = \frac{1}{rn^2}\sum_{i=1}^{n} \frac{(y_i - p_i(\hat{\boldsymbol{\beta}}))^2 X_i X_i^T}{\pi_i}$ *and* $w_i(\beta) = p_i(\boldsymbol{\beta})(1 - p_i(\boldsymbol{\beta}))$*. This result can be rewritten as*

$$\hat{\boldsymbol{\beta}}_{sub} - \hat{\boldsymbol{\beta}} | F_N \sim N(0, V).$$

This theorem builds up the bridge between maximum likelihood estimator of subdata and full data for all leveraging type subsampling procedures. On the other hand, it does not

show how much improvement the leveraging type approaches can have on estimation accuracy with fixed subsample size r. The following theorem investigates I_{sub} and gives some insight information on this term.

Theorem 7. *For leveraging type algorithms, if the conditions in Wang et.al(2016) is satisfied and the consistency described in 6 holds, then the information matrix* $I_{sub} = V^{-1}$ *is dominated, in term of Loewring ordering, by a matrix proportional to the size of subdata, i.e.,*

$$I_{sub} \leq r(\sum_{i=1}^{n} \pi_i X_i X_i^T).$$

Applying Theorem 7 to sampling-based methods, we can show that the information from the subdata is bounded by a term associated with subdata size r.

Theorem 8. *For simple random sampling, suppose* $X_i, i = 1, \ldots, n$ *are generated independently from the same distribution* X *which has finite forth moment, then we have*

$$I_{sub} \leq r(E(X_i X_i^T) + o(\mathbf{1})) \text{ almost surely as } n \to \infty$$

where $o(\mathbf{1})$ *is a matrix with all elements at order* $o(1)$.

The mVc algorithm is one of the algorithms proposed by Wang et.al(39). When implement the mVc procedure, each data line is assigned with a weight

$$\pi_i^{mVc} = \frac{|y_i - p_i(\hat{\boldsymbol{\beta}})|\|X_i\|}{\sum_{j=1}^{n} |y_j - p_j(\hat{\boldsymbol{\beta}})|\|X_j\|}.$$

Through simulation studies and real data analysis, Wang et.al(39) shows the algorithm has high estimation accuracy on slope parameters while the computational cost is endurable. However, we can show that, as one of the leveraging type algorithm, the expectation of the information matrix of the subdata extracted by using mVc procedure is still bounded by the subsize r.

Theorem 9. *For mVc algorithm, suppose covariate vector* $X_i, i = 1, \ldots, n$*, are generated independently from the same distribution* X*, which satisfies the following conditions:*

1. *There exists fixed constant* s*, such that* $E(||X_i||^6) < s$.
2. *For some constant* $B_2 > B_1 > 0$*,* $Prob(B_1 < ||X_i|| < B_2) > 0$.

Then

$$I_{sub} \leq \frac{r}{a}(E(||X||XX^T) + o(\mathbf{1})) \text{ almost surely as } n \to \infty$$

where $o(\mathbf{1})$ *is a matrix with all elements at order* $o(1)$.

These two theorems show that the information from the subdata sampled through existing subsampling strategies for logistic regression model is bounded from above by a term related to the size of the subdata even when $n \to \infty$. A new strategy proposed in the next section can break the restriction.

3.3 Extended IBOSS Algorithm for Logistic Regression Models

3.3.1 Optimal design and IBOSS algorithm for linear case

Optimal design focuses on finding the best way to allocate design points when design experiment under pre-specified models and estimation interests. The optimization idea, based on

information matrix can be borrowed to identify the most informative datalines from big data for estimating unknown parameters.

Recently statisticians have started thinking about utilizing optimal design theory in building up subsampling strategies. Wang et.al (37) proposed the novel IBOSS subsampling approach for linear models. Unlike the leveraging type strategies, which aim to adjust probability weight among datalines with ranking of importance, the IBOSS algorithm directly utilizes the structure of D-optimal design for linear models and then sequentially selects the boundary datalines on each dimension, which by theory are likely to be most informative. The subsampling procedure for IBOSS strategy in Wang et.al(37) is as follows:

Step 1 From full data set (X, Y), with subdata size r and covariate vector Z_i of m dimension for $i = 1, \cdots, m$, calculate $r_s = \lfloor \frac{r}{2m} \rfloor$

Step 2 Starting from $k = 1$, inside $\{(Y_i, X_i^T),\ i \in B\}$, pick r_s data rows with largest value, and r_s data lines with smallest value on the k-th dimension, put these datalines into the newly constructed subsample, and then erase them from the whole data. Repeat these steps for $k = 2, \ldots, m$. Combined all datalines picked as the newly constructed sub-sample.

The results by implementing this procedure with data simulated and real data show apparent improvement on estimating efficiency. Theoretically, Wang et.al(37) also shows that the information matrix of the subdata with IBOSS approach are not bounded by the subsample size r as long as $n \to \infty$. Readers are referred to (37) for more details. These exciting properties of the IBOSS algorithm shown both in theory and simulation drive us to think about the possibility

of extending the IBOSS strategy to logistic regression models. We first review a D-optimality result for logistic regressions.

Theorem 10 ((47)). *Under the logistic model (Equation 3.1) and certain conditions specified in (47), consider finding optimal points within the transformed design space* $\Xi = (1, x_{i1}, \cdots, x_{im-1}, c_i = X_i^T\beta)$. *Then the D-optimal design for estimating parameter* β *is* $\xi^* = \{(C_{l1}^*, 1/2^m)\&(C_{l2}^*, 1/2^m),\ l = 1, \cdots, 2^{m-1}\}$, *where* $C_{l1}^* = (1, a_{l,1}, \cdots, a_{l,m-1}, c^*)$ *and* $C_{l2}^* = (1, a_{l,1}, \cdots, a_{l,m-1}, -c^*)$.

- c^* minimize function $f(c)$, where $f(c) = c^{-2}(\Psi(c))^{-m-1}$, $\Psi(c) = \frac{[p'(c)]^2}{p(c)(1-p(c))}$, $p(c) = \frac{e^c}{1+e^c}$ and $c = X^T\beta$ for a covariate vector X.
- $a_{l,i}$ is the boundary of the design space in the i-th dimension, $i = 1, \cdots, m-1$

This theorem shows that, unlike the D-optimal design points for linear models which are simply located on the boundary of the design space, all the optimal design points for logistic regression models also have to meet the criteria that $c_i = \pm c^*$. And the last dimension of these points may not necessarily be on the boundary. The optimal value c^*, by formula, is a fixed constant which only concerns with the dimension and value of parameters. These differences indicates that directly borrowing the IBOSS procedure of linear case, which sequentially picks extreme datalines from each dimension, cannot be applied to logistic models. However we can use the similar strategy, i.e., select design points closest to the optimal design points and then balance it with the concern of computational cost to build up the new IBOSS strategy for logistic regression. Motivated by this idea, an extended IBOSS procedure for logistic regression is presented in the next subsection.

3.3.2 Overview of the New Algorithm

The IBOSS algorithm for logistic regression case is designed to find the most informative subsample from the full data by utilizing optimal design theory for logistic regression. The goal is to have the best possible estimation accuracy for all slope parameters while control the computational cost in a reasonable range. In building up the new IBOSS procedure, the D-optimality, which minimizes the determinant of the inverse of the information matrix of interested parameters, is used to characterize the datalines to build the most informative subsample. Since the D-optimal points for logistic models are on the boundary for the first $m-1$ dimension of the design space as well as having $c = X^T\hat{\beta}$ fixed at constant $\pm c^*$, to best meet the logic of picking datalines closest to this characterization, a two-stage subsampling strategy is employed. At the first stage, a relative large portion of the full data with their c value falling into a pre-specified neighbor of $\pm c^*$ is selected. For example, we can set up a $\delta > 0$ and then collect all the data lines (X_i, Y_i) with $\{i \mid min\{|c_i - c^*|, |c_i + c^*|\} \leq \delta\}$, where $c_i = X_i^T\hat{\beta}$. These datalines picked will be treated as the new database for the second-stage subsampling procedure. The second-stage procedure is similar to the IBOSS procedures proposed for linear models. The only difference is that we only do extreme value selection for the first $m-1$ dimension instead of all the m dimensions. One might notice that the calculation of c for each dataline and the optimal value c^* needs pre-knowledge on interest parameters β, which are unknown. To address this issue, a rough estimation of the parameter β with a small subdata from SRS is first conducted before this two stage procedure. The entire extended IBOSS procedure is described in details as follows:

1. Prefix a constant δ as maximum tolerance on the c dimension.
2. Given data set $\{(Y_i, X_i^T),\ i = 1, \cdots, n\}$, first do random sampling and pick r_0 sub-samples, fitting the data and get estimate $\hat{\beta}_{r_0}$.
3. Compute $c_i = X_i^T \hat{\beta}_{r_0}$, pick $B = \{i \mid \min\{|c_i - c^*|, |c_i + c^*|\} \leq \delta\}$.
4. Start from $k = 1$, inside $\{(Y_i, X_i^T),\ i \in B\}$, pick $\left\lceil \frac{r_1}{2(m-1)} \right\rceil$ data lines with largest value and $\left\lceil \frac{r_1}{2(m-1)} \right\rceil$ data lines with smallest value on the k-th dimension, put the datalines into the newly constructed subsample, erase these datalines from the whole data.
5. Repeat Step 4 for $k = 2, \ldots, m-1$. The newly constructed sub-sample is the subdata selected.

In the following subsection we will show some theoretical results which indicates that the extended IBOSS strategy maintain some good asymptotic properties of the IBOSS procedure for linear case.

3.3.3 Asymptotic Results

As we have shown, in the former section, one big restriction for the SRS procedure and mVc procedure is that the Fisher information matrix of the subdata is bounded by the size of the subsample r even n goes to infinity. Next we shall show that the extended IBOSS procedure proposed in this paper is not limited by this restriction.

Due to the complexity of logistic regression model, as well as the two stage procedures of the extended IBOSS strategy, it is extremely challenging to investigate the asymptotic property of the general m dimension case. Here we shall focus on a simple case: when the slope parameter

is two-dimensional, and it's corresponding slope covariate vector $Z = (x_1, x_2)^T$ is generated independently from a multivariate normal distribution with mean vector u and variance matrix Σ. Under this set up, we prove that, as long as $n_1 \to \infty$, all the elements in information matrix of the subdata goes to infinity with the new IBOSS subsampling procedure except the first diagonal element for intercept parameter. Here n_1 represents the size of remaining datalines after first stage procedure.

Theorem 11. *For logistic regression with* $X_i = (1, x_{i1}, x_{i2})^T$ *and* $\beta = (\beta_0, \beta_1, \beta_2)^T$*, assume that the two dimension slope covariate* $Z_i = (x_{i1}, x_{i2})$*,* $i = 1, \cdots, n$*, are generated independently from the same multivariate normal distribution* Z *with mean vector* u *and variance-covariance matrix* Σ*. Denote the correlation between* $c = X^T\beta$ *and* x_1 *as* ρ*, the pre-specified neighbor of* $\pm c^*$ *when implementing the first stage procedure of the new IBOSS strategy as* C*. Let* F_1 *and* F_2 *be the distribution function of* $\pm x_1$ *conditional on* $c = X^T\beta \in C$ *and* n_1 *represent the number of remaining datalines after first stage. Then if* $\rho \neq \pm 1$*, the elements for the slope parameters in the information matrix of the picked subdata with the new IBOSS procedure will go to* ∞ *as long as* $n_1 \to \infty$*.*

Remark 1. *Denote the information matrix of selected subdata as* I^{IBOSS}*, then it can be rewritten as*

$$\begin{aligned} I^{IBOSS} &= \sum_{i=1}^{n} \alpha_i \Psi(c_i) X_i X_i^T \\ &= \begin{pmatrix} \sum_{i=1}^{n} \alpha_i \Psi(c_i) & M_{12} \\ M_{12}^T & M_{22}. \end{pmatrix} \end{aligned} \tag{3.3}$$

The theorem is equal to saying that all the elements in M_{22} *will goes to infinity as* n_1 *goes to infinity.*

From this theorem, one can find that, unlike the SRS algorithm and the leveraging type mVc algorithm, the slope elements of the information matrix of sub data picked using extended IBOSS subsampling procedure under this two-dimensional case is not limited by any upper bound related to the subsample size. As long as the size of remaining data after first stage $n_1 \to \infty$, these elements in the information matrix can goes to infinity. In implementations, the δ is usually specified to approximately keep a certain percentage of the full data after the first stage. Thus, slope elements in the information will increase, as the full datasize n increases even with fixed r. This is actually a necessary condition to have the variance of slope parameters goes to zero as $n \to \infty$. Sure, this case studied here is the simplest case, but it still shows the great potential on estimation efficiency of the extended IBOSS procedure. Due to the complexity of the conditional distribution F_1 and F_2, it's hard to get any explicit forms for these infinity terms as well as their orders. However, the performance of the IBOSS optimal subsampling strategy can be demonstrated in various simulated scenarios which is presented in the next section.

3.4 Simulation Settings and Results

In this section, the new subsampling procedure are tested with various distribution settings for generating Z_i. There are several scenarios. Each scenario is targeting on answering one specific performance question. For most scenarios, the distribution settings to generate covariate vectors are the same. The distribution used to generate these Z_i's are:

- **MzNormal**

Multivariate-normal distribution with mean vector $u = (0, \cdots, 0)$ and variance-covariance matrix Σ.

- **NzNormal**

  Multivariate-normal distribution with mean vector $u = (1, \cdots, 1)$ and variance-covariance matrix Σ.

- **Mixed Normal**

  X_i is simulated randomly from mixed normal distribution $\frac{1}{2}N(u, \Sigma) + \frac{1}{2}N(-u, \Sigma)$, where vector $u = (1, \cdots, 1)$, for $i = 1, \cdots, n$.

- **T3**

  Multivariate T distribution with degree freedom 3 and variance-covariance matrix $\Sigma/10$.

These distributions have been used in Wang et.al(39) to demonstrate the efficiency of the mVc subsampling strategy. In this dissertation, the dimension size m might change from one scenario to another. The performance of new IBOSS strategy is compared with SRS strategy and mVc strategy in all scenarios, and sometimes, performance of the full data regression is also attached for reference. To be consistent with the simulation settings in Wang et.al(39), we assume that there is no intercept parameter β_0, unless otherwise specified.

Now we are ready to introduce the fisrt scenario, which is to test the performance of the new IBOSS strategy with smalll datasets.

3.4.1 Small Data Size Scenario

For small data size scenario, the X_i's are 1 x 7 vector generated randomly from the distribution settings mentioned above. In this scenario, $\Sigma_{ij} = 0.5$ for $i \neq j$ and $\Sigma_{ij} = 1$ for $i = j$, $n = 10000$, $r_0 = 200$, and β is 1 x 7 vector and true value $\beta_0 = (0.5, \cdots, 0.5)$. These settings are also exactly borrowed from (39). The size of subsample is fixed at $600, 700, 800, 900$ and 1000. Each case is simulated 1000 times.

The result of these scenarios with small sample size are shown in Figure 1. For all the subfigures, the x axis represents the size of the subdata, and the y axis is the average mean square error of all iterations. Under all the distribution settings, the performance of the new subsampling strategy is compared to the the commonly used SRS strategy and a newly developed leverage algorithm(mVc). Under Mixed Normal, MzNormal and NzNormal settings, the new IBOSS strategy and the mVc strategy outperforms the SRS strategy. The performance of the mVc algorithm is very similar to that of the new strategy with MixedNormal and MzNormal distribution. Sometimes the mVc strategy is slightly better than the extended IBOSS strategy under the MzNormal distribution. With NzNormal distributions, we see a difference between performance of the mVc strategy and the new IBOSS stategy, as the subdata size r grows larger.

When we compare these strategies under T3 distributions, the performance of the new strategy stands out. The performance of the new IBOSS strategy is significantly better than the performance of the other other two strategies, this means the subdata picked with new strategy is much more informative than the subdata picked with the mVc approach or the SRS approach.

3.4.2 Big Data Size Scenario

For the second scenario, we compare these three approaches on data relative large on size. Under all the distribution settings, $n = 500000$, $r_0 = 1000$, β is 1 x 7 vector and true value $\beta_0 = (0.5, \cdots, 0.5)$, variance-covariance structure Σ, $\Sigma_{ij} = 0.5$ for $i \neq j$ and $\Sigma_{ij} = 1$ for $i = j$. The size of final subdata is fixed at 2000, 5000 and 8000. 1000 iterations is ran for all strategies and all simulation settings.

The result of these runs are shown in Figure 2. All the notations on the figures are the same as these on the figures of the small size scenario. For the Mix Normal distribution, the performance of the mVc algorithm is almost equivalent to the commonly used SRS strategy. Also, one can find an obvious improvement on estimation accuracy when the new IBOSS strategy is used. Under the NzNormal distributions, the mVc strategy slightly outperforms the new strategy. However, the performance difference becomes smaller and smaller as subsize r grows larger and finally are merely can be seen with $r = 8000$. Under the T3 settings, the performance of the new subsampling strategy performs even better. The average MSE of extended IBOSS can be less than $\frac{1}{4}$ of the average MSE of mVc strategy or SRS strategy. This means the subsample picked with the new strategy is more than four times as efficient as the mVc and SRS strategies.

3.4.3 Increas in Estimation Efficiency as n Grows

Intuitively, the new IBOSS approach tries to pick points close to D-optimal points of logistic model. As we showed in theory, the subdata should become more informative as n grows larger. This indicates that the new algorithm should enjoy an improve on estimation accuracy as n

increases, even when the size of subdata r is fixed. In this subsection, simulations are conducted to see whether we can observe this trend. Here we stick to the distribution and parameter settings from previous example. The parameters β is fixed as a 9 dimensional vector of 0.5, while the number of datalines for the whole dataset is picked as 200000, 400000, 600000, 800000. The size of the subdata is fixed at 5000. The log scale average MSEs are shown in Figure 3.

In this figure, one can see that for MixedNormal, MzNormal and NzNormal distribution, the estimation accuracy increase on the new IBOSS approach and mVc approach can be barely seen as n increases. This is consistent with the result by (37) for the linear model. However, for T3 distributions, a clear trend of increasing estimation efficiency as data size grow larger can be observed for the new IBOSS strategy. The average MSE from regression with full data is also presented for reference in this figure.

3.4.4 Some Insights on Determining δ

In all the simulation scenarios discussed above, the δ is pre-specified for the first stage filtering of data. The value of δ, along with full data size n and distributions to generate Z_i, will largely affect the estimation accuracy of the new algorithm. Unfortunately, there is no theoretical result to help us find the proper delta when handling any given dataset. However, we can use simulations to give us a rough idea about the proper range for setting up δ value. In this part, we still use the distribution settings from the top scenarios. The slope parameter $(\beta_1, \cdots, \beta_9)$ is again set as a 9 dimensional parameter with value 0.5. Since the δ criteria

directly works on $c = X^T\beta$, to make our simulation more general, we consider the following 3 cases:

- Balanced case: Intercept β_0 is not considered in the model, or to say known as 0. In this case, with Z_i generated from the top four distributions, the c value will be center around 0 for MixedNormal, MzNormal and T3 distributions. The c value for Nz Normal distribution will be centered around some positive number, depending on the number of covariates.
- Right shift case: Intercept $\beta_0 = 2$, then the distribution center for c is shifted to the right hand side of the center of the balanced case.
- Left shift case: Intercept $\beta_0 = -2$, then the distribution center for c is shifted to the left hand side of the center of the balanced case.

For all these 3 cases, at first stage of the IBOSS procedure, we pick certain percent of the full data according to the distance criteria we set up with an increasing order. The percentage tested are $c(0.25, 0.35, \cdots, 0.75)$. The full data size is $n = 500000$. For each test setting in this scenario, 200 iterations are used and average MSE of the slope parameters are calculated. The result for all these cases are shown in the following Figure 4, Figure 5, and Figure 6.

From the top figures, one can easily find that, regardless of the distributions of covariates Z_i and the shift of center of c, the 30% extraction rate for first stage procedure seems to have robust good performance across all distributions. This means that when we try to pick

a proper value for δ, we can pick a δ which filtered out around 70% percent of full data and keep around 30% according to the distance criteria we set up. Also for the T3 distribution, the different quantiles tested generally perform well. Also, a slight gain on accuracy can be obtained as the percentage grows higher, which indicates that there is truly some difference from one distribution to another.

3.5 Further Studies: Directional Derivative Subsampling Approach for Asymmetric Data Case

Good performance of the IBOSS algorithm was demonstrated by simulated data in the previous section. However, in real cases, if the range of $c = X^T\beta$ is not symmetric, for example, when X only takes positive values and β are also a positive vector, it might be impossible for us to simultaneously select data points within neighbors of $\pm c^*$. Thus the efficiency of the extended IBOSS procedure might not be guaranteed. With these un-symmetric cases, some alternative subsampling procedures should be used. The directional derivative theory in optimal design theory inspired us to propose the following directional derivative optimal subsampling strategy (DDOSS) in order to handle subdata selection cases under this situation.

Definition 12. *Denote the optimal design for logistic model of* m *dimension with optimal criteria* Ψ *as* ξ^* *and it's information matrix as* I_{ξ^*}*. Thus for a covariate vector* $X = (1, x_1, \cdots, x_m)$*, the directional derivative of* X *to the optimal design* ξ^* *with optimal criteria* Ψ *is defined as*

$$D_X = \lim_{\alpha \to 0} \frac{\Psi((1-\alpha)I_{\xi^*} + (\alpha I_X)) - \Psi(I_{\xi^*})}{\alpha}$$

The directional derivative D_X can help us evaluate the information contained in data line (X, Y), compared to points in the optimal design, ξ^*. The larger the form D_X is, the closer dataline X is to the best design points. Thus, a very natural idea for picking out subsamples is to pick out datalines with the largest D_X values under the commonly used D-optimal criteria. The DDOSS procedure is given as follows:

1. Given data set $\{(Y_i, X_i^T),\ i = 1, \cdots, n\}$, first do random sampling and pick r_0 sub-samples, fitting the data and get estimate $\hat{\boldsymbol{\beta}}_{r_0}$.
2. Numerically find the optimal design ξ^* under the given range of X values for all dimensions with D-optimal criteria.
3. Compute $m_i = tr(I_\xi^{-1}(I_{X_i} - I_\xi))$ for each dataline X_i.
4. Pick out (X_i, Y_i) with the highest r $m_i's$ as the final subdata.

Here m_i is just the simplified version of the directional derivative D_{X_i} under D-optimal criteria. The whole subsampling procedure is independent of the response value Y. Thus the asymptotic consistency is not a problem. In order to demonstrate the performance of DDOSS, we will test it with a simulation study by simulating data from the following distributions:

1. T distribution case: Generate data from $\frac{\textbf{T-dist}(\text{df=3})}{5} + 3$;
2. The Exponential $\mathbf{Exp}(1)$ distribution;
3. Mixed Normal distribution: Generate data from $0.5N(2, \Sigma) + 0.5N(3, \Sigma)$;
4. The Normal distribution $N(2, \Sigma)$ (Normal distribution I);

5. The Normal distribution $N(3, \Sigma)$ (Normal distribution II).

Here, the variance matrix Σ is taken as $diag(2, \cdots, 2)$. All of the tested distributions are borrowed from the former simulation settings except the $Exp(1)$ distribution, however, the mean parameter and variance matrix Σ is revised to make sure all dimension of X is falling into a range around $[0, 6]$. Under the $Exp(1)$ distribution, the covariate vector Z_i's are i.i.d generated from $\mathbf{E}xp(1)$. The logistic model is 3 dimension with true parameter $\beta = (0.5, 0.5, 0.5)$, assuming there is no intercept parameter β_0. The D-optimal design is then built up using the β estimate from the initial SRS sample for each iteration. Subsample size r is fixed as 5000, with initial simple random sampling size $r_0 = 1000$. The average mean square error from 1000 iterations with SRS, mVc and DDOSS is shown in Table Table XIII.

TABLE XIII

PERFORMANCE OF SUBSAMPLING PROCEDURES UNDER ASYMMETRIC CASES

Distribution	N	MSE_{mVc}	MSE_{SRS}	MSE_{DDOSS}
T	400000	$1.43x10^{-2}$	$7.73x10^{-2}$	$3.91x10^{-3}$
Mixed	400000	$2.37x10^{-3}$	$1.16x10^{-2}$	$2.07x10^{-3}$
Normal I	400000	$1.14x10^{-3}$	$3.10x10^{-3}$	$9.80x10^{-4}$
Normal II	400000	$2.62x10^{-3}$	$9.21x10^{-3}$	$1.30x10^{-3}$
EXP	400000	$1.37x10^{-3}$	$1.59x10^{-3}$	$2.10x10^{-4}$

From the table, the DDOSS algorithm obvious beats SRS and mVc with all ditribution tested. Meanwhile, the computational cost to find the optimal design with given β estimate only depends on the dimension of β. Thus this cost will be independent of full data set size as well as the subdata set size. If we ignore the computational cost for this part, the computational cost of the DDOSS is still endurable, compare to the mVc procedure. The average computation time of all iteration in seconds under some of the tested distributions is shown below in Table XIV:

TABLE XIV

COMPUTATIONAL COST OF DIFFERENT APPROACHES WITH DIFFERENT DATA SIZES

Distribution	N	mVc	SRS	DDOSS
T	400000	$2.32\text{x}10^{-3}$	$1.68\text{x}10^{-4}$	$3.58\text{x}10^{-3}$
Mixed	400000	$1.80\text{x}10^{-3}$	$1.72\text{x}10^{-4}$	$2.07\text{x}10^{-3}$
Normal I	400000	$9.12\text{x}10^{-4}$	$1.32\text{x}10^{-4}$	$1.98\text{x}10^{-3}$
Normal II	400000	$8.44\text{x}10^{-4}$	$1.00\text{x}10^{-4}$	$1.99\text{x}10^{-3}$

3.6 Discussion

In this chapter, the IBOSS subsampling strategy for linear models was extended to the logistic regression models. Under the framework of the logistic regression models, the existing popular subsampling approaches, the mVc strategy and the SRS strategy were analyzed and

an upper bound of the information from these strategies were derived. For the extended IBOSS procedure proposed in this dissertation, assuming the covariates are from two dimensional multivariate normal distribution, we showed that the diagonal elements (corresponding to the slope parameters) of the information matrix goes to infinity, even when subsample size is fixed as long as full sample size goes to infinity. This gives some theoretical justification of the proposed approach. We expect such results also hold for the general case, which are confirmed by simulation results. However, it would be rather challenging, if not impossible, to derive such asymptotically theoretical conclusions due to model complexity. The performance of the new IBOSS approach is also demonstrated with simulated data under various distribution settings for $\mathsf{Z_i}$. The new strategy generally outperforms the simple random sampling strategy and at least does no worse than the newly developed leveraging algorithm under most distribution settings. For the T3 distribution case, the extended IBOSS procedure is clearly superior to the other procedures discussed in this dissertation.

When the range of c value is extremely asymmetric, and the IBOSS procedure cannot be properly implemented, an alternative subsampling approach(DDOSS) based on directional derivative was also provided. The DDOSS approach clearly outperformed the other approaches with the simulated datasets. The computational cost of DDOSS is higher than the cost of the SRS and mVc procedures, but it's still endurable.

However, the new strategies proposed are still under development. The asymptotic property for the DDOSS approach and extended IBOSS approach is not well understood for the general case of $\mathfrak{m}$-dimensional logistic model . There also seems to be many potential related research

topics worth investigating. For example, one may study how to develop a general strategy to find suitable δ defined in this paper, how to combine the idea of this approach with penalty functions for regularized regressions, and so forth.

In this information explosion age, big data with complex data structure can be easily obtained via various sources. While it provides us more valuable information, the computational cost of analyzing them can be way much out of capacity and endurability. Efforts on developing subsampling strategies has greatly improved the quality of the subdata and saved tremendous computational resources. However, subsampling strategies for nonlinear models, like the logistic regression model considered in this chapter, are still not well developed. We hope that this work can stimulate more ideas and more researches in this direction.

Figure 1. Comparison of subsampling algorithms for the small size scenario

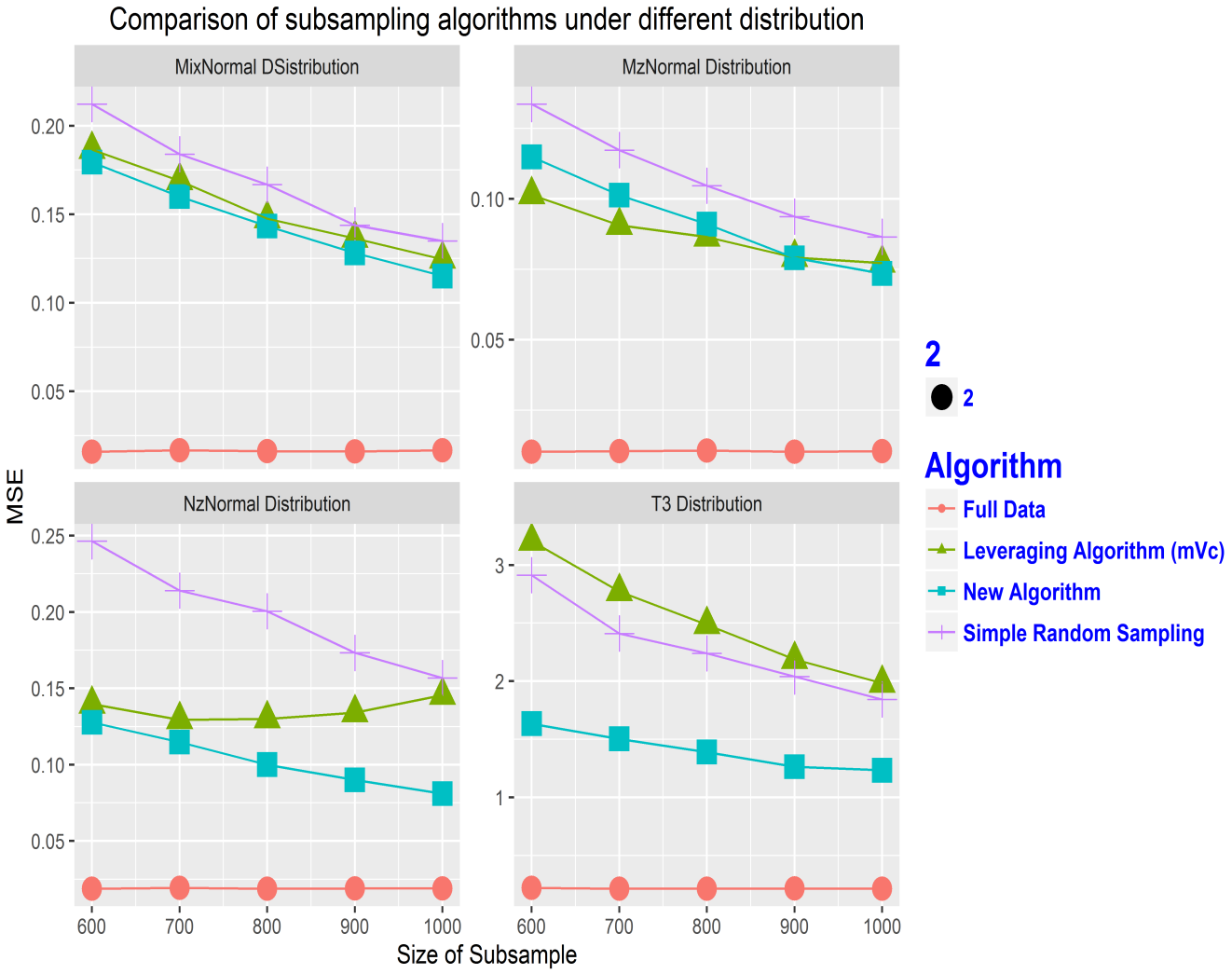


Figure 2. Comparison of subsampling algorithms for the large size scenario

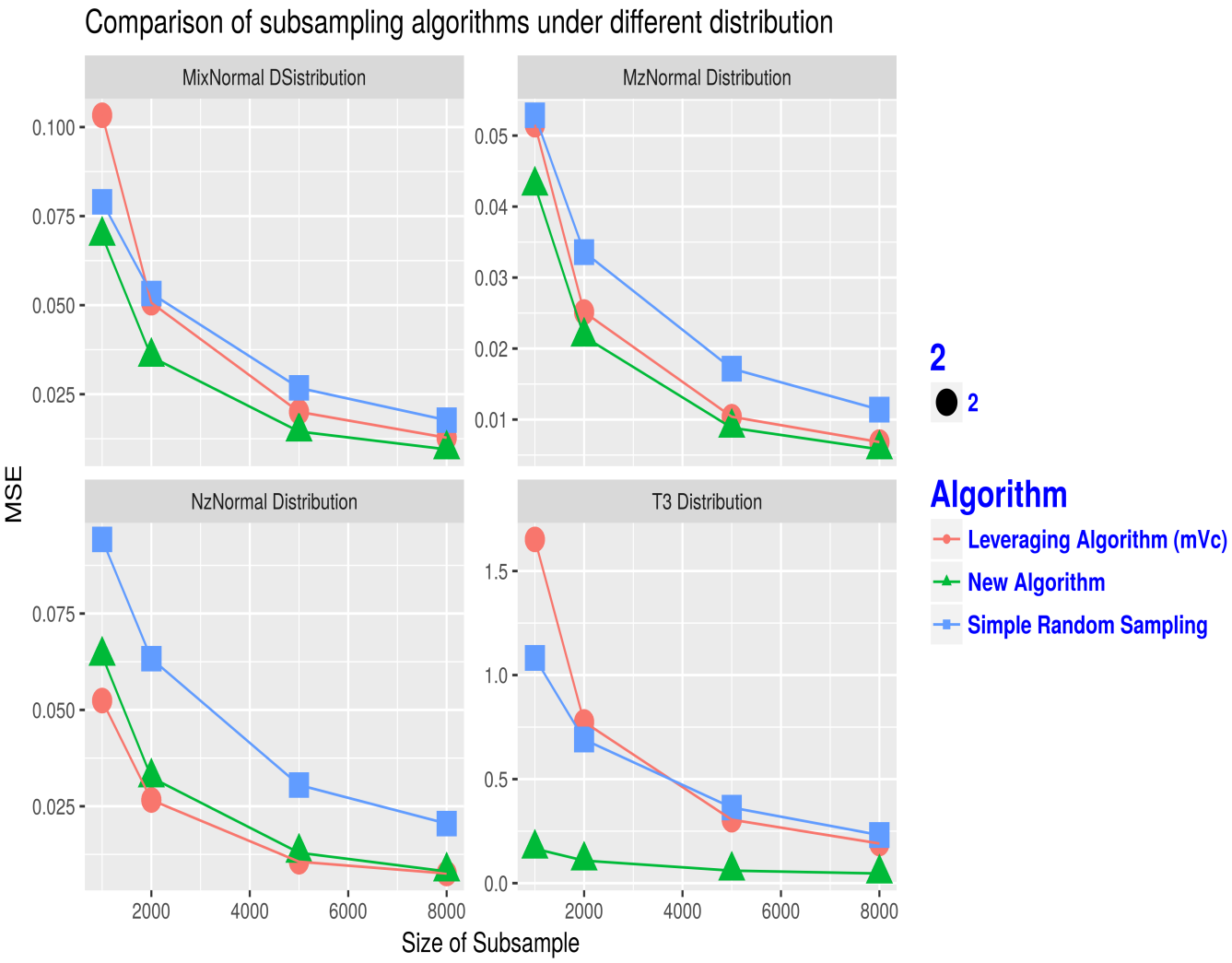


Figure 3. Comparison of subsampling algorithms for different full data sizes

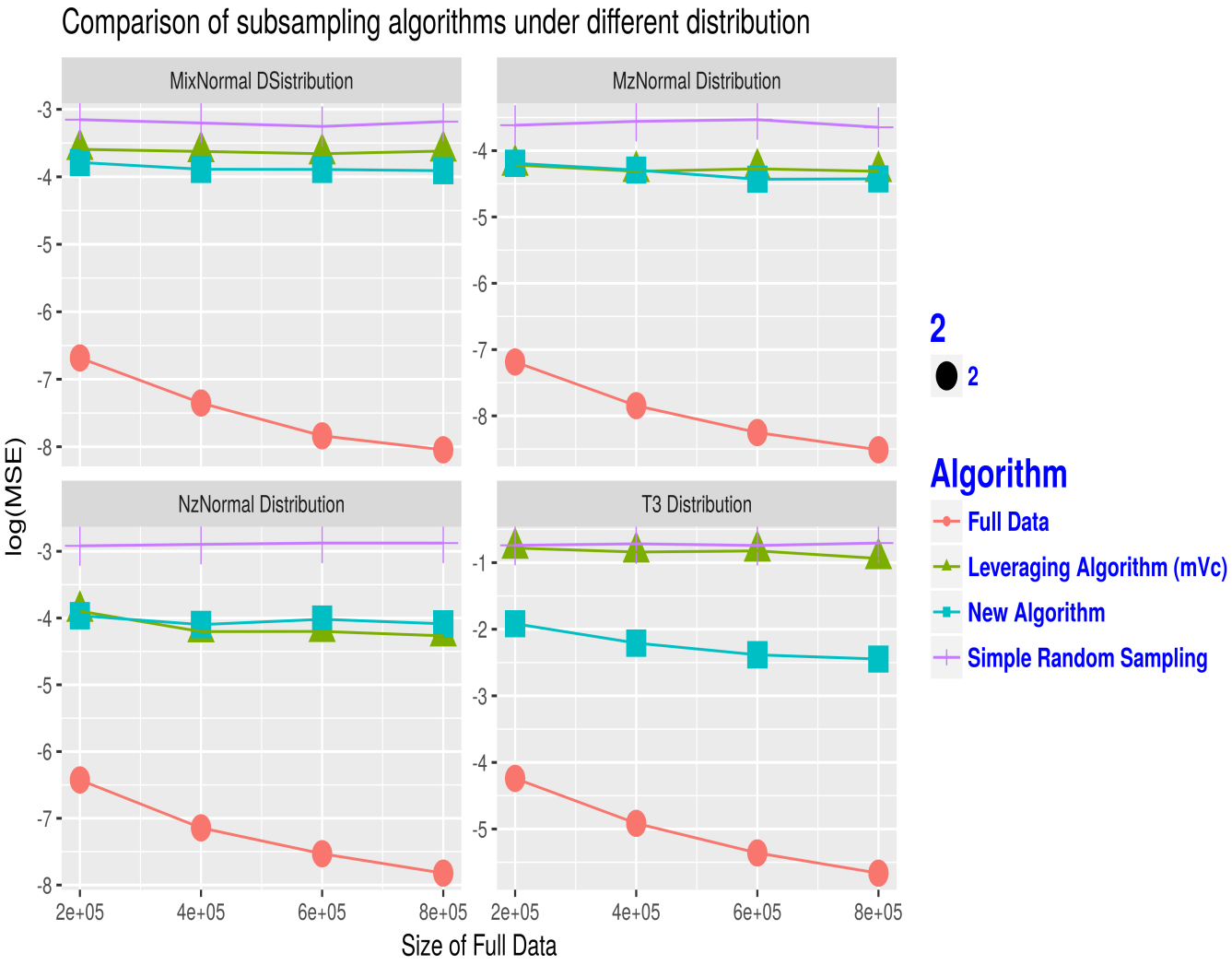


Figure 4. For new algorithm, comparison of efficiency under different percentages: Balanced case

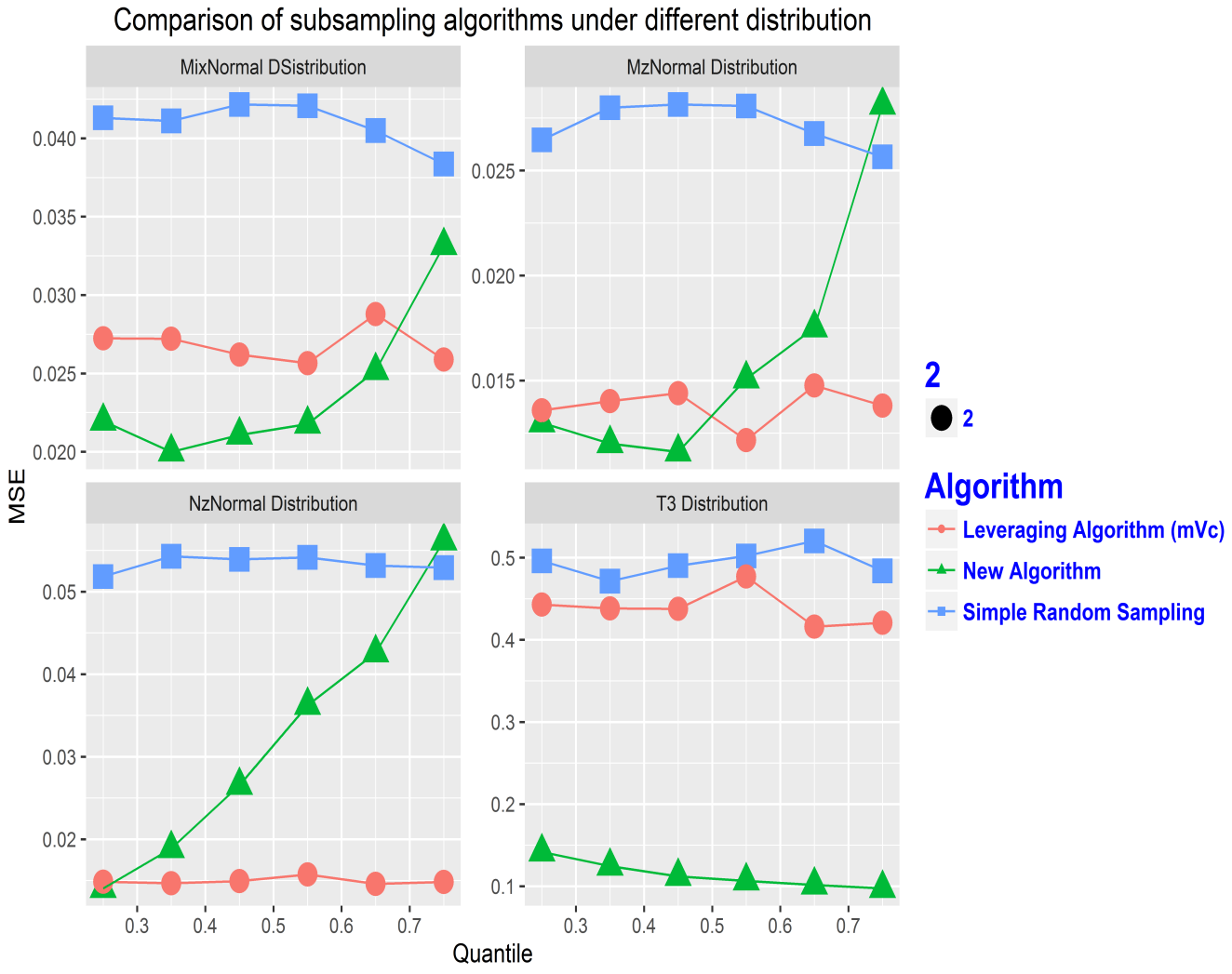


Figure 5. For new algorithm, comparison of efficiency under different percentage: Right skewed case

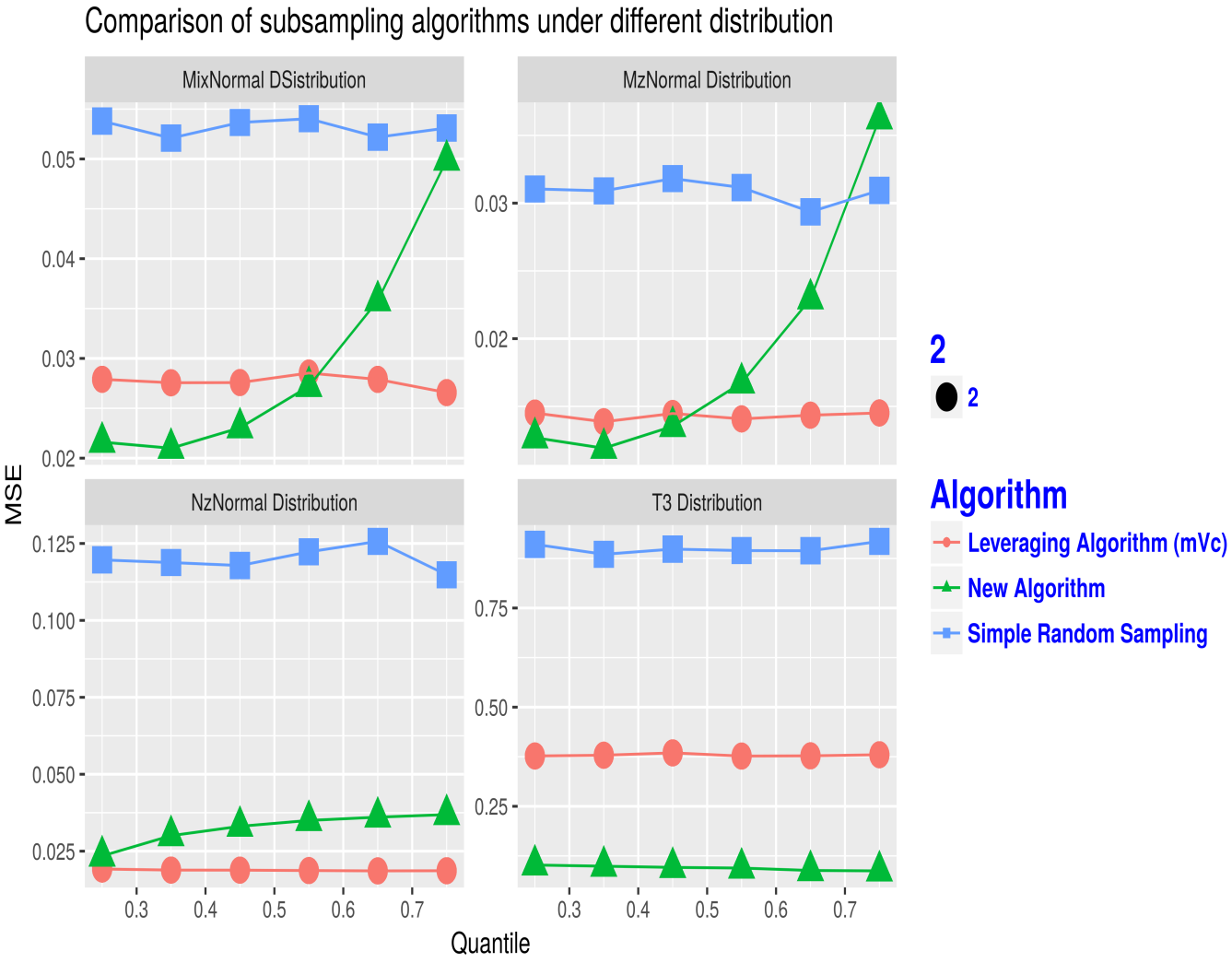
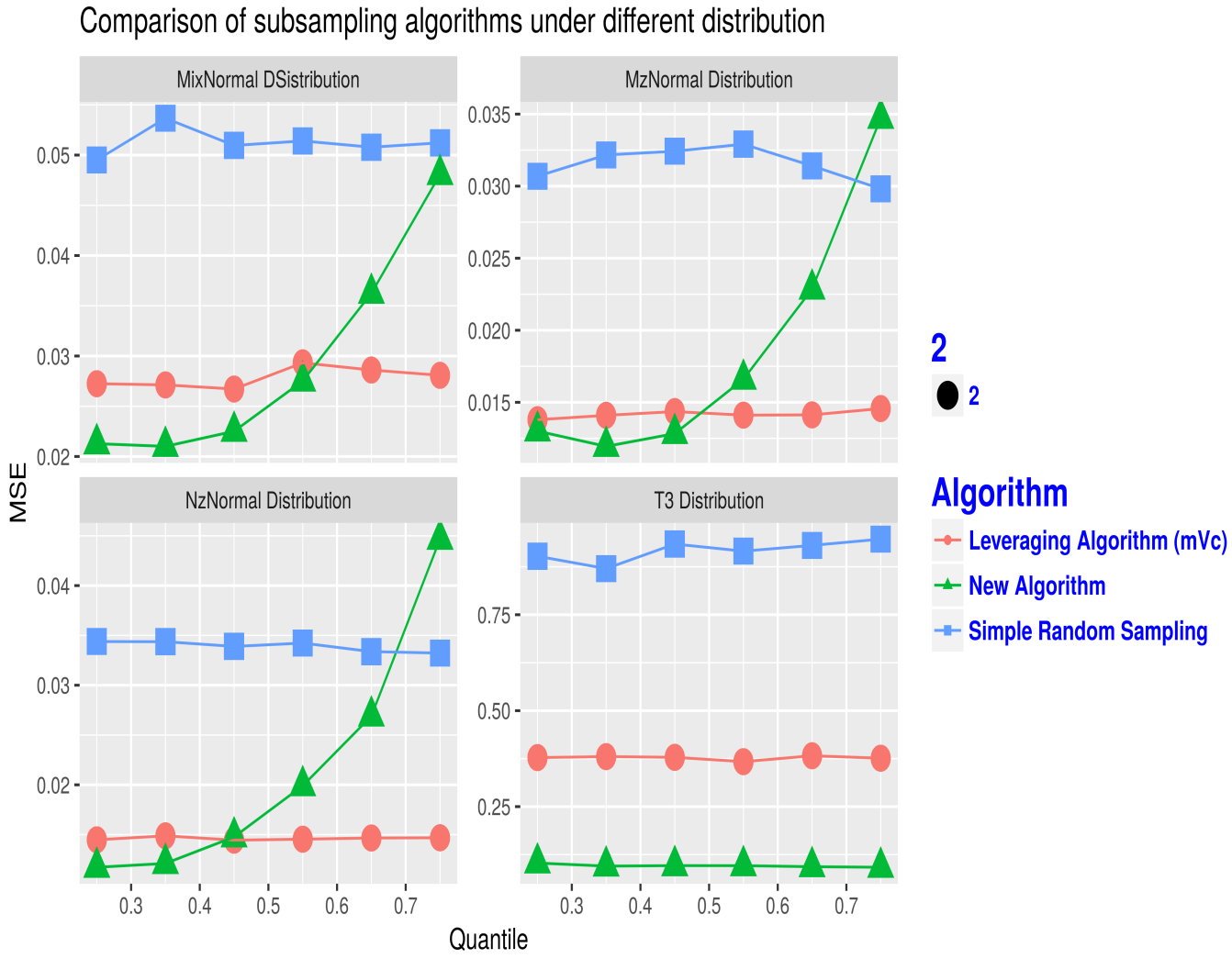


Figure 6. For new algorithm, comparison of efficiency under different percentage: Left skewed case



CHAPTER 4

SUPPLEMENTAL MATERIALS FOR THE EXTENDED IBOSS APPROACH

Proof of Theorem 7. By the definitions of M_X and V_c in Theorem 1, Matrix V can be written as

$$\begin{aligned} V &= \left(\sum_{i=1}^{n} \frac{1}{n} p_i(\hat{\boldsymbol{\beta}})(1-p_i(\hat{\boldsymbol{\beta}}))X_i X_i^T\right)^{-1} \left(\frac{1}{rn^2}\sum_{i=1}^{n} \frac{(y_i - p_i(\hat{\boldsymbol{\beta}}))^2 X_i X_i^T}{\pi_i}\right) \left(\sum_{i=1}^{n} \frac{1}{n} p_i(\hat{\boldsymbol{\beta}})(1-p_i(\hat{\boldsymbol{\beta}}))X_i X_i^T\right)^{-1} \\ &= \frac{1}{r}\left(\sum_{i=1}^{n} p_i(\hat{\boldsymbol{\beta}})(1-p_i(\hat{\boldsymbol{\beta}}))X_i X_i^T\right)^{-1} \left(\sum_{i=1}^{n} \frac{(y_i - p_i(\hat{\boldsymbol{\beta}}))^2 X_i X_i^T}{\pi_i}\right) \left(\sum_{i=1}^{n} p_i(\hat{\boldsymbol{\beta}})(1-p_i(\hat{\boldsymbol{\beta}}))X_i X_i^T\right)^{-1}. \end{aligned}$$

Then the information matrix I_{sub} can be written as

$$
\begin{aligned}
I_{sub} &= V^{-1} \\
&= r\left(\sum_{i=1}^{n} p_i(\hat{\beta})(1-p_i(\hat{\beta}))X_i X_i^T\right)\left(\sum_{i=1}^{n}\frac{(y_i-p_i(\hat{\beta}))^2 X_i X_i^T}{\pi_i}\right)^{-1}\left(\sum_{i=1}^{n} p_i(\hat{\beta})(1-p_i(\hat{\beta}))X_i X_i^T\right) \\
&= r\left[\frac{p_1(\hat{\beta})(1-p_1(\hat{\beta}))X_1\sqrt{\pi_1}}{y_1-p_1(\hat{\beta})},\cdots,\frac{p_n(\hat{\beta})(1-p_n(\hat{\beta}))X_n\sqrt{\pi_n}}{y_n-p_n(\hat{\beta})}\right]\begin{bmatrix}\frac{(y_1-p_1(\hat{\beta}))X_1^T}{\sqrt{\pi_1}}\\ \vdots \\ \frac{(y_n-p_n(\hat{\beta}))X_n^T}{\sqrt{\pi_n}}\end{bmatrix} \\
&\left(\left[\frac{(y_1-p_1(\hat{\beta}))X_1}{\sqrt{\pi_1}},\cdots,\frac{(y_n-p_n(\hat{\beta}))X_n}{\sqrt{\pi_n}}\right]\begin{bmatrix}\frac{(y_1-p_1(\hat{\beta}))X_1^T}{\sqrt{\pi_1}}\\ \vdots \\ \frac{(y_n-p_n(\hat{\beta}))X_n^T}{\sqrt{\pi_n}}\end{bmatrix}\right)^{-1} \\
&\left[\frac{(y_1-p_1(\hat{\beta}))X_1}{\sqrt{\pi_1}},\cdots,\frac{(y_n-p_n(\hat{\beta}))X_n}{\sqrt{\pi_n}}\right]\begin{bmatrix}\frac{p_1(\hat{\beta})(1-p_1(\hat{\beta}))X_1^T\sqrt{\pi_1}}{y_1-p_1(\hat{\beta})}\\ \vdots \\ \frac{p_n(\hat{\beta})(1-p_n(\hat{\beta}))X_n^T\sqrt{\pi_n}}{y_n-p_n(\hat{\beta})}\end{bmatrix}.
\end{aligned}
$$

Set $W = \text{diag}(\frac{(y_1 - p_1(\hat{\boldsymbol{\beta}}))}{\sqrt{\pi_1}}, \cdots, \frac{(y_n - p_n(\hat{\boldsymbol{\beta}}))}{\sqrt{\pi_n}})$. Then I_{sub} can be re-written as

$$I_{\text{sub}} = r \left[\frac{p_1(\hat{\boldsymbol{\beta}})(1 - p_1(\hat{\boldsymbol{\beta}}))X_1\sqrt{\pi_1}}{y_1 - p_1(\hat{\boldsymbol{\beta}})}, \cdots, \frac{p_n(\hat{\boldsymbol{\beta}})(1 - p_n(\hat{\boldsymbol{\beta}}))X_n\sqrt{\pi_n}}{y_n - p_n(\hat{\boldsymbol{\beta}})} \right] WX(X^T W^2 X)^{-1} X^T W$$

$$\begin{bmatrix} \frac{p_1(\hat{\boldsymbol{\beta}})(1-p_1(\hat{\boldsymbol{\beta}}))X_1^T\sqrt{\pi_1}}{y_1 - p_1(\hat{\boldsymbol{\beta}})} \\ \vdots \\ \frac{p_n(\hat{\boldsymbol{\beta}})(1-p_n(\hat{\boldsymbol{\beta}}))X_n^T\sqrt{\pi_n}}{y_n - p_n(\hat{\boldsymbol{\beta}})} \end{bmatrix}$$

$$= r \left[\frac{p_1(\hat{\boldsymbol{\beta}})(1 - p_1(\hat{\boldsymbol{\beta}}))X_1\sqrt{\pi_1}}{y_1 - p_1(\hat{\boldsymbol{\beta}})}, \cdots, \frac{p_n(\hat{\boldsymbol{\beta}})(1 - p_n(\hat{\boldsymbol{\beta}}))X_n\sqrt{\pi_n}}{y_n - p_n(\hat{\boldsymbol{\beta}})} \right] \mathbf{Proj}_{WX} \begin{bmatrix} \frac{p_1(\hat{\boldsymbol{\beta}})(1-p_1(\hat{\boldsymbol{\beta}}))X_1^T\sqrt{\pi_1}}{y_1 - p_1(\hat{\boldsymbol{\beta}})} \\ \vdots \\ \frac{p_n(\hat{\boldsymbol{\beta}})(1-p_n(\hat{\boldsymbol{\beta}}))X_n^T\sqrt{\pi_n}}{y_n - p_n(\hat{\boldsymbol{\beta}})} \end{bmatrix},$$

where $X = \begin{bmatrix} X_1^T \\ \vdots \\ X_n^T \end{bmatrix}$. Define $B_{WX} = \begin{bmatrix} w_1 X_1^T & \cdots \\ \vdots & \\ \cdots & w_n X_n^T \end{bmatrix}$, where w_i is the i-th diagonal element in W. Clearly the columns of WX are in the column space of B_{WX}. Thus, we have

$$I_{sub} <= r[\frac{p_1(\hat{\beta})(1-p_1(\hat{\beta}))X_1\sqrt{\pi_1}}{y_1 - p_1(\hat{\beta})}, \cdots, \frac{p_n(\hat{\beta})(1-p_n(\hat{\beta}))X_n\sqrt{\pi_n}}{y_n - p_n(\hat{\beta})}]\mathbf{Proj}_{B_{WX}} \begin{bmatrix} \frac{p_1(\hat{\beta})(1-p_1(\hat{\beta}))X_1^T\sqrt{\pi_1}}{y_1-p_1(\hat{\beta})} \\ \vdots \\ \frac{p_n(\hat{\beta})(1-p_n(\hat{\beta}))X_n^T\sqrt{\pi_n}}{y_n-p_n(\hat{\beta})} \end{bmatrix}$$

$$= r[\frac{p_1(\hat{\beta})(1-p_1(\hat{\beta}))\sqrt{\pi_1}X_1}{y_1 - p_1(\hat{\beta})}, \cdots, \frac{p_n(\hat{\beta})(1-p_n(\hat{\beta}))\sqrt{\pi_n}X_n}{y_n - p_n(\hat{\beta})}] \begin{bmatrix} X_1^T(X_1X_1^T)^{-1}X_1 & \cdots \\ \vdots & \\ \cdots & X_n^T(X_nX_n^T)^{-1}X_n \end{bmatrix}$$

$$\begin{bmatrix} \frac{X_1^T p_1(\hat{\beta})(1-p_1(\hat{\beta}))\sqrt{\pi_1}}{y_1-p_1(\hat{\beta})} \\ \vdots \\ \frac{X_n^T p_n(\hat{\beta})(1-p_n(\hat{\beta}))\sqrt{\pi_n}}{y_n-p_n(\hat{\beta})} \end{bmatrix}$$

$$= r(\sum_{i=1}^{n} \frac{p_i^2(\hat{\beta})(1-p_i(\hat{\beta}))^2\pi_i}{(y_i - p_i(\beta))^2} X_i X_i^T) \leq r(\sum_{i=1}^{n} \pi_i X_i X_i^T).$$

Here, the inequalities are under the context of Lowering ordering. □

Proof of Theorem 8. Notice that $\pi_i = \frac{1}{n}$ for $i = 1, \cdots, n$. The conclusion follows by applying Theorem 7 and the strong law of large numbers. □

Proof of Theorem 9. Applying Theorem 7 and $\pi_i^{mVc} = \frac{|y_i - p_i(\hat{\beta})|||X_i||}{\sum_{j=1}^n |y_j - p_j(\hat{\beta})|||X_j||}$, we have

$$\begin{aligned} I_{sub} &= r(\sum_{i=1}^n \frac{|y_i - p_i(\hat{\beta})|||X_i||}{\sum_{j=1}^n |y_j - p_j(\hat{\beta})|||X_j||} X_i X_i^T) \\ &= (\frac{r}{\sum_{j=1}^n |y_j - p_j(\hat{\beta})|||X_j||} \sum_{i=1}^n ||X_i|||y_i - p_i(\beta)|X_i X_i^T) \\ &\leq r(\frac{\sum_{i=1}^n ||X_i||X_i X_i^T}{\sum_{\{y_j=1\}} |1 - p_j(\hat{\beta})|||X_j|| + \sum_{\{y_j=0\}} |p_j(\hat{\beta})|||X_j||}). \end{aligned} \tag{4.1}$$

Notice that $E((||X_i||x_{ij}x_{ij'})^2) \leq E(||X||^6) < s$ for $j, j' = 1, \cdots, p$ and $X_i, i = 1, \ldots, n$ are identically independent distributed. By strong law of large numbers, we have

$$\frac{\sum_{i=1}^n ||X_i||X_i X_i^T}{n} = E(||X||XX^T) + o(\mathbf{1}) \text{ almost surely as } n \to \infty. \tag{4.2}$$

For a given set of parameter $\boldsymbol{\beta}$,

$$\begin{aligned} &\frac{\sum_{\{y_j=1\}} |1 - p_j(\hat{\beta})|||X_j|| + \sum_{\{y_j=0\}} |p_j(\hat{\beta})|||X_j||}{n} \\ &= \frac{\sum_{\{y_j=1\}} |1 - p_j(\beta) + p_j(\beta) - p_j(\hat{\beta})|||X_j|| + \sum_{\{y_j=0\}} |p_j(\hat{\beta}) - p_j(\beta) + p_j(\beta)|||X_j||}{n} \\ &\geq \frac{\sum_{\{y_j=1\}} |1 - p_j(\beta)|||X_j|| + \sum_{\{y_j=0\}} |p_j(\beta)|||X_j||}{n} - \frac{\sum_{j=1}^n |p_j(\hat{\beta}) - p_j(\beta)|\ ||X_j||}{n}. \end{aligned}$$

Notice that $p_j(\boldsymbol{\beta}) = \frac{e^{X_j^T \beta}}{1+e^{X_j^T \beta}}$ and $p_j(\hat{\boldsymbol{\beta}}) = \frac{e^{X_j^T \hat{\beta}}}{1+e^{X_j^T \hat{\beta}}}$, we have

$$|p_j(\boldsymbol{\beta}) - p_j(\hat{\boldsymbol{\beta}})| = \frac{e^{X_j^T \bar{\beta}}}{(1 + e^{X_j^T \bar{\beta}})^2} |X_j^T \hat{\boldsymbol{\beta}} - X_j^T \boldsymbol{\beta}| \leq ||X_j|||\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}||$$

for any given X, where $X^T\bar{\beta}$ is a value between $X^T\beta$ and $X^T\hat{\beta}$. Consequently, we have

$$\begin{aligned}
&\frac{\sum_{\{y_j=1\}}|1-p_j(\hat{\beta})|\|X_j\|+\sum_{\{y_j=0\}}|p_j(\hat{\beta})|\|X_j\|}{n} \\
&\geq \frac{\sum_{\{y_j=1\}}|1-p_j(\beta)|\|X_j\|+\sum_{\{y_j=0\}}|p_j(\beta)|\|X_j\|}{n}-\|\hat{\beta}-\beta\|\ \frac{\sum_{j=1}^{n}\|X_j\|}{n} \\
&= \frac{\sum_{\{y_j=1\}}|1-p_j(\beta)|\|X_j\|+\sum_{\{y_j=0\}}|p_j(\beta)|\|X_j\|}{n}-o(\frac{\sum_{i=1}^{n}\|X_i\|}{n}) \\
&= \frac{\sum_{\{y_j=1\}}|1-p_j(\beta)|\|X_j\|+\sum_{\{y_j=0\}}|p_j(\beta)|\|X_j\|}{n}-o(E(\|X_i\|)) \\
&= \frac{\sum_{i=1}^{n}(1-p_i(\beta))^{y_i}(p_i(\beta))^{1-y_i}\|X_i\|}{n}+o(1)
\end{aligned} \tag{4.3}$$

almost surely as $n\to\infty$.

By applying the law of large numbers to the term $\sum_{i=1}^{n}(1-p_i(\beta))^{y_i}(p_i(\beta))^{1-y_i}\|X_i\|$, we have

$$\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n}(1-p_i(\beta))^{y_i}(p_i(\beta))^{1-y_i}\|X_i\| &= E((1-p(\beta))^{y}(p(\beta))^{1-y}\|X\|)+o(1) \\
&= 2E(p(\beta)(1-p(\beta))\|X\|)+o(1)
\end{aligned} \tag{4.4}$$

where y is a Bernoulli trial with probability of success $p(\beta)=\frac{e^{X^T\beta}}{1+e^{X^T\beta}}$. Next we shall show that $2E(p(\beta)(1-p(\beta))\|X\|)>a$ for some constant $a>0$. Notice that when $X\in[B_1,B_2]$, $|X^T\beta|\leq\sqrt{\|X\|^2\|\beta\|^2}$, thus $p(\beta)(1-p(\beta))>k$ for some positive constant k. Let $I_{B_1\leq X\leq B_2}=1$ when $X\in[B_1,B_2]$ and 0 otherwise. We have

$$\begin{aligned}
E(p(\beta)(1-p(\beta))\|X\|) &\geq E(p(\beta)(1-p(\beta))\|X\|I_{B_1\leq X\leq B_2}) \\
&\geq kB_1\text{Prob}(B_1<\|X\|<B_2) \\
&\geq a
\end{aligned} \tag{4.5}$$

for some constant $a > 0$. By applying (Equation 4.1), (Equation 4.2), (Equation 4.3), (Equation 4.4), and (Equation 4.5), we have

$$I_{sub} \leq \frac{r}{a}(E(\|X\|XX^T) + o(\mathbf{1})) \text{ almost surely as } n \to \infty.$$

□

Proof of Theorem 11. To prove this theorem, first we need to derive the explicit form of pdf $f_1(x_1)$ and $f_2(v_1)$ for F_1 and F_2.

Since $Z \sim N(\mu, \Sigma)$, then $Z^t = (x_1, x^T\beta) \sim N((u_1, u_c)^T, \Sigma_t)$. According to the first stage procedure of new IBOSS algorithm,

$$C = \{c \mid |c - c^*| < \delta \text{ or } |c + c^*| < \delta\} = (a, b) \cup (-b, -a)$$

for some constants $a < b$.

All the proof works here is built with cases that $(a, b) \cap (-b, -a) = \emptyset$. For cases when $(a, b) \cap (-b, -a) \neq \emptyset$, one can rewrite $(a, b) \cap (-b, -a)$ as (a', b') for some constant a', b' and prove the same result using exactlty the same framework.

Denote $Var(c)$ by σ_c^2 and $Var(x_1)$ by σ_1^2 with $\sigma_1, \sigma_c > 0$. Thus by using the conditional distribution forms derived in Arnold et.al(48), we can directly obtain that

$$f_1(x_1) = \frac{\frac{1}{\sigma_1} e^{-\frac{(x_1-u_1)^2}{2\sigma_1^2}} g_1(x_1)}{\Phi(\frac{b-u_c}{\sigma_c}) - \Phi(\frac{a-u_c}{\sigma_c}) + \Phi(\frac{-a-u_c}{\sigma_c}) - \Phi(\frac{-b-u_c}{\sigma_c})}$$

and

$$f_2(v_1) = \frac{\frac{1}{\sigma_1} e^{-\frac{(v_1+u_1)^2}{2\sigma_1}} g_2(v_1)}{\Phi(\frac{b+u_c}{\sigma_c}) - \Phi(\frac{a+u_c}{\sigma_c}) + \Phi(\frac{-a+u_c}{\sigma_c}) - \Phi(\frac{-b+u_c}{\sigma_c})},$$

where

$$g_1(x_1) = \Phi(\frac{\frac{b-u_c}{\sigma_c} - \rho\frac{x_1-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c} - \rho\frac{x_1-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) + \Phi(\frac{\frac{-a-u_c}{\sigma_c} - \rho\frac{x_1-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{-b-u_c}{\sigma_c} - \rho\frac{x_1-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})$$

and

$$g_2(v_1) = \Phi(\frac{\frac{b+u_c}{\sigma_c} - \rho\frac{v_1+u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a+u_c}{\sigma_c} - \rho\frac{v_1+u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) + \Phi(\frac{\frac{-a+u_c}{\sigma_c} - \rho\frac{v_1+u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{-b+u_c}{\sigma_c} - \rho\frac{v_1+u_1}{\sigma_1}}{\sqrt{1-\rho^2}}).$$

Now we investigate the information matrix of subdata from the new IBOSS algorithm. By implementing the new algorithm, we will first pick candidate sample whose $c = X^T\beta \in C$. And for the two dimension case discussed here, the second stage of the procedure picks the data rows with the largest $\lceil \frac{r}{2} \rceil$ x_1 values, and the smallest $\lceil \frac{r}{2} \rceil$ x_1 values, in order to build the final subdata with size around r.

In other words, with the remaining datalines $(X'_1, \cdots, X'_{n_1})$ after first stage procedure, $(X'^1, \cdots, X'^{\lceil \frac{r}{2} \rceil})$ and $(X'^{n_1-\lceil \frac{r}{2} \rceil+1}, \cdots, X'^{n_1})$ will form the final subdata, where $X'^i = (1, x_1'^i, x_2'^i)$ is the covariate vector with the $i-th$ largest x'_1 values. Also one can easily find that random variable x'_{i1} follow distribution F_1 with pdf f_1, $i = 1, \cdots, n_1$.

As

$$\begin{aligned} I^{IBOSS} &= \sum_{i=1}^{n} \alpha_i \Psi(c_i) X_i X_i^T \\ &= \sum_{i=1}^{\lceil \frac{r}{2} \rceil} \Psi(c'_i) X'^i X'^{iT} + \sum_{i=n_1-\lceil \frac{r}{2} \rceil+1}^{n_1} \Psi(c'_i) X'^i X'^{iT}, \end{aligned} \tag{4.6}$$

to prove the theorem, the explicit forms for $(X'^1, \cdots, X'^{\lceil \frac{r}{2} \rceil})$ and $(X'^{n_1-\lceil \frac{r}{2} \rceil+1}, \cdots, X'^{n_1})$ need to be derived.

First let us focus on getting explicit form for $X'_1 = (1, x_1'^1, x_2'^1)$. As $x_1'^1$ is the largest value among $(x'_{11}, \cdots, x'_{n_1 1})$ and x'_{i1} are independently generate from F_1, if we can prove that F_1 belongs to the Gumbel type and $F_1^{-1}(1 - \frac{1}{n_1}) \to \infty$ as $n_1 \to \infty$, then the distribution of $x_1'^1$ will satisfy

$$F_{x_1'^1}(a_{n_1} x + b_{n_1}) = e^{-e^{-x}} \tag{4.7}$$

when $n_1 \to \infty$, where $a_{n_1} = \frac{\sigma_1^2(1-\rho^2)}{F_1^{-1}(1-\frac{1}{n_1})}$ and $b_{n_1} = F_1^{-1}(1 - \frac{1}{n_1})$.

By plugging in $x = \sqrt{F_1^{-1}(1-\frac{1}{n})}$ and $x = -\sqrt{F_1^{-1}(1-\frac{1}{n})}$ to (Equation 4.7), one can obtain $x_1'^1 = F_1^{-1}(1-\frac{1}{n_1}) + o(1)$. Then by Theorems 2.8.1 and 2.8.2 (49), we can derive

$$x_1'^i = F_1^{-1}(1-\frac{1}{n_1}) + o(1) \text{ for } i = 1, \cdots, \left\lceil \frac{r}{2} \right\rceil. \tag{4.8}$$

For $x_1'^{n_1-\lceil \frac{r}{2} \rceil+1}, \cdots, x_1'^{n_1}$, consider random variables $(v_{11} = -x_{11}', \cdots, v_{n_1 1} = -x_{n_1 1}')$ and $c_v = V^T\beta = -c$. Thus v_{i1} follows distribution F_2 for $i = 1, \cdots, n_1$ and $(-x_1'^{n-\lceil \frac{r}{2} \rceil+1}, \cdots, -x_1'^{n})$ will be corresponding to the largest $[\frac{r}{2}]$ values in $(v_{11}, \cdots, v_{n_1 1})$. Reorder $(v_{11}, \cdots, v_{n_1 1})$ as $((v_1^1, \cdots, v_1^{n_1}))$ in descending order. Similarly, by assuming that F_2 belongs to the Gumbel type and $F_2^{-1}(1-\frac{1}{n_2}) \to \infty$, we can get explicit forms for v_1^1, as:

$$v_1^1 = F_2^{-1}(1-\frac{1}{n_1}) + o(1).$$

Again by Theorems 2.8.1 and 2.8.2 from (49), we have

$$v_1^i = F_2^{-1}(1-\frac{1}{n_1}) + o(1) \to -\infty \text{ for } i = 1, \cdots, \left\lceil \frac{r}{2} \right\rceil,$$

which is equivalent to

$$x_1'^i = -F_2^{-1}(1-\frac{1}{n_1}) + o(1) \text{ for } i = n_1 - \left\lceil \frac{r}{2} \right\rceil + 1, \cdots, n_1. \tag{4.9}$$

As $c = \beta_0 + x_1\beta_1 + x_2\beta_2$ and $\rho \neq \pm 1$, we have $\beta_2 \neq 0$. By applying (Equation 4.8), (Equation 4.9) and the fact that $c \in C$ is bounded, we have

$$x_2'^i = -\frac{\beta_1 F_1^{-1}(1-\frac{1}{n})}{\beta_2} + O(1) \quad \text{for } i = 1, \cdots, \left\lceil \frac{r}{2} \right\rceil \tag{4.10}$$

and

$$x_2'^i = \frac{\beta_1 F_2^{-1}(1-\frac{1}{n})}{\beta_2} + O(1) \quad \text{for } i = n_1 - \left\lceil \frac{r}{2} \right\rceil + 1, \cdots, n_1 \tag{4.11}$$

Let e denote the minimum value for $\Psi(\beta) = p(\beta)(1-p(\beta)) = \frac{e^{X^T\beta}}{(1+e^{X^T\beta})^2}$ in range C. Thus apply (Equation 4.8), (Equation 4.9), (Equation 4.10), (Equation 4.11) to (Equation 4.6), we can prove

$$I^{\text{IBOSS}} \geq e \begin{pmatrix} r & I_{12} \\ I_{12}^T & I_{22} \end{pmatrix}$$

where

$$I_{12} = \begin{pmatrix} \left\lceil \frac{r}{2} \right\rceil (F_1^{-1}(1-\frac{1}{n_1}) - F_2^{-1}(1-\frac{1}{n_1})) + o(1) & \left\lceil \frac{r}{2} \right\rceil \frac{\beta_1}{\beta_2}(F_2^{-1}(1-\frac{1}{n_1}) - F_1^{-1}(1-\frac{1}{n_1})) + O(1) \end{pmatrix}$$

and

$$I_{22} = \begin{pmatrix} \lceil \frac{r}{2} \rceil ((F_1^{-1}(1-\frac{1}{n_1}))^2 + (F_2^{-1}(1-\frac{1}{n_1}))^2) + o(F) & -\lceil \frac{r}{2} \rceil (\frac{\beta_1 (F_1^{-1}(1-\frac{1}{n_1}))^2}{\beta_2} + \frac{\beta_1 (F_2^{-1}(1-\frac{1}{n_1}))^2}{\beta_2}) + O(F) \\ -\lceil \frac{r}{2} \rceil (\frac{\beta_1 (F_1^{-1}(1-\frac{1}{n_1}))^2}{\beta_2} + \frac{\beta_1 (F_2^{-1}(1-\frac{1}{n_1}))^2}{\beta_2}) + O(F) & \lceil \frac{r}{2} \rceil (\frac{\beta_1^2 (F_1^{-1}(1-\frac{1}{n_1}))^2}{\beta_2^2} + \frac{\beta_1^2 (F_2^{-1}(1-\frac{1}{n_1}))^2}{\beta_2^2}) + o(F) \end{pmatrix},$$

$F = max(F_1^{-1}(1-\frac{1}{n}), F_2^{-1}(1-\frac{1}{n})) \to \infty.$

Since $F_i^{-1}(1-\frac{1}{n_1}) \to \infty$, for $i = 1, 2$, then all the elements of the I_{22} part, which is for slope parameters, will goes to infinity.

Now the remaining part is to prove that the assumptions we used hold for F_i, $i = 1, 2$. They are

- F_i belongs to Gumbel type for $i = 1, 2$.
- $F_i^{-1}(1-\frac{1}{n_1}) \to \infty$ as $n_1 \to \infty$ for $i = 1, 2$.

Here, we will just show that the framework used to prove these assumptions with F_1, one can similarly prove them using a similar framework with F_2.

By Leadbetter et.al(50), the necessary and suffcient condition for distribution F_1 to be the Gumbel type is that

$$\lim_{t \to \infty} \frac{1 - F_1(t + xr(t))}{1 - F_1(t)} = e^{-x} \text{ for } x \in \Re, \tag{4.12}$$

where $r(t)$ is a positive function when t is big enough.

Thus, as long as (Equation 4.12) holds for F_1, the first assumption will holds for F_1.

Set $r(t) = \sigma_1^2(1-\rho^2)/(t-u_1)$, then $r(t) > 0$ for t big enough and $\lim_{t\to\infty} r(t) = \lim_{t\to\infty} r'(t) = 0$. Thus

$$\begin{aligned}\lim_{t\to\infty} \frac{1-F_1(t+xr(t))}{1-F_1(t)} &= \lim_{t\to\infty} \frac{f_1(t+xr(t))(1+xr'(t))}{f_1(t)} \\ &= \lim_{t\to\infty} \frac{e^{-\frac{(t+xr(t)-u_1)^2}{2\sigma_1^2}} g_1(t+xr(t))(1+xr'(t))}{e^{-\frac{(t-u_1)^2}{2\sigma_1^2}} g_1(t)} \\ &= \lim_{t\to\infty} \frac{e^{-\frac{(t+xr(t)-u_1)^2}{2\sigma_1^2}}}{e^{-\frac{(t-u_1)^2}{2\sigma_1^2}}} \lim_{t\to\infty} \frac{g_1(t+xr(t))}{g_1(t)}.\end{aligned} \tag{4.13}$$

Consider $\lim_{t\to\infty} \frac{e^{-\frac{(t+xr(t)-u_1)^2}{2\sigma_1^2}}}{e^{-\frac{(t-u_1)^2}{2\sigma_1^2}}}$ first, one can directly derive that

$$\begin{aligned}\lim_{t\to\infty} \frac{e^{-\frac{(t+xr(t)-u_1)^2}{2\sigma_1^2}}}{e^{-\frac{(t-u_1)^2}{2\sigma_1^2}}} &= \lim_{t\to\infty} e^{-\frac{(xr(t))^2}{2\sigma_1^2}} e^{-\frac{(xr(t)(t-u_1))}{\sigma_1^2}} \\ &= \lim_{t\to\infty} e^{-\frac{(xr(t)(t-u_1))}{\sigma_1^2}}.\end{aligned} \tag{4.14}$$

Thus by plugging in $r(t) = \sigma_1^2(1-\rho^2)/(t-u_1)$, we can obtain

$$\lim_{t\to\infty} \frac{e^{-\frac{(t+xr(t)-u_1)^2}{2\sigma_1^2}}}{e^{-\frac{(t-u_1)^2}{2\sigma_1^2}}} = e^{-(1-\rho^2)x}. \tag{4.15}$$

Then to calculate $\lim_{t\to\infty} \frac{g_1(t+xr(t))}{g_1(t)}$, first consider

$$\lim_{t\to\infty} \frac{\Phi(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{a+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}{\Phi(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}$$

Then, one can derive that

$$\lim_{t\to\infty} \frac{\Phi(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{a+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}{\Phi(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}$$

$$= \lim_{t\to\infty} \frac{\Phi'(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi'(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}{\Phi'(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi'(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}$$

$$= \lim_{t\to\infty} \frac{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1})^2}{2(1-\rho^2)}} \frac{-\rho(1+xr'(t))}{\sqrt{1-\rho^2}\sigma_1} - e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1})^2}{2(1-\rho^2)}} \frac{-\rho(1+xr'(t))}{\sqrt{1-\rho^2}\sigma_1}}{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} \frac{-\rho}{\sqrt{1-\rho^2}\sigma_1} - e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} \frac{-\rho}{\sqrt{1-\rho^2}\sigma_1}}.$$

$$= \lim_{t\to\infty} \frac{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1})^2}{2(1-\rho^2)}} - e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1})^2}{2(1-\rho^2)}}}{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} - e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}}(1+xr'(t))$$

$$= \lim_{t\to\infty} \frac{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} e^{-\frac{(\rho\frac{xr(t)}{\sigma_1})^2}{2(1-\rho^2)}} e^{\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}} - e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} e^{-\frac{(\rho\frac{xr(t)}{\sigma_1})^2}{2(1-\rho^2)}} e^{\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}}}{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} - e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}}$$

$(1+xr'(t))$.

If $\rho > 0$, one can show that

$$\lim_{t\to\infty} \frac{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}}{e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}} = \lim_{t\to\infty} \frac{e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} e^{-\frac{(\frac{b-a}{\sigma_c})^2}{2(1-\rho^2)}} e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\frac{b-a}{\sigma_c})}{(1-\rho^2)}}}{e^{\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}} \tag{4.16}$$

$$= \infty$$

Thus

$$\lim_{t\to\infty} \frac{\Phi(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{a+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}{\Phi(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}$$

$$= \lim_{t\to\infty} \left(\frac{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} e^{-\frac{(\rho\frac{xr(t)}{\sigma_1})^2}{2(1-\rho^2)}} e^{\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}}}{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}} - \frac{e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} e^{-\frac{(\rho\frac{xr(t)}{\sigma_1})^2}{2(1-\rho^2)}} e^{\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}}}{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}}\right)$$

$$(1 + xr'(t))$$

$$= \lim_{t\to\infty} \left(e^{-\frac{(\rho\frac{xr(t)}{\sigma_1})^2}{2(1-\rho^2)}} e^{\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}} - \frac{e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}} e^{-\frac{(\rho\frac{xr(t)}{\sigma_1})^2}{2(1-\rho^2)}} e^{\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}}}{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}}\right)(1 + xr'(t)) \tag{4.17}$$

And, since $\lim_{t\to\infty} r(t) = 0$, we have

$$\lim_{t\to\infty} e^{-\frac{(\rho\frac{xr(t)}{\sigma_1})^2}{2(1-\rho^2)}} e^{\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}} = \lim_{t\to\infty} e^{-\frac{(-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}}$$

$$= e^{-\rho^2 x}. \tag{4.18}$$

and

$$\lim_{t\to\infty} e^{-\frac{(\rho\frac{xr(t)}{\sigma_1})^2}{2(1-\rho^2)}} e^{\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}} = \lim_{t\to\infty} e^{-\frac{(-\rho\frac{t-u_1}{\sigma_1})(\rho\frac{xr(t)}{\sigma_1})}{(1-\rho^2)}} \\ = e^{-\rho^2 x}. \tag{4.19}$$

Applying (Equation 4.16),(Equation 4.18) ,(Equation 4.19) to (Equation 4.17), we have

$$\lim_{t\to\infty} \frac{\Phi(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{a+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}{\Phi(\frac{\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}$$

$$= \lim_{t\to\infty}(1+xr'(t))(e^{-\rho^2 x} - e^{-\rho^2 x}\lim_{t\to\infty}\frac{e^{-\frac{(\frac{a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}}{e^{-\frac{(\frac{b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1})^2}{2(1-\rho^2)}}}). \tag{4.20}$$

As $\lim_{t\to\infty} r'(t) = 0$,

$$= e^{-\rho^2 x}$$

Similarly, we can prove that

$$\lim_{t\to\infty} \frac{\Phi(\frac{\frac{-a-u_c}{\sigma_c}-\rho\frac{t+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{-b-u_c}{\sigma_c}-\rho\frac{a+xr(t)-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})}{\Phi(\frac{\frac{-a-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}}) - \Phi(\frac{\frac{-b-u_c}{\sigma_c}-\rho\frac{t-u_1}{\sigma_1}}{\sqrt{1-\rho^2}})} \tag{4.21}$$

$$= e^{-\rho^2 x}$$

Apply (Equation 4.20), (Equation 4.21) to $\lim_{t\to\infty} \frac{g_1(t+xr(t))}{g_1(t)}$, we can obtain

$$\lim_{t\to\infty} \frac{g(t+xr(t))}{g(t)} = e^{-\rho^2 x}$$

for the $\rho > 0$ case.

For the $\rho < 0$ case, one can follow similar frame work and get $\lim_{t\to\infty} \frac{g(t+xr(t))}{g(t)} = e^{-\rho^2 x}$.

For the $\rho = 0$ case, one can easily find $\lim_{t\to\infty} \frac{g(t+xr(t))}{g(t)} = 1 = e^{-\rho^2 x}$.

Thus, we have

$$\lim_{t\to\infty} \frac{g(t+xr(t))}{g(t)} = e^{-\rho^2 x} \tag{4.22}$$

as long as $\rho \neq \pm 1$.

With (Equation 4.22) and (Equation 4.14), (Equation 4.12) can be written as

$$\lim_{t\to\infty} \frac{1-F_1(t+xr(t))}{1-F_1(t)} = e^{-x} \text{ for } x \in \Re.$$

So the necessary and sufficient condition (Equation 4.12) holds and therefore the first assumption holds for F_1.

Now we prove the second assumption that $F_1^{-1}(1-\frac{1}{n_1}) \to \infty$ as $n_1 \to \infty$.

Suppose $F_1^{-1}(1-\frac{1}{n_1}) \to h < \infty$. Then there exists N, for all $n_1 > N$, we have $|F_1^{-1}(1-\frac{1}{n_1}) - h| < \epsilon_0$, where ϵ_0 is a fixed positive constant. Then consider

$$\begin{aligned}
\int_{h+\epsilon_0}^{\infty} f_1(x_1)\ dx_1 &\geq \int_{h+\epsilon_0}^{h+2\epsilon_0} f_1(x_1)\ dx_1 \\
&= \int_{h+\epsilon_0}^{h+2\epsilon_0} \frac{\frac{1}{\sigma_1} e^{-\frac{(x_1-u_1)^2}{2\sigma_1}} g_1(x_1)}{\Phi(\frac{b-u_c}{\sigma_c}) - \Phi(\frac{a-u_c}{\sigma_c}) + \Phi(\frac{-a-u_c}{\sigma_c}) - \Phi(\frac{-b-u_c}{\sigma_c})}\ dx_1 \\
&= \frac{1}{\Phi(\frac{b-u_c}{\sigma_c}) - \Phi(\frac{a-u_c}{\sigma_c}) + \Phi(\frac{-a-u_c}{\sigma_c}) - \Phi(\frac{-b-u_c}{\sigma_c})} \int_{h+\epsilon_0}^{h+2\epsilon_0} \frac{1}{\sigma_1} e^{-\frac{(x_1-u_1)^2}{2\sigma_1^2}} g_1(x_1)\ dx_1
\end{aligned}$$

Since $g_1(x_1)$ is a positive continuous function of x_1 and x_1 is bounded, thus the minimum value of $g(x_1)$ is $g_0 > 0$. Then

$$\begin{aligned}
\int_{h+\epsilon_0}^{\infty} f_1(x_1)\ dx_1 &\geq \frac{g_0}{\Phi(\frac{b-u_c}{\sigma_c}) - \Phi(\frac{a-u_c}{\sigma_c}) + \Phi(\frac{-a-u_c}{\sigma_c}) - \Phi(\frac{-b-u_c}{\sigma_c})} \int_{h+\epsilon_0}^{h+2\epsilon_0} \frac{1}{\sigma_1} e^{-\frac{(x_1-u_1)^2}{2\sigma_1^2}}\ dx_1 \\
&= \frac{g_0}{\Phi(\frac{b-u_c}{\sigma_c}) - \Phi(\frac{a-u_c}{\sigma_c}) + \Phi(\frac{-a-u_c}{\sigma_c}) - \Phi(\frac{-b-u_c}{\sigma_c})} \sqrt{2\pi}(\Phi(\frac{h+2\epsilon_0-u_1}{\sigma_1}) - \Phi(\frac{h+\epsilon_0-u_1}{\sigma_1})) \\
&\geq l > 0
\end{aligned}$$

Thus, as long as we take any n satisfy $n \geq \frac{1}{l}$, we can obtain $F_1^{-1}(1-\frac{1}{n}) > h + \epsilon_0$, which is conflict with our assumption. Thus $F_1^{-1}(1-\frac{1}{n_1}) \to \infty$ if $\rho \neq \pm 1$. The second assumption also holds for F_1.

□

APPENDIX

PERMISSION OF COPYRIGHT HOLDER

Springer
the language of science

Title: On Multiple-Objective Nonlinear Optimal Designs
Author: Qianshun Cheng
Publication: Springer eBook
Publisher: Springer
Date: Jan 1, 2016

Copyright © 2016, Springer International Publishing Switzerland

Logged in as:
Qianshun Cheng
Account #:
3001173915
LOGOUT

Order Completed

Thank you for your order.

This Agreement between Qianshun Cheng ("You") and Springer ("Springer") consists of your license details and the terms and conditions provided by Springer and Copyright Clearance Center.

Your confirmation email will contain your order number for future reference.

Printable details.

License Number	4159111003158
License date	Jul 30, 2017
Licensed Content Publisher	Springer
Licensed Content Publication	Springer eBook
Licensed Content Title	On Multiple-Objective Nonlinear Optimal Designs
Licensed Content Author	Qianshun Cheng
Licensed Content Date	Jan 1, 2016
Type of Use	Thesis/Dissertation
Portion	Full text
Number of copies	1
Author of this Springer article	Yes and you are a contributor of the new work
Order reference number	
Title of your thesis / dissertation	Novel Algorithm for Constrained Optimal Design and Information-based Subdata Selection for Logistic Model
Expected completion date	Aug 2017
Estimated size(pages)	108
Requestor Location	Qianshun Cheng 7514 Ethel Avenue Richmond Heights SAINT LOUIS, MO 63117 United States Attn: Qianshun Cheng
Billing Type	Invoice
Billing address	Qianshun Cheng 7514 Ethel Avenue Richmond Heights SAINT LOUIS, MO 63117 United States Attn: Qianshun Cheng
Total	0.00 USD

ORDER MORE CLOSE WINDOW

CITED LITERATURE

1. Chernoff, H.: Locally optimal designs for estimating parameters. Annals of Mathematical Statistics, 24:586–602, 1953.

2. Yang, M. and Stufken, J.: Support points of locally optimal designs for nonlinear models with two parameters. Annals of Statistics, 37:518–541, 2009.

3. Yang, M. and Stufken, J.: Identifying locally optimal designs for nonlinear models: A simple extension with profound consequences. Annals of Statistics, 40:1665–1681, 2012.

4. Yang, M.: On the de la garza phenomenon. Annals of Statistics, 38:2499–2524, 2010.

5. Dette, H. and Melas, V. B.: A note on the de la garza phenomenon for locally optimal designs. Annals of Statistics, 39:1266–1281, 2011.

6. Dette, H. and Schorning, K.: Complete classes of designs for nonlinear regression models and principal representations of moment spaces. Annals of Statistics, 41:1260–1267, 2013.

7. Pronzato, L. and Pázman, A.: Design of Experiments in Nonlinear Models. Springer, 2013.

8. Yang, M., Biedermann, S., and Tang, E.: On optimal designs for nonlinear models: a general and efficient algorithm. Journal of the American Statistical Association, 108:1411–1420, 2013.

9. Rosenberger, W. F. and Grill, S. E.: A sequential design for psychophysical experiments: an application to estimating timing of sensory events. Statistics in Medicine, 16:2245–2260, 1997.

10. Park, Y. J.: Multi-optimal designs for second-order response surface models. Communications of the Korean Statistical Society, 16(1):195–208, 2009.

11. Park, L., Anderson-Cook, C. M., and Robinson, T. J.: Optimization of designed experiments based on multiple criteria utilizing a pareto frontier. Technometrics, 61:353–365, 2011.

12. Cao, Y., Smucker, B. J., and Robinson, T. J.: On using the hypervolume indicator to compare pareto fronts: Applications to multi-criteria optimal experimental design. Journal of Statistical Planning and Inference, 160:60–74, 2015.

13. Clyde, M. and Chaloner, K.: The equivalence of constrained and weighted designs in multiple objective design problems. Journal of the American Statistical Association, 91:1236–1244, 1996.

14. Cook, R. D. and Wong, W. K.: On the equivalence of constrained and compound optimal designs. Journal of the American Statistical Association, 89:687–692, 1994.

15. Huang, Y. C. and Wong, W. K.: Sequential construction of multiple-objective optimal designs. Biometrics, 54:1388–1397, 1998.

16. Mandal, S., Torsney, B., and Carriere, K. C.: Constructing optimal designs with constraints. Journal of Statistical Planning and Inference, 128:609–621, 2005.

17. Sagnol, G. and Harman, R.: Computing exact d-optimal designs by mixed integer second order cone programming. Annals of Statistics, 43:2198–2224, 2015.

18. Mandal, A., Wong, W. K., and Yu, Y.: Algorithmic searches for optimal designs. Handbooks on Modern Statistical Methods, Design of Experiments (Bingham, D., Dean, A., Morris, M., and Stufken, J. ed.), pages 755–783, 2015.

19. Yu, Y.: D-optimal designs via a cocktail algorithm. Statistics and Computing, 21:475–481, 2011.

20. Pukelsheim, F.: Optimal Design of Experiments. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 2006.

21. Notari, R. E.: Biopharmaceutics and Clinical Pharmacokinetics. New York and Basel: Marcel Dekker, 1980.

22. Dette, H., Bretz, F., Pepelyshev, A., and Pinheiro, J.: Optimal designs for dose-finding studies. Journal of the American Statistical Association, 103:1225–1237, 2008.

23. Atkinson, A. C., Chaloner, K., Juritz, J., and Herzberg, A. M.: Optimum experimental designs for properties of a compartmental model. Biometrics, 49:325–337, 1993.

24. Cook, D. and Fedorov, V.: Constrained optimization of experimental design. Statistics, 26:129–178, 1995.

25. Mattmann, C. A., Hart, A., L., C., Lazio, J., Khudikyan, S., Jones, D., Preston, R., Bennett, T., Bulter, B., Harland, D., Glendenning, B., Kern, J., and Robnett, J.: Scalable data mining, archiving, and big data management for the next generation astronomical telescopes. In Big Data Management, Technologies, and Applications, eds. W.-C. Hu and N. Kaabouch, pages 196–221. IGI Global, 2014.

26. Lin, N. and Xi, R.: Aggregated estimating equation estimation. Statistics and Its Interface, 4:73–83, 2011.

27. Chen, X. and Xie, M.-g.: A split-and-conquer approach for analysis of extraordinarily large data. Statistica Sinica, 24:1655–1684, 2014.

28. Schifano, E. D., Wu, J., Wang, C., Yan, J., and Chen, M.-H.: Online updating of statistical inference in the big data setting. Tech. Rep., Department of Statistics, University of Connecticut, Storrs, Connecticut, pages 14–22, 2014.

29. Politis, D. N., Romano, J. P., and Wolf, M.: Subsampling. 1999.

30. Efron, B.: Bootstrap methods: Another look at the jackknife. The annals of Statistics, 7:1–26, 1979.

31. Bickel, P. J., Gotze, F., and van Zwet, W. R.: Resampling fewer than n observations: Gains, losses, and remedies for losses. Statistica Sinica, 7:1–31, 1997.

32. Kleiner, A., Talwalkar, A., Sarkar, P., and Jordan, M. I.: A scalable bootstrap for massive data. Journal of the Royal Statistical Society: Series B, 76:795–816, 2014.

33. Liang, F., Cheng, Y., Song, Q., Park, J., and Yang, P.: A resampling-based stochastic approximation method for analysis of large geostatistical data. Journal of the American Statistical Association, 108:325–339, 2013.

34. Liang, F. and Kim, J.: A bootstrap metropolishastings algorithm for bayesian analysis of big data. Tech. rep., Department of Statistics, Texas A & M University, 2013.

35. Ma, P. and Sun., X.: Leveraging for big data regression. Wiley Interdisciplinary Reviews: Computational Statistics, 7:70–76, 2015.

36. Ma, P., Mahoney, M., and Yu, B.: A statistical perspective on algorithmic leveraging. In Proceedings of the 31st International Conference on Machine Learning (ICML-14), pages 91–99, 2014.

37. Wang, H., Yang, M., and Stufken, J.: Information-based optimal subdata selection for big data in linear models. Journal of the American Statistical Association, invited revision under review.

38. Cox, D.: The regression analysis of binary sequences. Journal of the Royal Statistical Society. Series B, 20(2):215–242, 1958.

39. Wang, H., Zhu, R., and Ma, P.: Optimal subsampling for large sample logistic regression. Journal of the American Statistical Association, 2017.

40. Kiefer, J.: Optimum experimental designs. Journal of the Royal Statistical Society. Series B, 21:272–319, 1959.

41. Yildirim, A. A., Özdogan, C., and Watson, D.: Parallel data reduction techniques for big datasets. In Big Data Management, Technologies, and Applications, eds. W.-C. Hu and N. Kaabouch, pages 72–93. IGI Global, 2014.

42. Hosmer, D. W. and Lameshow, S.: Applied logistic regression.. Join Wiley and Sons, New York, 2000.

43. Gourieroux, C. and Monfort, A.: Asymptotic properties of the maximum likelihood estimator in dichotomous models. Journal of Econometrics, 17:83–97, 1981.

44. Gourieroux, C. and Monfort, A.: Asymptotic properties of the maximum likelihood estimator in dichotomous logistic regression models. Diploma Thesis in Mathematics. University of Fribourg Switzerland, 2001.

45. Rashid, M. and Shifa, N.: Consistency of the maximum likelihood estimator in logistic regression model: a different approach. Journal of Statistics, 16:1–11, 2009.

46. Nordberg, L.: Asymptotic normality of maximum likelihood estimators based on independent, unequally distributed observations in exponential family models. Scandinavian Journal of Statistics, 24:1655–1684, 1980.

47. Yang, M., Zhang, B., and Huang, S.: Optimal designs for binary response experiments with multiple variables. Statistica Sinica, 21:1415–1430, 2011.

48. Arnold, B., Beaver, R., Groeneveld, R., and Meeker, W.: The nontruncated marginal of a truncated bivariate normal distribution, volume 58. 1993.

49. Galambos, J.: The asymptotic theory of extreme order statistics. large data.. Florida: Robert E. Krieger, 1987.

50. Leadbetter, M. R., Lindgren, G., and Rootzen, H.: Extremes and related properties of random sequences and processes. Springer, New York, 1983.

VITA

Name: **Qianshun Cheng**

Education:

2007-2011 Bachelor of Science, Xiamen University.

- Major in Mathematics and Applied Mathematics, Minor in Economy.

2011-2012 MS in Mathematics, University of Illinois at Chicago, IL

2011-Present PhD in Mathematics, University of Illinois at Chicago, IL

Abstract:

1. Cheng, Q. and Yang, M.: On Multiple-Objective Optimal Designs. (Ready to submit)
2. Cheng, Q., Yang, M. and Wang, H.: Information-Based Optimal Subdata Selection for Big Data Logistic Regression. (Ready to submit)

Publications:

- Cheng, Q., Gao, X. and Ryan M.: Exact prior-free probabilistic inference on the heritability coefficient in a linear mixed model. Electronic Journal of Statistics. 8:3062–3076, 2014.

Honors:

- 2014-2015 Reseach Award, Statistic Group, department of Mathematics, Statistics and Computer Science, University of Illinois at Chicago.

- 2011-2016 Full graduate scholarship, department of Mathematics, Statistics and Computer Science, University of Illinois at Chicago.